Dr. goodfit: finding the top modeling technique to streamline data analysis

Jennefer Maldonado

Applied Mathematics and Statistics, Stony Brook University, Stony Brook, NY 11794

Daniel Olds

Photon Sciences, Brookhaven National Laboratory, Upton, NY 11973

Abstract

At the National Synchrotron Light Source II (NSLS-II) researchers are able to study materials in order to gain information about next generation energy applications, as well as the electronics and computers of the future. As these studies progress, large amount of data rapidly accumulate that need to be analyzed. The speed of modern synchrotron data collection has outperformed data analysis techniques for years. A new approach to this problem, machine learning, seeks to automate data analysis preventing researchers from having to hand pick through large quantities of data. The goal of this project is to identify if interpolation methods can be used to better process noisy data. This project will also develop methods to construct new data points given limited information in order to optimize the beamlines in real time. This will benefit situations where it is unknown whether it is necessary to continue measuring samples to get a clearer reading. This project will evaluate various methods to create mathematical models, such as an ensemble model, which analyze data and determine which are best fit for applications at NSLS-II. I now am proficient in machine learning techniques such as supervised, unsupervised, reinforcement learning, and their implementations in Python. I have gained skills in software development, big data analysis, mathematical modeling, and how to apply all of these newly learned techniques to real world problems.

I. INTRODUCTION

The National Synchrotron Light Source II (NSLS-II) generates an unprecedented amount of data from experiments designed to discover next generation energy applications and electronics of the future. Once data is measured, it must be processed and analyzed by a scientist. As data sets become large, conventional hand-made analysis can become intractable and there are promising machine learning techniques for automation. In a situation where a scientist is uncertain whether further measurements are required or if it is necessary to repeat measurements, regression can predict future trends to provide a clear idea. Conventional models are able to interpolate points and regress linear, quadratic, and cubic equations. However, real world data is imperfect posing major risks for conventional techniques to over and underfit the data. It is difficult to then confirm a line of best fit due to the noise. A technique called ensemble modeling combines multiple neural networks and averages the predictions. When there is high diversity in individual models the ensemble model yields better results and balances out models that overfit and underfit given data[1]. Therefore, the ensemble model provides a greater certainty that a line of best fit has been found. In this paper, I will discuss the importance of tools such as machine learning, interpolation, and regression at NSLS-II and discuss how ensemble models improve the analysis of real-world data.

II. BACKGROUND

A. Machine Learning

Machine learning is the study of training a computer to accomplish a given task or method. The computer in turn should be able to complete the task despite not being programmed explicitly to do so, only using what it has learned from its training[2]. In order to train a machine on given data, a neural network is often used. A neural network is a series of algorithms designed

to parallel the neuron structure of the human brain. They were also designed to identify patterns. They can be used in a wide variety of applications such as classification, clustering, and regression[3]. Tensorflow is a platform used in Python to develop machine learning programs. Tensorflow contains Keras which is an API that allows for neural network development[4]. All code for this project is developed in Python using Tensorflow and Keras.

B. Interpolation

Experiments are summarized by data points in a graph, and it is useful to continuously update the summary throughout the experiment's duration. Interpolation can be used in this project for data reconstruction and gaining more information from a discrete data set. In a situation where an experiment doesn't provide the amount or quality of data required, interpolation is used to form new data points in between the points already measured. With more data points available, it becomes easier to fit a simpler function to the data[5]. This saves beamtime at NSLS-II as well as other facilities in need of data interpolation. Conversely, there can be a situation where a scientist is unable to take more measurements and the tool assists in a similar manner. There are two types of samples that can be measured. The first kind is a physical sample which can be comprised of a single material or a systematic variation of the material. The second is taking multiple measurements of a single sample to increase knowledge for statistics or to have data for a sample at different time periods. Both of these types of samples are talked about in this paper. First, examining the scattering pattern of a terbium cobalt and using interpolation to reconstruct the pattern. Second, predicting the intensity of an image given the burn and heal time over five second measurements. These are only two applications that interpolation is useful for.

C. Regression

To find a line of best fit, regression is the statistical tool that will aid in accomplishing this. The tool attempts to fit the maximum number of points in a data set to a line with the least amount of error. The error is computed by determining the sum of the squared difference between the predicted line and the points in the data set[6].

Regression is an important tool for determining future trends since it has the ability to accurately predict outside of the range of a discrete data set. For this project, regression is a useful tool when a scientist is unsure whether it is worth their time to continue measuring a sample or measuring the sample in the same experimental condition. If there is an upward trend towards the end of the measurements, regression may predict a peak which can be important data for an experiment. This in turn provides scientists with certainty to continue or allows them to prioritize different experimental conditions for the same physical sample.

III. MOTIVATION

In machine learning when a model provides unsatisfactory results it is due to overfitting and underfitting the data. For a neural network to have satisfactory performance, the network should be able to learn concepts from training data and apply this to examples it has not seen during the training process. Generalization is the idea that the neural network was able to apply these concepts learned in training in a general way to successfully predict results on new data. If a model happens to train too well, meaning it picks up on the noise or other artifacts in the data, it is considered an overfit model. Since this model has now learned the fine details of the training dataset, it is difficult to generalize these concepts to new data thus lowering the accuracy and performance of the model.

An underfitting model can also not generalize to new data but also fails to learn concepts from the training data. Underfitting also results in lowered performance and accuracy and can be fixed using a deeper network or another model structure[7].

Real world data is not as smooth as defined equations such as a linear, quadratic, or cubic equation. The data becomes messy and noisy leading to the overfitting and underfitting we wish to avoid in models. The model is more susceptible to overfitting due to the messy details in these datasets which does not aid in understanding the data as a scientist expects. If a scientist has sparse data, the model will also have difficulty training due to limited data and high noise. In an ideal situation there is a model that is perfectly in the middle of overfitting and underfitting that benefits real-world data analysis.

IV. ENSEMBLE MODEL INTERPOLATION

A. Ensemble Learning

Ensemble learning is a technique that includes multiple independent models to improve predictions or classifications for a machine learning problem. In this paper, multiple individual neural networks with distinct initial conditions are used to create an ensemble neural network. The distinct initial conditions allow the model as whole to effectively explore the global phase-space of possibilities. Each individual network has an output layer, which are averaged together using TensorFlow's average layer.

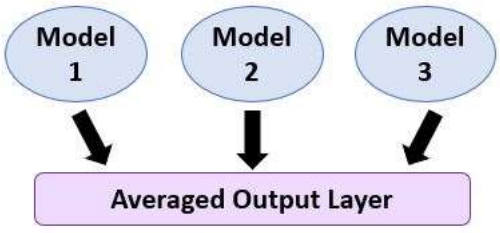

Figure 1 Ensemble Model Flow Chart

The technique of ensemble learning is unique due to the use of multiple models for more confidence. Predicting on a single model can lead to unwanted results because of overfitting, underfitting, or other factors about the model configuration. When there is a group of models, the performance will be less impacted by a single poor performing model[8]. Therefore, greater diversity in each individual model yields to better results for the entire ensemble[1].

B. The Ensemble Class

In the Motivation section the problem of overfitting and underfitting was discussed. Since an ensemble model is less impacted by a single model's poor performance, the hypothesis is that this model will also prevent the effects of the over and under fits. Incorporating interpolation to this model aids in the reduction of noise in datasets and provides more information despite having a discrete dataset. For this project I developed a class in Python to create an ensemble model given certain parameters. The user is responsible for choosing the ensemble model structure by entering parameters for the number of models, inputs, outputs, hidden layers, and nodes. The number of inputs and outputs is determined by the size and shape of the desired data set, this creates the input and output layers. The hidden layers are added in between the input and output layers. The greater number of hidden layers and nodes for each layer creates a deeper model. The user also decides how many epochs the model should train on the data. For each epoch the model trains on the data once which means sufficient epochs provide stronger results. Then the user passes as a parameter the desired training data and test data.

The class contains four methods that create models and train on the data. The `build model` method loops through the number of hidden layers desired, creates a model based on the given parameters, and returns a newly created model. The next method trains each individual model by a period also determined by the user. This method ensures each model is checkpointing

its progress and provides useful metrics after the ensemble model has completed training. The `ensemble` method calls upon these two methods in order to create the ensemble model and to average the output layers of each individual model. Once it runs each model and averages the output layers, the method returns the built ensemble model. The final method is dedicated to computing the residuals in each model. Residuals are the difference between the actual test values and predicted values. It is desired that the residuals trend close to zero to ensure a good performance by the model. For more information about the code visit https://github.com/jennmald/EnsembleModelClass.

C. Applications

1. Intensity as a Function of Burn and Heal Time

When a sample is measured on the flat panel image detector on a beamline at NSLS-II, as it decays it will leave a burn in effect for some time after it has been removed. This effect produces a “light” image. This burnt effect will dissipate over time but if a new sample is measured before the proper cool down time is allocated a residual burn will appear damaging the new light image. If a scientist had a prediction as to when the ideal time is to measure a new sample with the least amount of residual burn, there would be fewer damaged images in experiments.

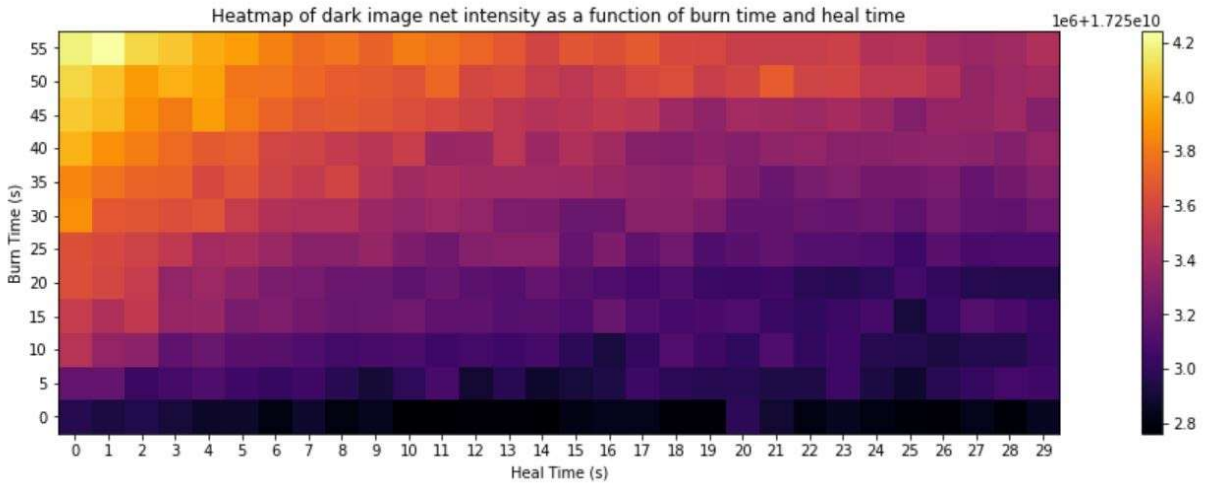

Figure 2 Heatmap of Raw Data

Figure 2 displays the dark image intensity in a heatmap which resembles a gradient. The intensity here represents the sum of all pixels on the Perkin-Elmer flat panel detector at the Pair Distribution Function (PDF) beamline as opposed to directly studying the dark image. A dark image measures the inherent signal that the detector measures when light is not hitting it directly. For the highest burn time and lowest heal time the highest intensity is produced. The lowest intensity occurs for the lower burn times and higher heal times. Each measurement is taken every five seconds, so our idea was to interpolate the data and predict intensities at times that were not measured in the raw data.

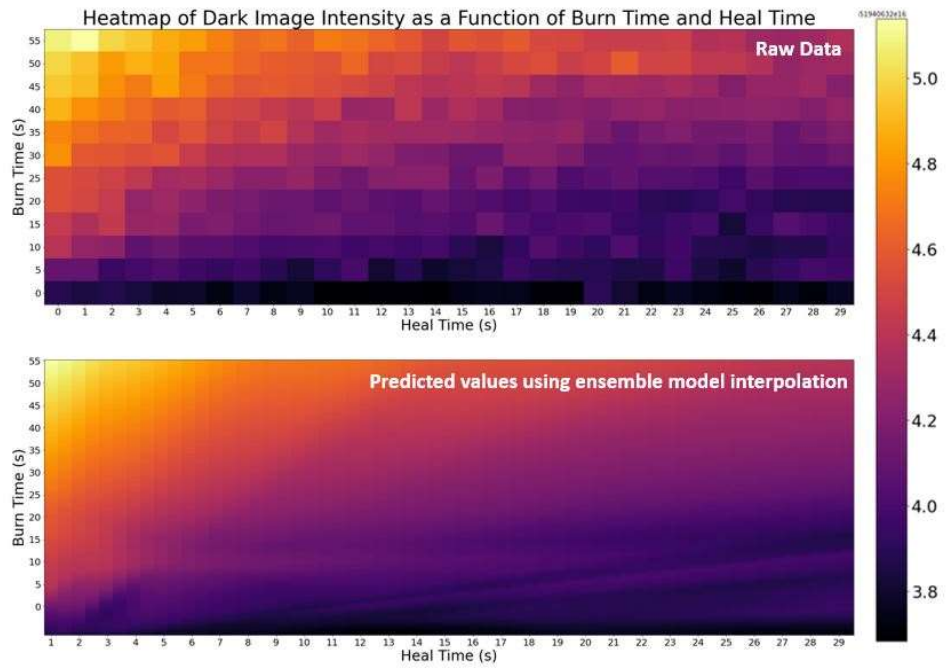

Figure 3 Comparison of Predicted Heatmap

The ensemble model, created with twenty models, received two inputs, a burn and heal time with respective intensity, and predicted the intensities for half second intervals. With these new predictions there is a new gradient corresponding to the raw data but smoother and defined with minimal noise. This prediction map now allows scientists to determine the correct timing for measuring new samples with limited burn residual and provides more insight into how long a burn effect can linger.

2. Image reconstruction

The Coherent Soft X-ray Scatter (CSX) beamline provides an understanding into the oxidation state of elements at varying energies, which is associated with a material's novel magnetic or electronic properties. Using different energy values provides different scattering patterns for a given element and demonstrates the sensitivity of CSX to magnetic and electronic order. Below are different scattering patterns for a terbium cobalt sample. Figure 4 displays the

expected speckled ring caused by the coherent x-ray light at CSX. The outer ring seen in Figure 5 signifies the presence of magnetic order and is speckled due to the coherent light.

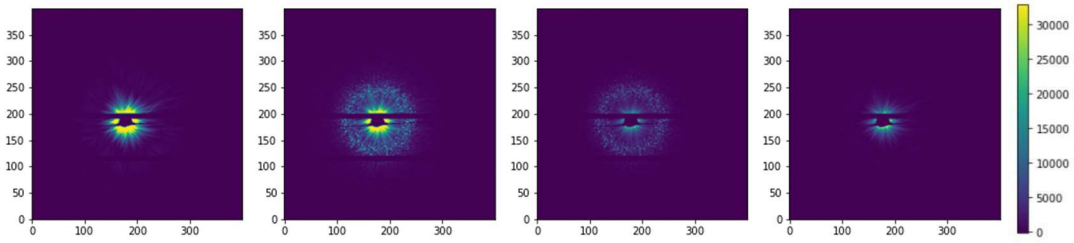

Figure 4: Terbium cobalt sample diffraction patterns for 764, 773, 775, 795 eV

In Figure 4, there are four different scattering images and each has a missing portion of the image at around y = 100, towards the center. Figure 5 highlights this part of the image. After the results with the interpolated heatmap from the previous section, the idea for these images is to train an ensemble model on a masked portion of an image, then use the trained model to construct the missing portion of the image.

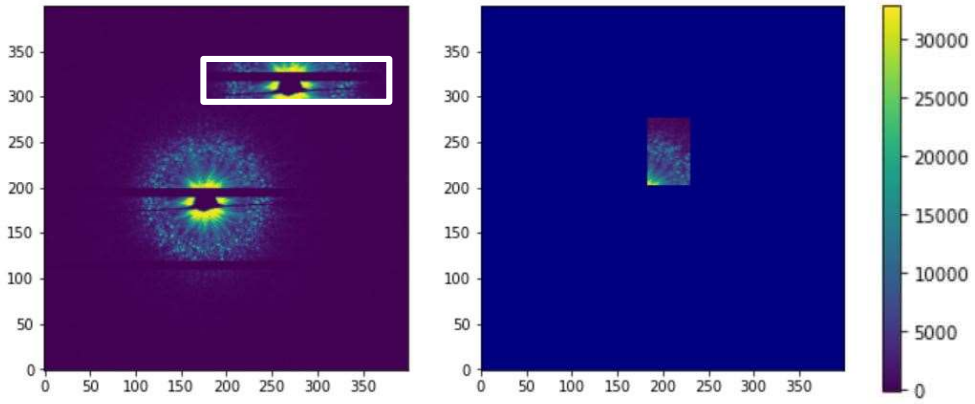

Figure 5 Diffraction patter from tbCo at 773eV

In order to do this, a simpler case was first examined. Here, only a one-dimensional portion of the image is trained on, keeping the y value constant. This line avoids any gaps in the image to avoid overfitting in the model.

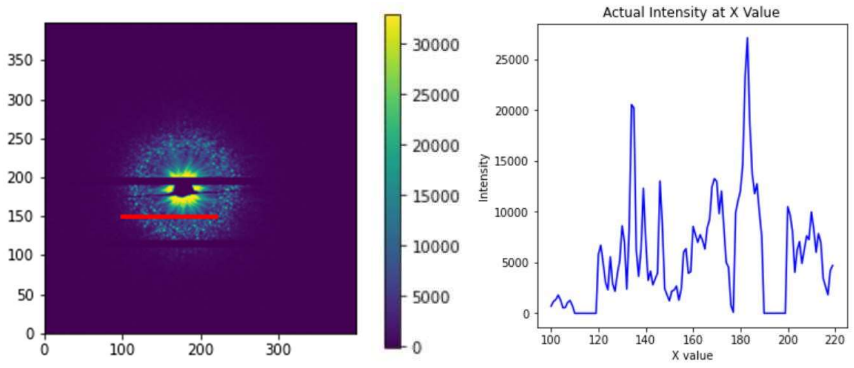

Figure 6 Image with Highlighted Training Data and Corresponding Intensity Graph

The ensemble model predicts intensities that are similar to the actual values as seen in Figure 7. The model is able to learn where the higher intensity spots in the image are and predict similar values given the x values to test on.

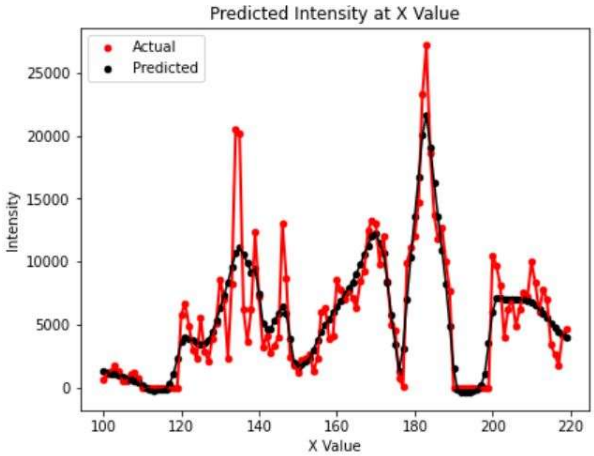

Figure 7 Actual Values vs Ensemble Model Predicted Values

Reconstructing Figure 4 will be discussed in the following section.

V. FUTURE WORK AND CONCLUSIONS

The goal of this project was to develop a general-purpose tool that can be used on a wide variety of data for interpolation and visualization purposes. The ensemble model class is a useful prediction tool and continuously provides checkpoints on training progress for insight into accuracy and loss. An ensemble model uses multiple models to overcome the problems of

overfitting and underfitting which have made it an ideal tool for real-world datasets like the ones generated at NSLS-II. Only two applications were discussed in this paper, but now with the tool developed scientists can apply it to their own data to improve the predictive data analysis for experiments.

The work done for reconstructing an image needs improvement in order for accurate use on real world data. At the moment, the ensemble model is continuously underfitting the data despite the attempts to deepen the model. The next steps are to continue to deepen the model and allow for enough learning time to lower the loss as far as possible. This requires more computational power using Google Colab and more time for the model to properly train.

VI. ACKNOWLEDGEMENTS

I would like to give a thank you to Daniel Olds, Andi Barbour, Joshua Lynch, Phil Maffettone, and Garrett Bischof for their kindness, knowledge, and support during this project. I would also like to thank my teammates, Clara, Víctor, Jake, and Alex, who made this summer the most enjoyable virtual experience I could have asked for. This project was supported in part by the U.S. Department of Energy, Office of Science, Office of Workforce Development for Teachers and Scientists (WDTS) under the Science Undergraduate Laboratory Internships Program (SULI).

VII. REFERENCES

1. Ensemble learning. (2020, July 13). Retrieved July 27, 2020, from https://en.wikipedia.org/wiki/Ensemble_learning
2. Faggella, D. (2020, February 26). What is Machine Learning? - An Informed Definition. Retrieved August 14, 2020, from https://emerj.com/ai-glossary-terms/what-is-machine-learning/
3. Pathmind. (2020, August 14). Retrieved August 14, 2020, from https://pathmind.com/
4. Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., & Corrado, G. S. (2015). TensorFlow. Retrieved from https://www.tensorflow.org/
5. The Editors of Encyclopaedia Britannica. (2016, October 16). Interpolation. Retrieved August 14, 2020, from https://www.britannica.com/science/interpolation
6. The Editors of Encyclopaedia Britannica. (2019, September 24). Regression. Retrieved from https://www.britannica.com/topic/regression-statistics
7. Brownlee, J. (2019, August 12). Overfitting and Underfitting With Machine Learning Algorithms. Retrieved August 14, 2020, from https://machinelearningmastery.com/overfitting-and-underfitting-with-machine-learning-algorithms/
8. Robi Polikar (2009) Ensemble learning. Scholarpedia, 4(1):2776.