

Automated Well to Well Correlation: A Machine Learning Study

BACHELOR THESIS

Fajar Tri Anggoro

12216024

Submitted as partial fulfillment of the requirements for the degree of

BACHELOR OF ENGINEERING

in Petroleum Engineering study program



PETROLEUM ENGINEERING STUDY PROGRAM
FACULTY OF MINING AND PETROLEUM ENGINEERING
INSTITUT TEKNOLOGI BANDUNG

2020

Automated Well to Well Correlation: A Machine Learning Study

BACHELOR THESIS

Fajar Tri Anggoro

12216024

Submitted as partial fulfillment of the requirements for the degree of
BACHELOR OF ENGINEERING
in Petroleum Engineering study program

Approved by:

Thesis Adviser,

.....

Dr. Dedy Irawan, S.T, M.T

197511052010121001

Automated Well to Well Correlation: A Machine Learning Study

Fajar Tri Anggoro* and Dedy Irawan**

Copyright 2020, Institut Teknologi Bandung

Abstract

The process of well to well correlation can be time-consuming as well log data grows massively. The conventional approach inclines to be subjective as it is based on one's perspective on the data. In this study, an automated well to well correlation using the machine learning method was used. Furthermore, this method can be considered to be convenient, and time-saving compared to the traditional approach.

A supervised learning method was used in this study, five types of logging data were used and labeled at a certain depth. Over ten thousand data points were used as an input and output of the machine learning model. 70% of the data was used to train the model while the other 30% was used to validate the trained model. Different approaches were used to create the model. The model was then tested on log data from different well to see the correlation within. Furthermore, the hyperparameter optimization was used as model evaluation to seek the best parameter. The model with the best performance was then justified as the selected model.

Three models were created using the K-Nearest Neighbors, Stochastic Gradient Descent, and Multilayer Perceptron approach respectively. Overall, the K-Nearest Neighbor approach was justified as the selected model. With a cross validation score of 0.981, the model resulted in 95.35% and 87.92% of accuracy on a first test set and second test set respectively. The created model based on the machine learning approach can harness massive log data and can be used on well to well correlation in a less time-consuming manner.

Keywords: well to well correlation, machine learning, supervised learning, K-Nearest Neighbors, Stochastic Gradient Descent, Multilayer Perceptron

Sari

Proses korelasi antar sumur dapat memakan waktu seiring dengan bertambahnya data log sumur. Pendekatan konvensional cenderung bersifat subjektif karena berdasarkan perspektif seseorang terhadap data. Dalam studi ini, digunakan sebuah korelasi antar sumur otomatis menggunakan metode pembelajaran mesin. Terlebih lagi, metode dapat dianggap mudah, dan hemat waktu dibandingkan dengan pendekatan tradisional.

Sebuah metode pembelajaran terawasi digunakan dalam studi ini, lima jenis data log sumur digunakan dan ditandai pada kedalaman tertentu. Lebih dari sepuluh ribu data digunakan sebagai input dan output pada model machine learning yang digunakan. 70% dari data digunakan untuk melatih model sementara 30% data lainnya digunakan untuk validasi model yang telah dilatih. Pendekatan – pendekatan berbeda telah digunakan dalam membuat model. Model lalu diuji dengan log data yang berasal dari sumur berbeda untuk dilihat korelasinya. Terlebih lagi, optimisasi hyperparameter digunakan sebagai evaluasi model untuk memperoleh parameter terbaik. Model dengan performa terbaik kemudian dijustifikasi sebagai model yang dipilih.

Tiga model dibuat dengan menggunakan pendekatan K-Nearest Neighbors, Stochastic Gradient Descent, dan Multilayer perceptron secara berurut. Secara keseluruhan, pendekatan K-Nearest Neighbors telah dijustifikasi sebagai model yang dipilih. Dengan skor validasi silang sebesar 0,981, model menghasilkan 95,35% dan 87,92% akurasi dalam set uji pertama dan set uji kedua secara berurut. Model yang telah dibuat berdasarkan pendekatan pembelajaran mesin mampu memanfaatkan data log massif dan dapat digunakan untuk korelasi antar sumur secara lebih sedikit memakan waktu.

Kata kunci: Korelasi antar sumur, pembelajaran mesin, pembelajaran terawasi, K-Nearest Neighbors, Stochastic Gradient Descent, Multilayer Perceptron

*) Student of Petroleum Engineering Study Program, Institut Teknologi Bandung, 2016 batch

**) Thesis Adviser in Petroleum Engineering Study Program, Institut Teknologi Bandung

1. Introduction

The well log analysis is one of the fundamental process in the development of oil and gas field, in which the process includes several other processes, one of which is the well to well correlation. The correlation process itself is a fundamental part of the stratigraphy. By definition, stratigraphy correlation is an act of measurement of the equivalent in stratigraphy unit, which in further definition is the measurement of equivalent in lithology, paleontology, and chronology.

In this study, the main focus is the lithology correlation using well log instrument, in the form of curves that represent the measurement of several properties of the well in the basis of electrical resistivity, acoustic transmission, and the absorption & emission of nuclear radiation. These variance itself is the reflection of lithology, mineralogy, or fluid content.

Some research regarding the lithology correlation using well log instrument with the assistance of modern computer have already been done since the 1970s (Rudman and Lankson, 1973; Mann and Dowell, 1978). Up until now, those researches keep on going as the number of method variety used has been kept on growing as well.

Unfortunately, some of the researches done have not been well applied. Furthermore, the limitation exists within the real-world application, for example, time limitation on data preprocessing. As a result, lithology correlation is still done manually, in which the process is time-consuming. The process itself inclines to limit the ability to utilize all existing data in order to reduce the subsurface uncertainty.

In this study, an artificial intelligence method was used to do well log lithology correlation, which is machine learning. Machine learning itself is a part of artificial intelligence which up until now, some researches using the machine learning method have been significantly increased. The method is fairly applicative, convenient, and is considered to be time-saving.

2. Basic Theory

2.1 Well to Well Correlation

As stated before, the well to well correlation is an integral part of well log analysis in field development. The well to well correlation based on well log instrument is carried out by logging tool itself. A logging tool is put inside the borehole to measure some of the parameters (i.e., electric resistivity, nuclear radiation, acoustic transmission) which these parameters are the reflection of the subsurface lithology. Usually, wells in the same field share a similarity in their well log reading, but it's not always the case. A geology expert would gather these well log data and conduct the well log correlation between wells. It is a strong tool, as geological experts are able to identify subsurface stratigraphic framework. The well log correlation can be time-consuming as well log

data grows massively, and inclines to be subjective, as it is based on one's perspective on the data. For the same reason, researches have been done on automated well correlation as it claims to reduce the subjectivity of conventional methods.

2.2 Machine Learning

In a simple way, machine learning is an art of computer programming so that it can learn from data. As Arthur Samuel (1959) states in a more general definition, "machine learning is the field of study that gives computers the ability to learn without being explicitly programmed". Figure 1 shows problem solving methods using the machine learning approach scheme.

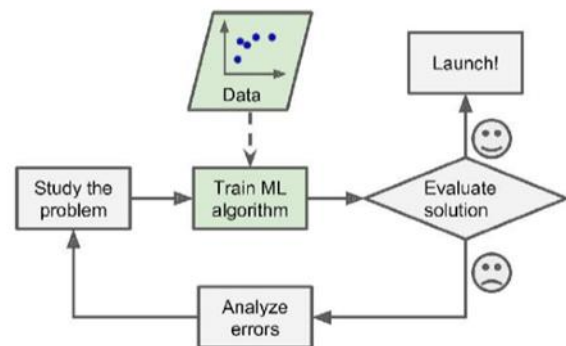


Figure 1. Machine Learning Approach (Géron, A. 2017)

Machine learning shows its strength when faced with dealing problems that either too complex for conventional approach or have no known algorithm. Machine learning systems can be classified according to the amount and type of supervision it gets during training. Four major categories exist are supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning. In this study, a machine learning system used is supervised learning, in supervised learning, the program is trained using labeled data, which means the desired solutions already exist within data training. Some of the existing supervised learning algorithms are k-Nearest neighbors, linear regression, support vector machines, decision trees and random forests.

2.3 K-Nearest Neighbor

The k-Nearest Neighbors is one of the supervised learning algorithms which can be used in regression and classification. The KNN uses the assumption that similar things are near to each other. As KNN uses the idea of similarity, its prediction is based on the closest distance from the training samples. The distance can be calculated in various ways, the most common equation used is the Euclidean distance (or straight-line distance). The Euclidean distance formula can be shown in Eq.1 below.

$$D(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad \dots(1)$$

The number of neighbors, k , is the only parameter and plays an important role in KNN algorithm, to find the optimum k value, one must evaluate the training data. The k with the smallest error between the predicted value and the actual value will be used for testing on unseen dataset. The error can be calculated using the mean squared error formula shown in Eq.2 below.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad \dots(2)$$

The advantages of KNN are:

1. Simple and easy to implement
2. Takes almost no time on data training phase as it uses a training set to make predictions
3. Versatile algorithm, it can be used in classification, regression, and recommender system.

The disadvantages of KNN include:

1. The algorithm gets significantly slower as the volume of data increases.
2. Unable to predict value outside the range of the input data.

2.4 Stochastic Gradient Descent

Gradient Descent refers to an iterative optimization algorithm which is used to find the values of the parameter of a function that minimize the cost function. The gradient itself refers to the slope of a function, the degree of change of a parameter with the amount of change in another parameter. As Gradient Descent is an iterative optimization algorithm, it can be problematic if we face larger data set, as it would be ineffective in the computation process. The Gradient Descent would find the values of the parameter of a function that minimize the cost function as much as possible by computing each data point and iteratively doing so. While the term “Stochastic” itself means “random”, the Stochastic Gradient Descent is the same as previously mentioned Gradient Descent algorithm, but it uses random data points instead of whole data points. Stochastic Gradient Descent (SGD) is a common algorithm used in Machine Learning algorithm forms on the basis of neural networks. In the training process, the algorithm performs a gradient update on each training sample. The algorithm would pass over the training set until the algorithm converges. It is common to use a small number of data points instead of just one random point, this is called mini-batch gradient descent.

The advantages of SGD are:

1. Efficiency on Large Data sets

2. Lots of opportunities in model tuning

The disadvantage of SGD includes:

1. SGD requires a number of hyperparameters such as the number of iterations and the regularization parameter

2.5 Multilayer Perceptron

The Multilayer Perceptron (MLP) is a class of a feedforward artificial neural network. Given a set of input (X) and a target (y), it can learn a non-linear function solver for a classification or regression problem. The MLP is different from logistic regression, in MLP, there can be more than one non-linear layer between the input and the output layer, these are called hidden layers. The input is fed through the input layer and the result can be obtained from output layer, the number of hidden layers can be adjusted accordingly.

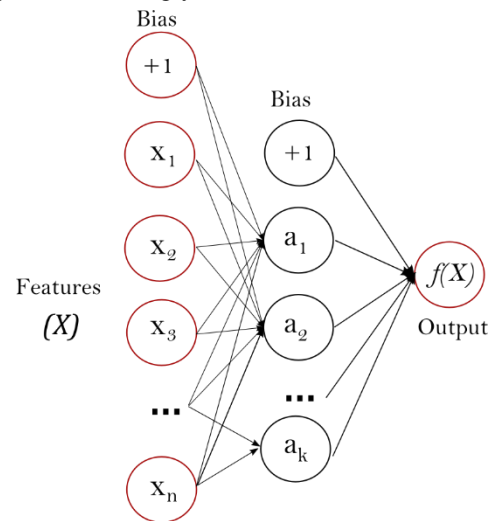


Figure 2. One Hidden Layer MLP (image source- scikit-learn.org)

A layer is a combination of neurons, each neuron in the hidden layer performs a transformation of a linear sum of inputs. The layers of MLP are connected to all the units in the previous layer. In a supervised classification, each input is labeled with the desired solution. As the output layer gives a prediction for each input given, a loss function is defined. A loss function is created to measure the performance of the classifier, higher value of loss function means the predicted value does not correspond to the actual value. Most of the time, an optimization procedure is required in order to avoid overfitting. The set of parameters, or weights, is initialized randomly and refined iteratively to get a lower loss.

The advantages of MLP are:

1. Able to learn non-linear models
2. Able to learn models in real-time using partial fit

The disadvantages of MLP include:

1. Different random weight initializations can lead to different validation accuracy
2. MLP requires tuning a number of hyperparameters such as the number of hidden layers, and iterations.

3. Methodology

Figure 3 shows the detailed workflow of this study, which can be divided into three major stages which are exploratory data analysis, model training & testing, and model evaluation & selection.

3.1 Exploratory Data Analysis

The data were gathered from a logging data, these raw data cannot be directly processed as the input of the machine learning model. At first, data obtained need to be preprocessed, this is to ensure that only good quality data are used in the training of the machine learning model. Before preprocessed, an exploratory data analysis was carried out to understand the data. This is carried out by performing initial investigation of the data with the help of statistics and graphical representations.

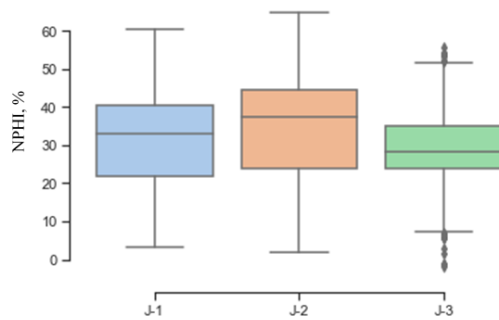


Figure 4. Boxplot of Neutron Porosity Log Data (Before Normalisation)

The data were then preprocessed, which includes removing the missing values, removing data outliers, and removing error data (i.e., an error reading from the logging tool). As the supervised learning is used in this study, the data were then tagged with the correct label, this includes labelling potential zones in a certain depth.

3.2 Model Training & Testing

As the data were labeled and ready to be used, the data were split into training, validation, and testing sets. Before data splitting, the data were scaled to within zero and one range, this is to ensure there will be no different weighting because of different unit/range. The scaled data were then split in 70: 30 ratios, meaning that 70% of the data will be used for model training while the other 30% will be used for model

validation, this is to prevent model overfitting. After the model was trained, model validation was then carried out by comparing the predicted value from the model, and the actual value from the validation data set. The accuracy of the prediction was then calculated.

$$Accuracy = \frac{n_{true\ prediction}}{n_{predicted\ data}} \times 100\% \quad \dots(3)$$

When the accuracy of the model is satisfying enough, the model will be tested with the unseen testing dataset. The accuracy in predicting unseen dataset was then calculated again for further model evaluation.

3.3 Model Evaluation and Selection

The model evaluation was carried out by evaluating the model performance on validation and testing set. The validation set plays an important part in model evaluation, model evaluation should be conducted to increase the performance of the trained model and seek for the model with the best result. The model with the least error and highest accuracy in predicting the validation set should be selected. But one has to make justification and finalize selected model based on the performance on unseen test set. The hyperparameter optimization was done as a model evaluation process. The model was tested with different parameter and then cross validated. Once the model with the best parameter was obtained, the model then tested again on testing set and finalized as the selected model.

4. Case Study

In this study, an actual logging data from a field in Java were used. Five logging data from three different wells (J-1, J-2, and J-3) were gathered and preprocessed. Four zones at certain depth of a well were labeled and used as input and output of the model. Over ten thousand data points were used to train the model. Overall, three models with different approaches were created, and then validated. The models were then tested by predicting unseen data, which in this study, was the logging data from other wells. Hyperparameter optimization was then done as a model evaluation in order to obtain the least error and to seek the best parameter within each model. The model with the best performance was then finalized and justified as the selected model.

5. Result and Discussion

Three models with different classifier were created, each from one well and then validated. A satisfying result was obtained from all three models, validation process resulted in a low error (< 5%). The first model was created using the KNN approach. The KNN model prediction algorithm is highly dependent on the number of neighbors and training data used in the model, it was concluded that higher number of neighbors resulted in the higher error value on validation set. The model was then tested on unseen data, testing on the first testing set resulted in 95.35%

of accuracy, while testing on the second testing set resulted in 87.92% of accuracy. It was also concluded that as the number of neighbors used was increased, the prediction accuracy was improved, up to a certain point where the accuracy eventually becomes stagnant or lower. The model was then evaluated by performing hyperparameter optimization. The grid search algorithm was chosen as the hyperparameter optimization method, and the parameter tuned was the number of neighbors, ranging between 1 – 29, since binary classification was implemented in this study, it is important to use an odd number of neighbors instead of even, this is to prevent tiebreaker during the classification algorithm works. It was obtained that the best number of neighbors to be used was 3, with a cross validation score of 0.981.

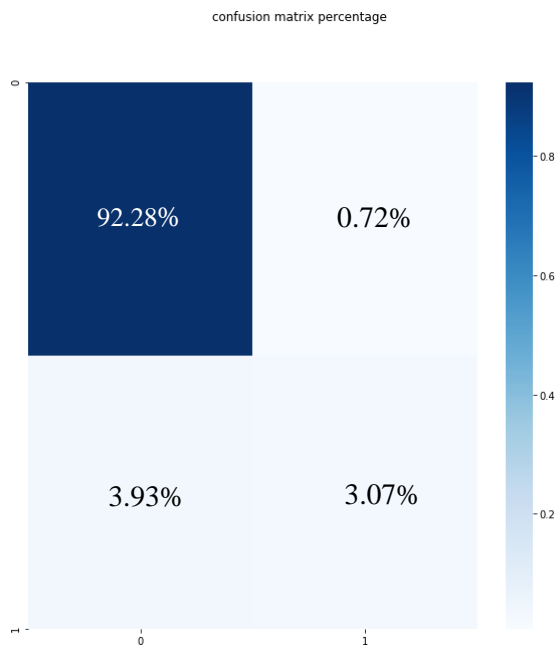


Figure 10. Confusion Matrix Percentage of KNN Model on First Test Set

The second model was created using the linear classifiers with SGD training. The SGD model prediction algorithm is dependent on some of the parameters used in the algorithm such as the learning rate and maximum iteration. The model was then tested on unseen data, testing on the first testing set resulted in 83.44% of accuracy, while testing on the second testing set resulted in 71.05% of accuracy. It was concluded that smaller values of the learning rate improve the accuracy of the model with the cost of computation time, while the higher number of maximum iterations usually improves the accuracy, but is not always the case. To obtain the best parameter to be used, the model was evaluated by performing hyperparameter optimization. The grid search algorithm was chosen as the hyperparameter optimization method, and the parameter tuned was the

value of learning rate and maximum iteration. It was obtained that the best value of learning rate to be used was 0.0001 while the best number of maximum iterations was 500, with a cross validation score of 0.957.

The third model was created using the Multilayer Perceptron classifiers with Stochastic Gradient Descent as an optimization procedure. The MLP model prediction algorithm is dependent on some of the parameters used in the algorithm such as the learning rate, maximum iteration, and the number of hidden neurons. The model was then tested on unseen data, testing on the first testing set resulted in 94.96% of accuracy, while testing on the second testing set resulted in 85.91% of accuracy. As SGD optimization was also used in this model, the model shows similar parameter performance. It was also concluded that smaller values of the learning rate improve the accuracy of the model with the cost of computation time, while the higher number of maximum iterations usually improves the accuracy, but is not always the case. In this case, if the number of maximum iterations used is too low, the model would not be able to reach convergence in the iteration process. To obtain the best parameter to be used, the model was evaluated by performing hyperparameter optimization. The randomized search algorithm was chosen as the hyperparameter optimization method, as using the grid search algorithm would be inefficient. The parameter tuned was the value of learning rate, maximum iteration, and the number neurons. It was obtained that the best value of learning rate to be used was 0.0001 while the minimum number of maximum iterations to reach convergence was 500. As for the number of neurons, the preferable value was 1000 based on randomized search, with a cross validation score of 0.974.

Table 1. Model Performance Comparison

Classifier	Cross Validation Score	Accuracy on 1 st Test Set	Accuracy on 2 nd Test Set
k-Nearest Neighbors	0.981	95.35%	87.92%
Stochastic Gradient Descent	0.957	83.44%	71.05%
Multilayer Perceptron	0.974	94.96%	85.91%

In well to well correlation, the most important thing is the ability to identify labeled zones correctly based on the available data and see the correlation between wells. To understand the performance of each model, confusion matrices and comparison on the actual prediction value were created. Figure 11 to figure 16 present the confusion matrices of all three models, while figure 17 shows the well to well correlation example on J-2 and J-1. It can be seen from table 1 that

testing on the first test set yields to a higher accuracy in all three models compared to the accuracy of testing on a second test set. If we look back on the boxplot of each variable used, it was concluded that the data range for each well is different. The models in this study were trained from log data on J-2 well, it was chosen because of the range data in this well is fairly wider than the other two. Testing on the first test set (J-1) yields a higher accuracy since most of the log data from J-2 have a wider range than J-1. While testing on the second test set (J-3) yields a lower accuracy since two of the log data used (Sonic and Gamma Ray) has wider range than the training data. Table 1 shows the comparison of each model's performance in this study, it can be seen clearly that the first model, KNN, has the highest cross validation score and accuracies on both test sets. Therefore, the KNN model is justified as the selected model.

6. Conclusion

The first model, which was created using the K-Nearest Neighbor approach, was selected and finalized as it has the highest cross validation score and accuracies on both test sets compared to the other two models. With a cross validation score of 0.981, the model resulted in an accuracy of 95.35% on the first test set, and 87.92% of the second testing set. The model was able to be used to identify the correlation between wells.

7. Recommendation

For further evaluation of the selected model, it is recommended to test the model with even more unseen dataset.

Another recommendation is to try different machine learning algorithms other than the algorithms used in this study, each machine learning method has different algorithm, each algorithm has different application and it depends on the data used.

8. Acknowledgement

The author would like to express his gratitude towards The Almighty Allah SWT for the grace and the gift, as the author is able to complete the study. The author would also like to express his gratitude to several people listed below for:

1. The author's family for the endless support and prayer.
2. Dr. Dedy Irawan, S.T, M.T as the author's thesis advisor for continuous support and guidance
3. Lectures, administrative officers and employees in the petroleum engineering department for every teaching and assistance.
4. Ayudhya Sukma Vidyaningtyas, Alysia Chaterine Hinjaya, and other friends in the petroleum engineering study program for always supporting each other.

9. References

- Boggs Jr., Sam, 1995. Principles of Sedimentology and Stratigraphy 2nd ed. Englewood-Cliffs: Prentice-Hall, pp.519-529,561,580-581,613-625,650-666
- Bottou, L, 1991. Stochastic Gradient Learning in Neural Networks. Nimes: EC2., pp.2-4
- Brazell, S., Bayeh, A., Ashby, M. & Burton, D.,2019. A Machine-Learning-Based Approach to Assistive Well-Log Correlation. Petrophysics, 60(4), pp. 469–479. DOI: 10.30632/PJV60N4-2019a1
- Géron, A, 2017. Hands-On Machine Learning with Scikit-Learn and TensorFlow. Sebastopol: O'Reilly Media Inc., pp.21-30,96-100
- Gozzoli, A. (2018, September 5). *Practical Guide to Hyperparameters Optimization for Deep Learning Models*. FloydHub. <https://blog.floydhub.com/guide-to-hyperparameters-search-for-deep-learning-models/>. Accessed June 12, 2020.
- Kingma, D.P., and Ba, D.L.,2015. ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION. 3rd International Conference for Learning Representations.
- Mann, C.J., and Dowell, T.P., Jr., 1978. Quantitative Lithostratigraphic Correlation of Subsurface Sequences. Computers & Geosciences, 4(3), pp. 295–306.
- Pant, A. (2019, January 11). *Workflow of a Machine Learning project*. Towards data science. <https://towardsdatascience.com/workflow-of-a-machine-learning-project-ec1dba419b94>. Accessed June 12, 2020.
- Stalfort, J. (2019, June 6). *Hyperparameter tuning using Grid Search and Random Search: A Conceptual Guide*. Medium. <https://medium.com/@jackstalfort/hyperparameter-tuning-using-grid-search-and-random-search-f8750a464b35>. Accessed June 12, 2020.
- Rudman, A.J., and Lankston, R.W., 1973. Stratigraphic Correlation of Well Logs by Computer Techniques. AAPG Bulletin, 57(3), pp.577–588.
- Zimmerman, T., Liang, L., & Zeroug, S.,2018. Machine-Learning-Based Automatic Well-Log Depth Matching. Petrophysics, 59(6), pp. 863–872. DOI: 10.30632/PJV59N6-2018a10

List of Figures

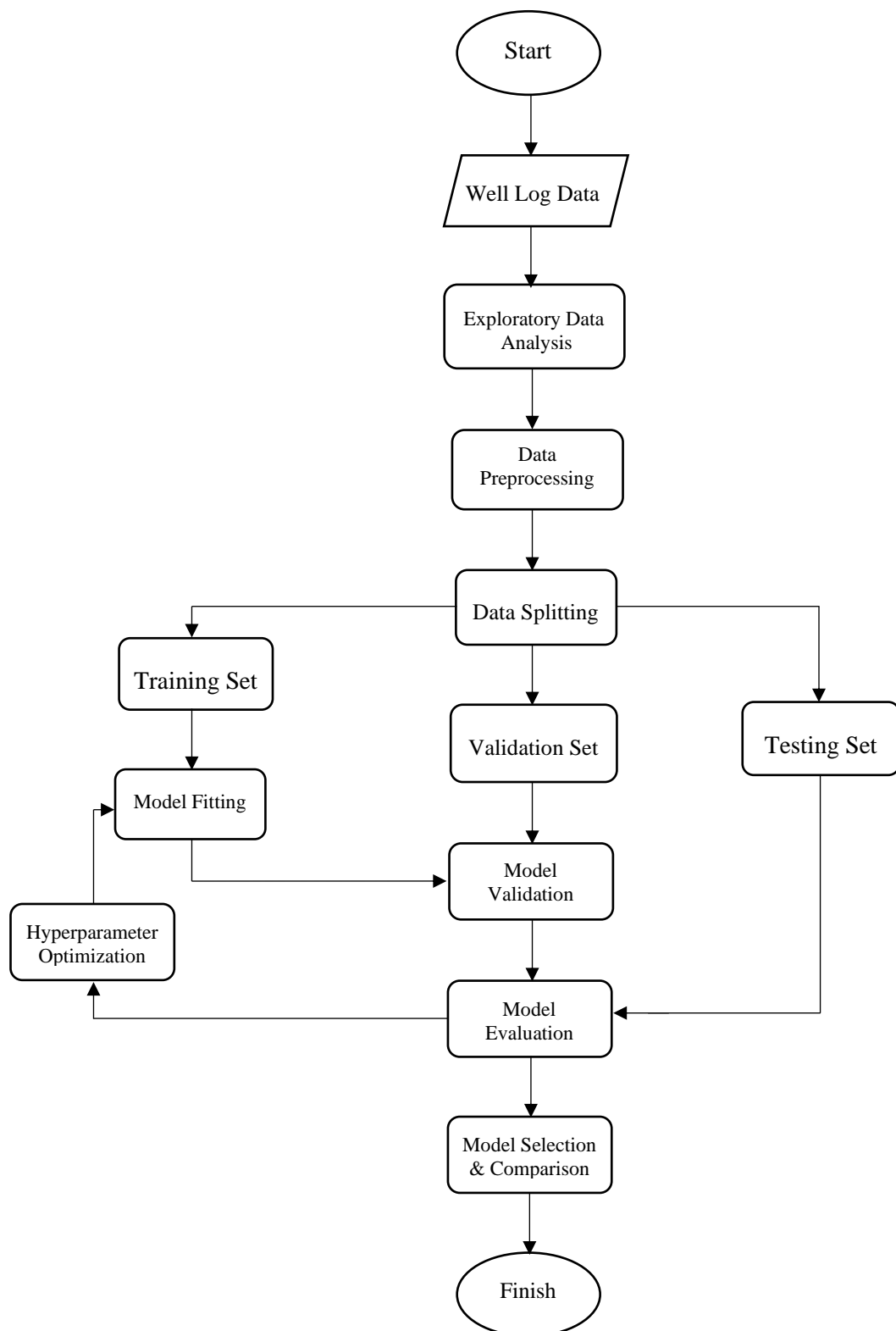


Figure 2. Study Workflow

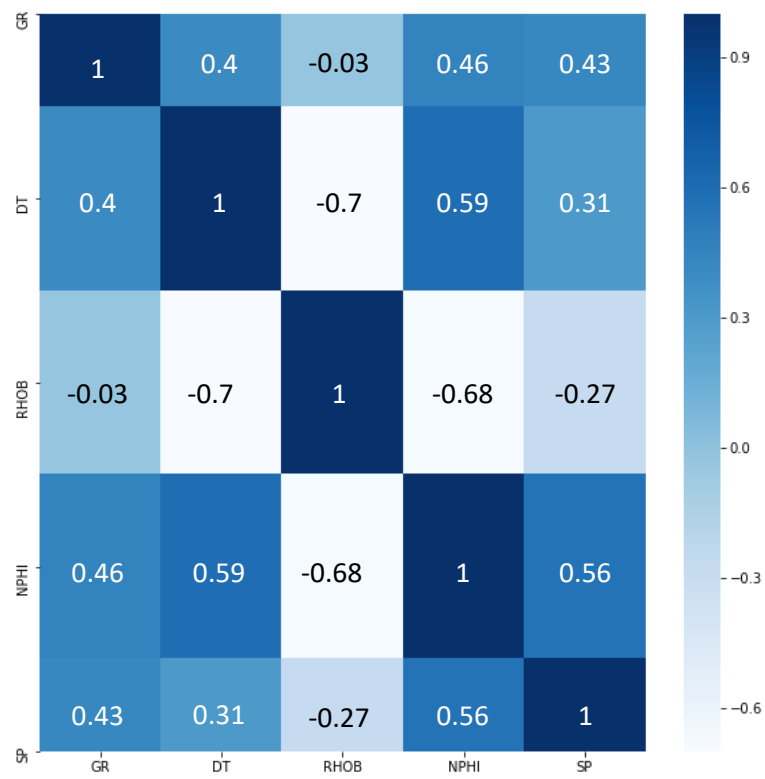


Figure 5. Correlation Matrix of The Log Data Used

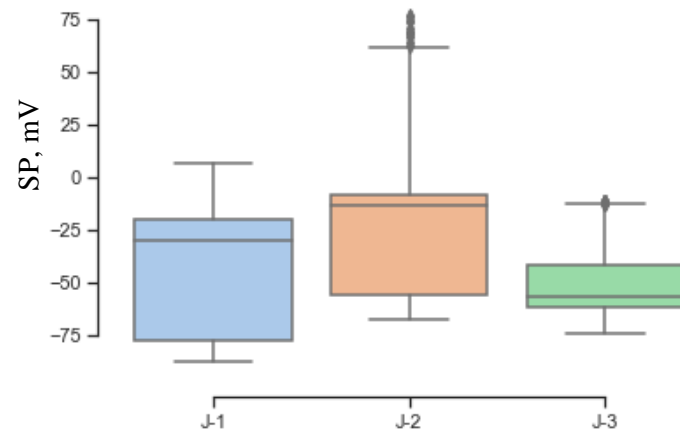


Figure 6. Boxplot of Spontaneous Potential Log Data (Before Normalisation)

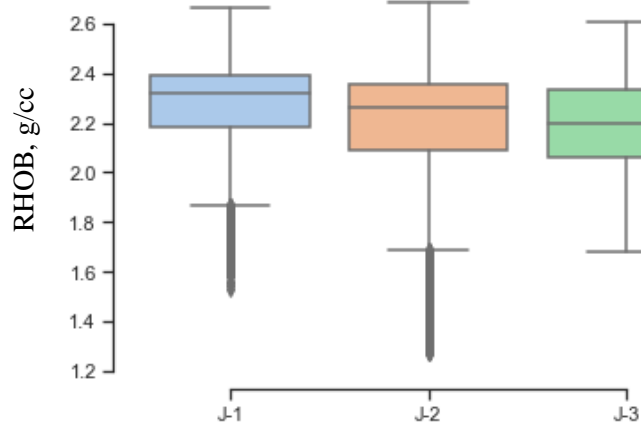


Figure 7. Boxplot of Bulk Density Log Data (Before Normalisation)

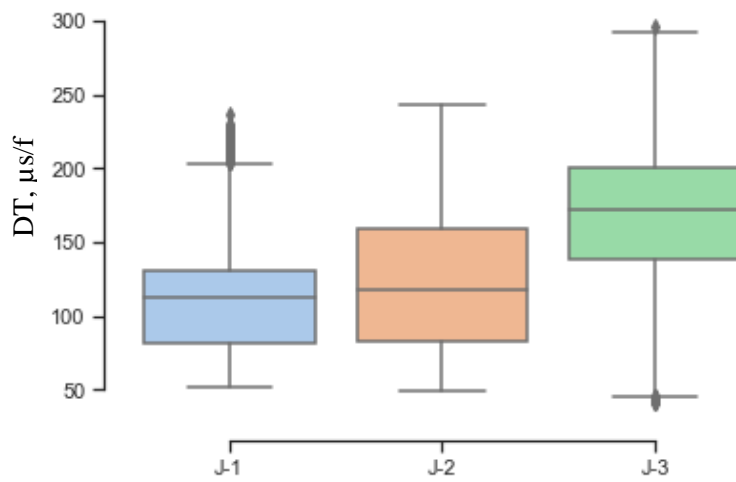


Figure 8. Boxplot of Sonic Log Data (Before Normalisation)

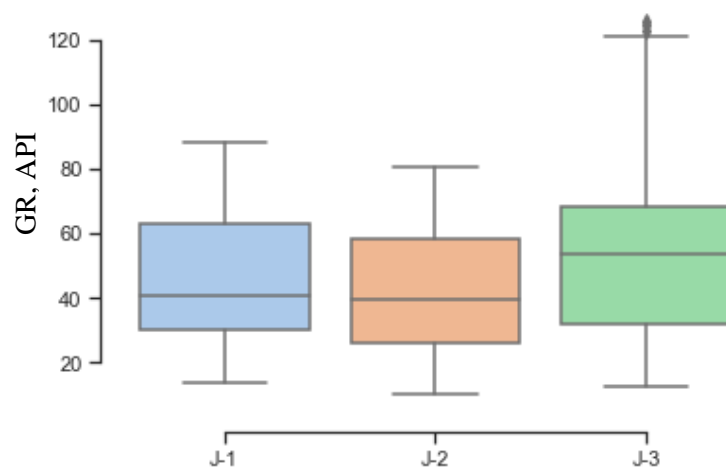


Figure 9. Boxplot of Gamma ray Log Data (Before Normalisation)

Actual Prediction Comparison on Test Set

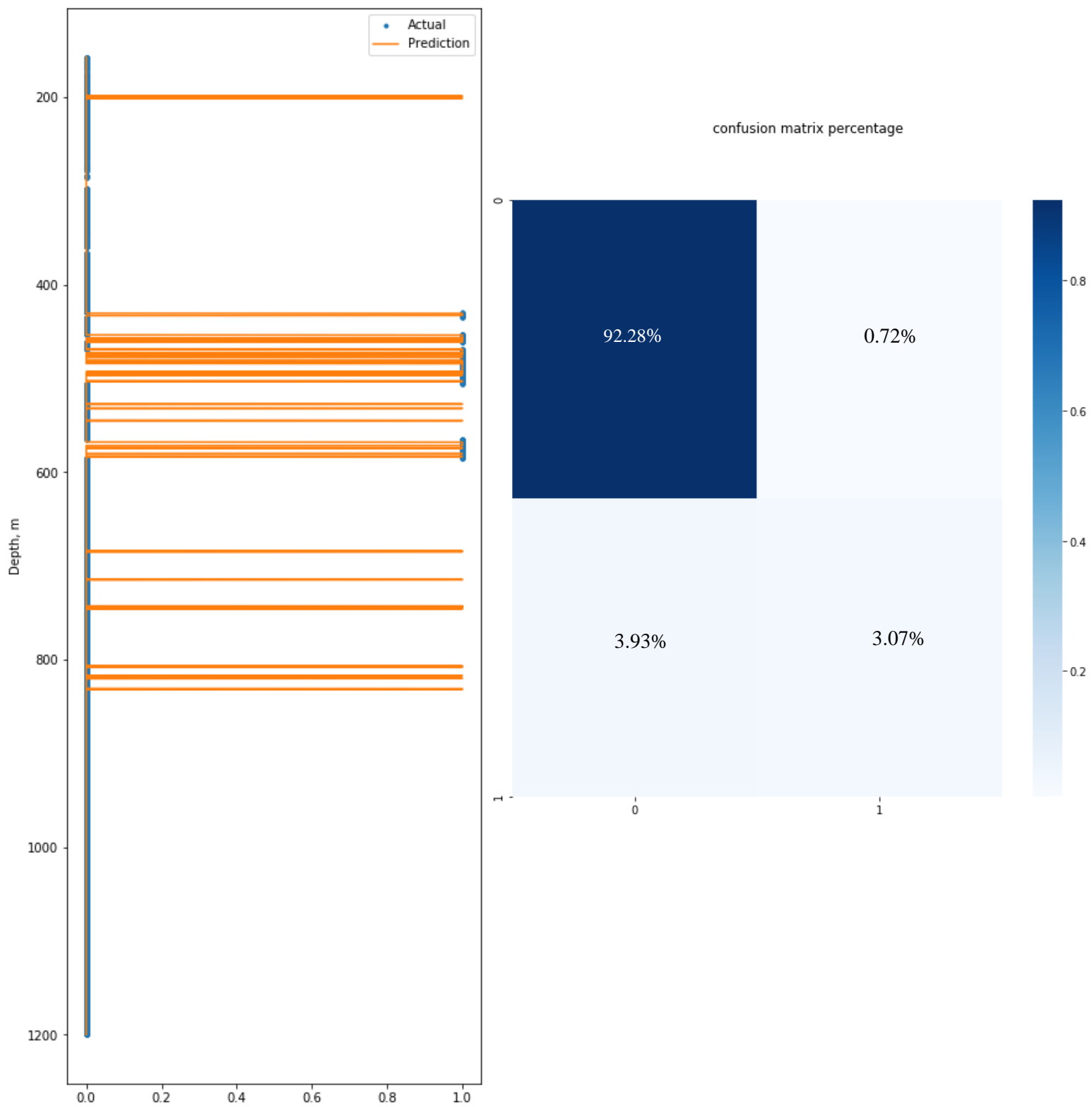


Figure 11. KNN Model Performance on First Test Set

Actual Prediction Comparison on Test Set

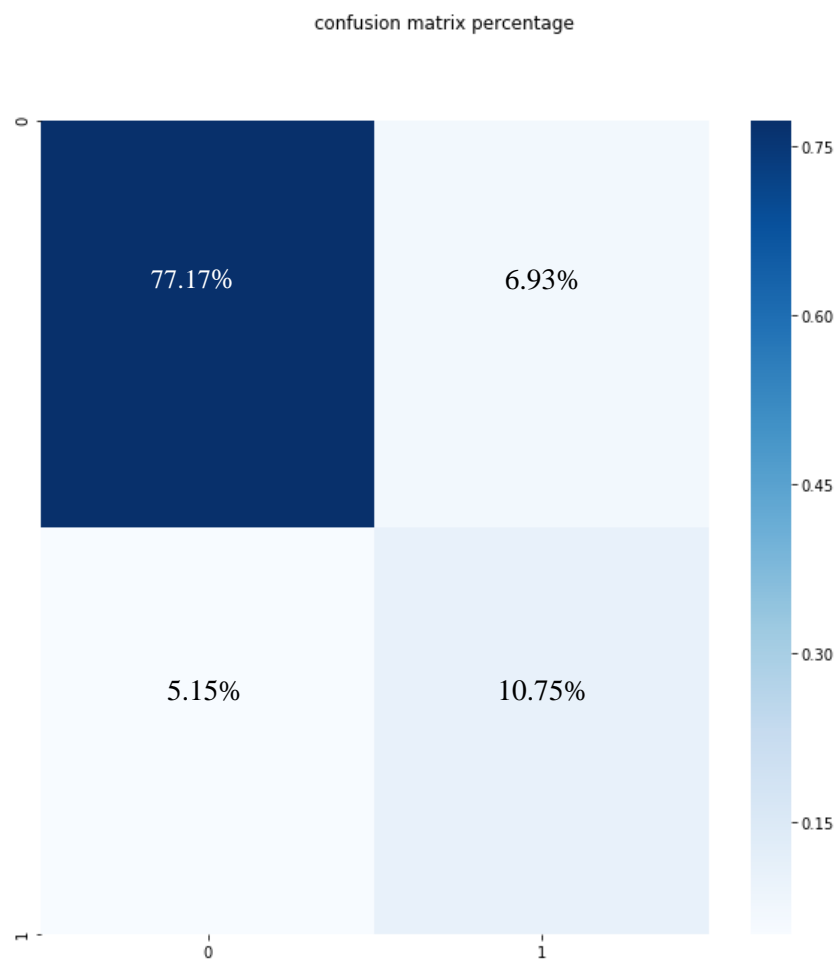
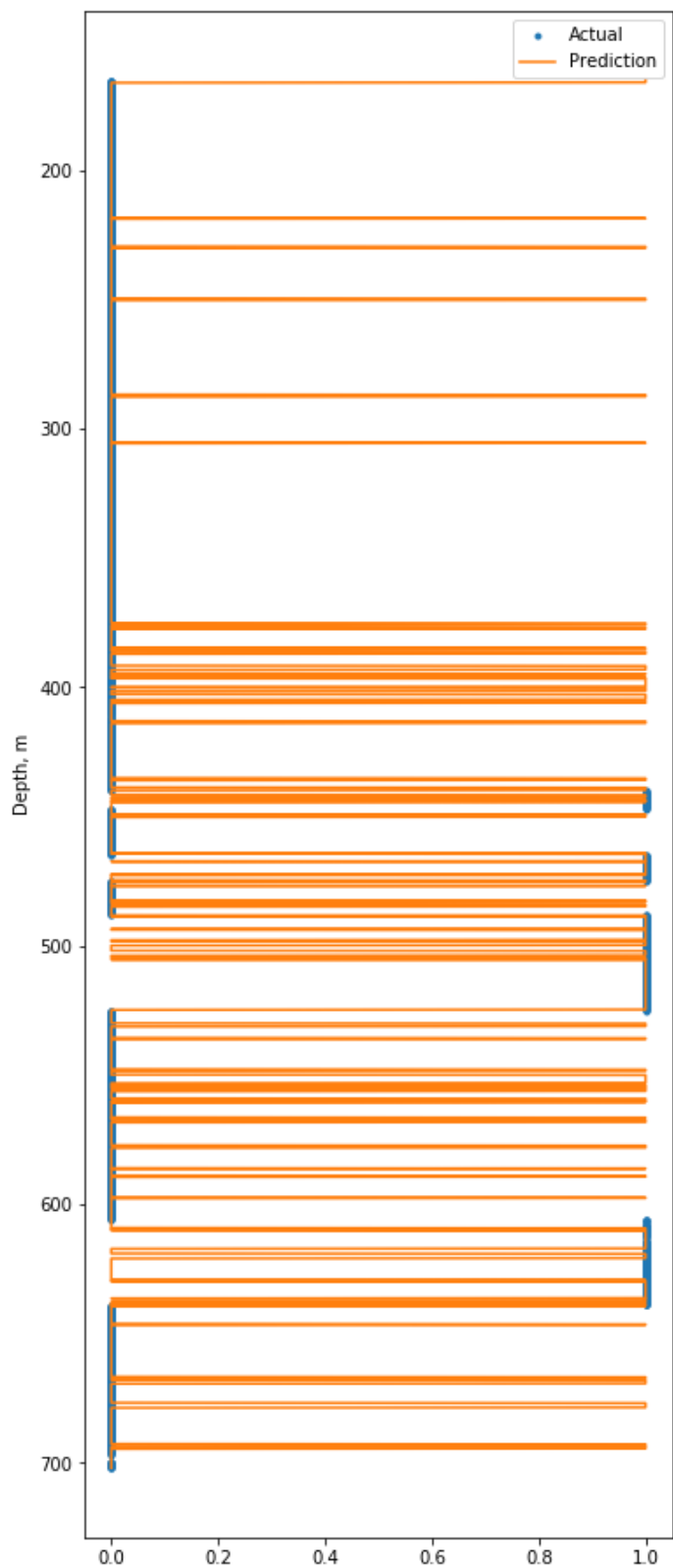


Figure 12. KNN Model Performance on Second Test Set

Actual Prediction Comparison on Test Set

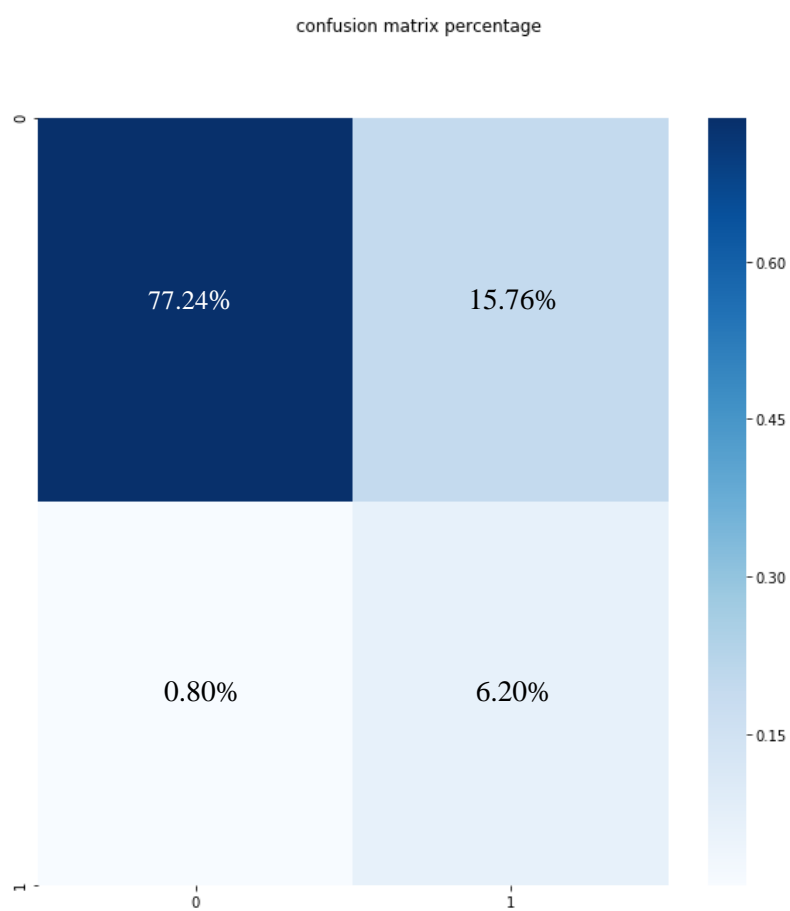
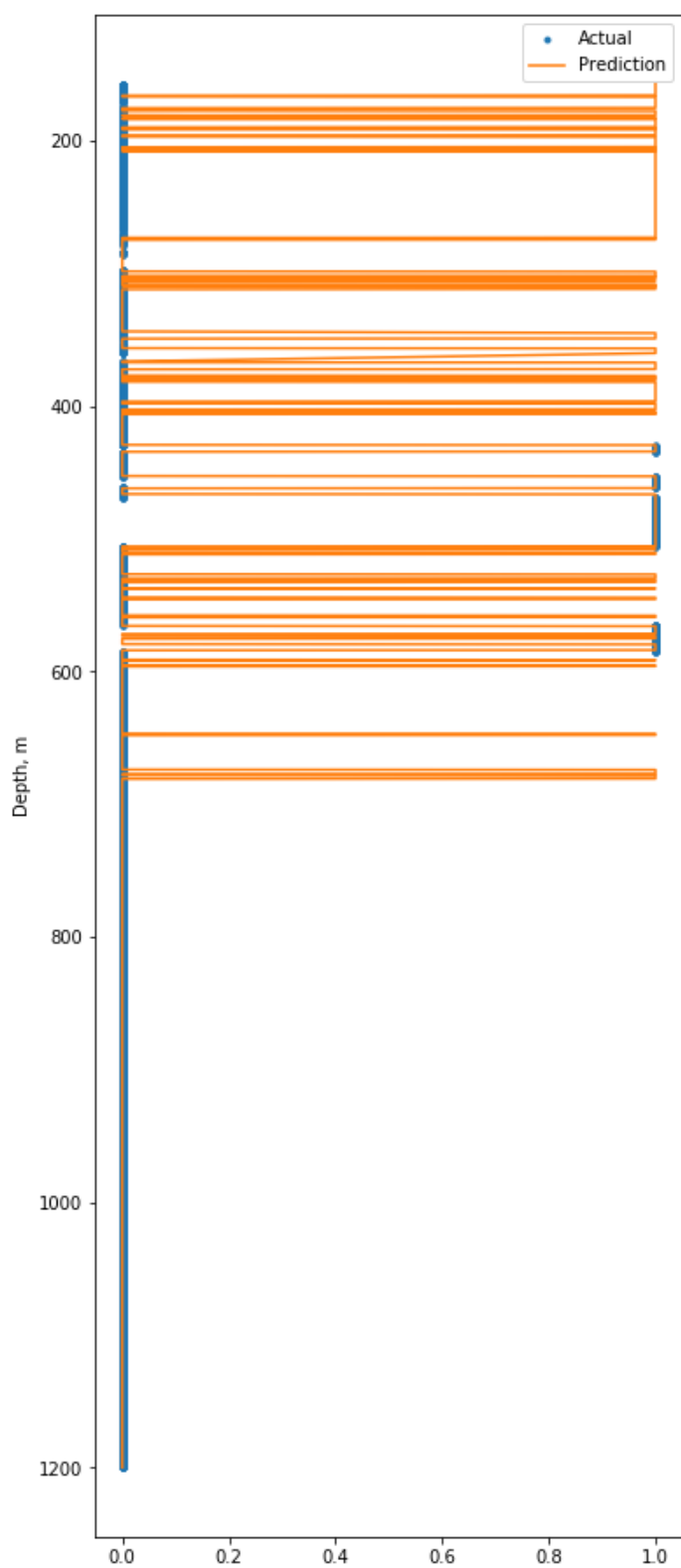


Figure 13. SGD Model Performance on First Test Set

Actual Prediction Comparison on Test Set

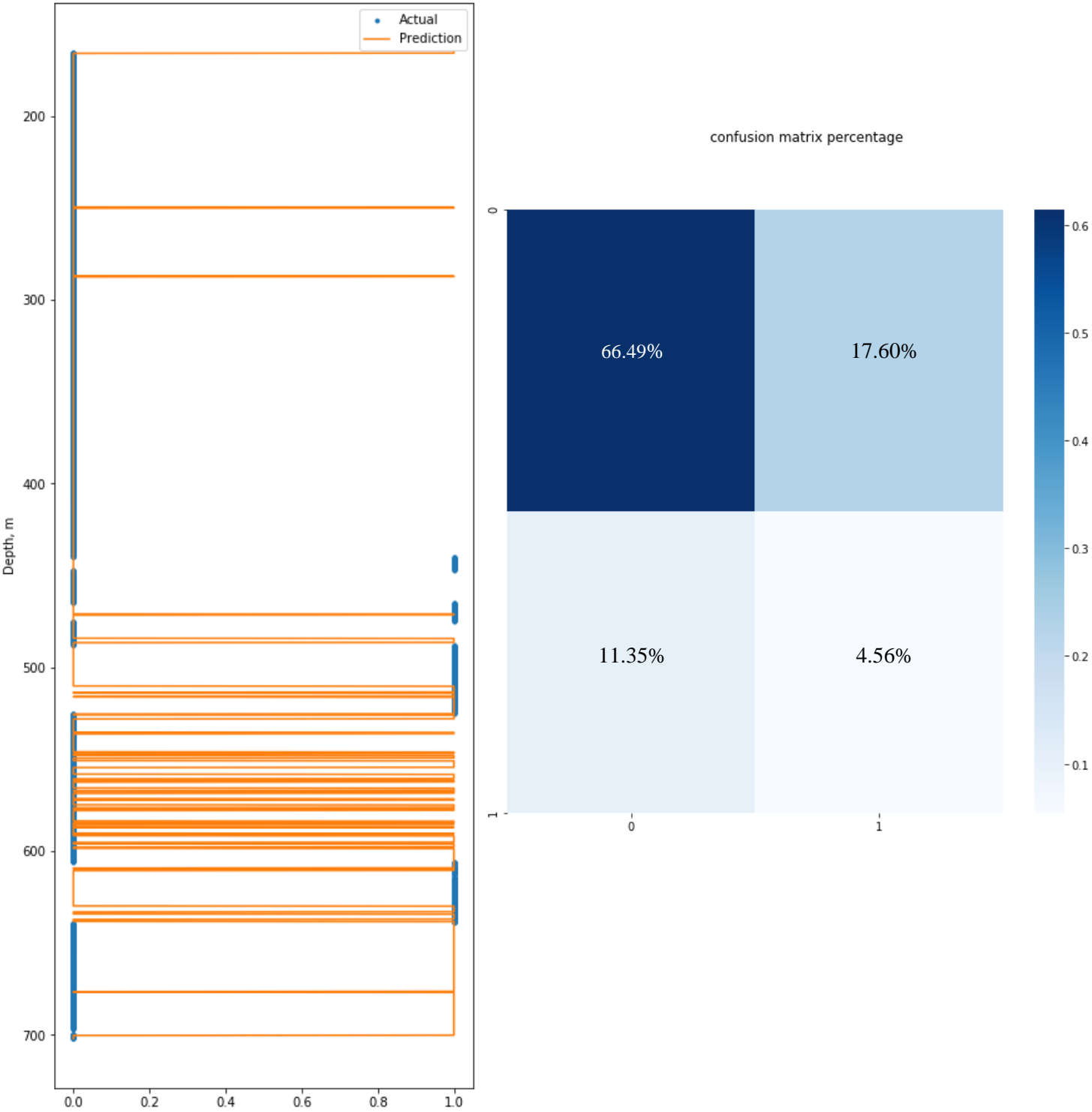


Figure 14. SGD Model Performance on Second Test Set

Actual Prediction Comparison on Test Set

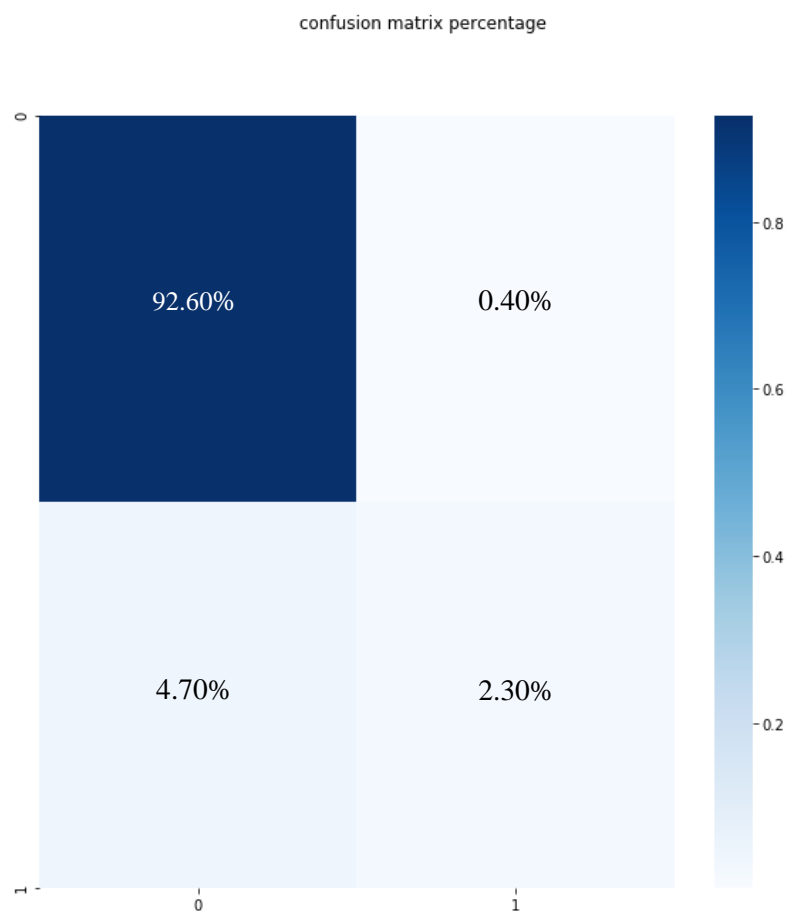
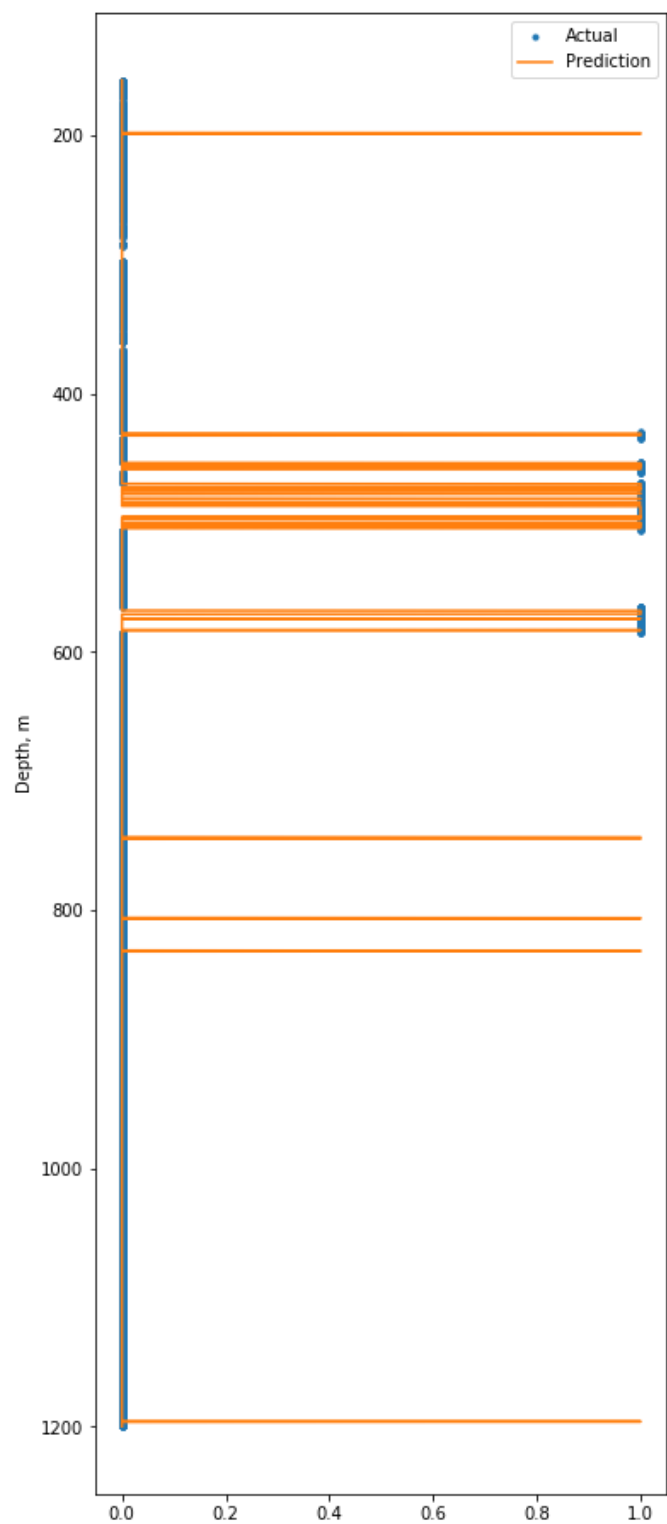


Figure 15. MLP Model Performance on First Test Set

Actual Prediction Comparison on Test Set

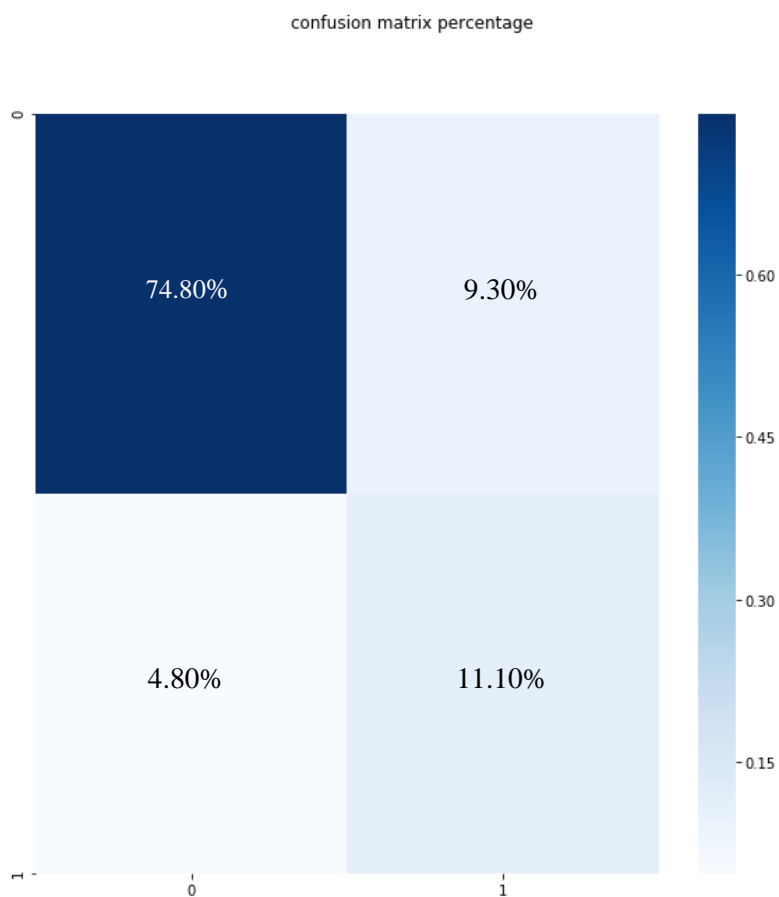
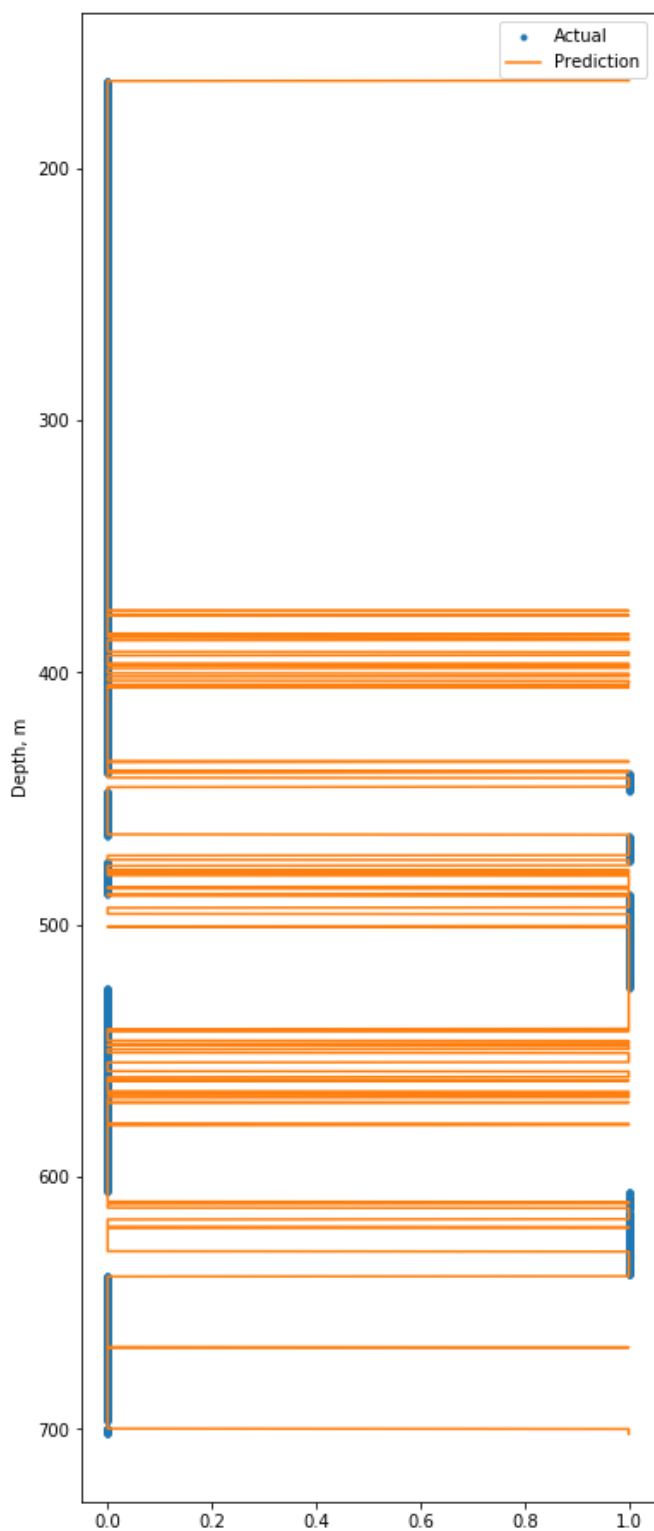


Figure 16. MLP Model Performance on Second Test Set

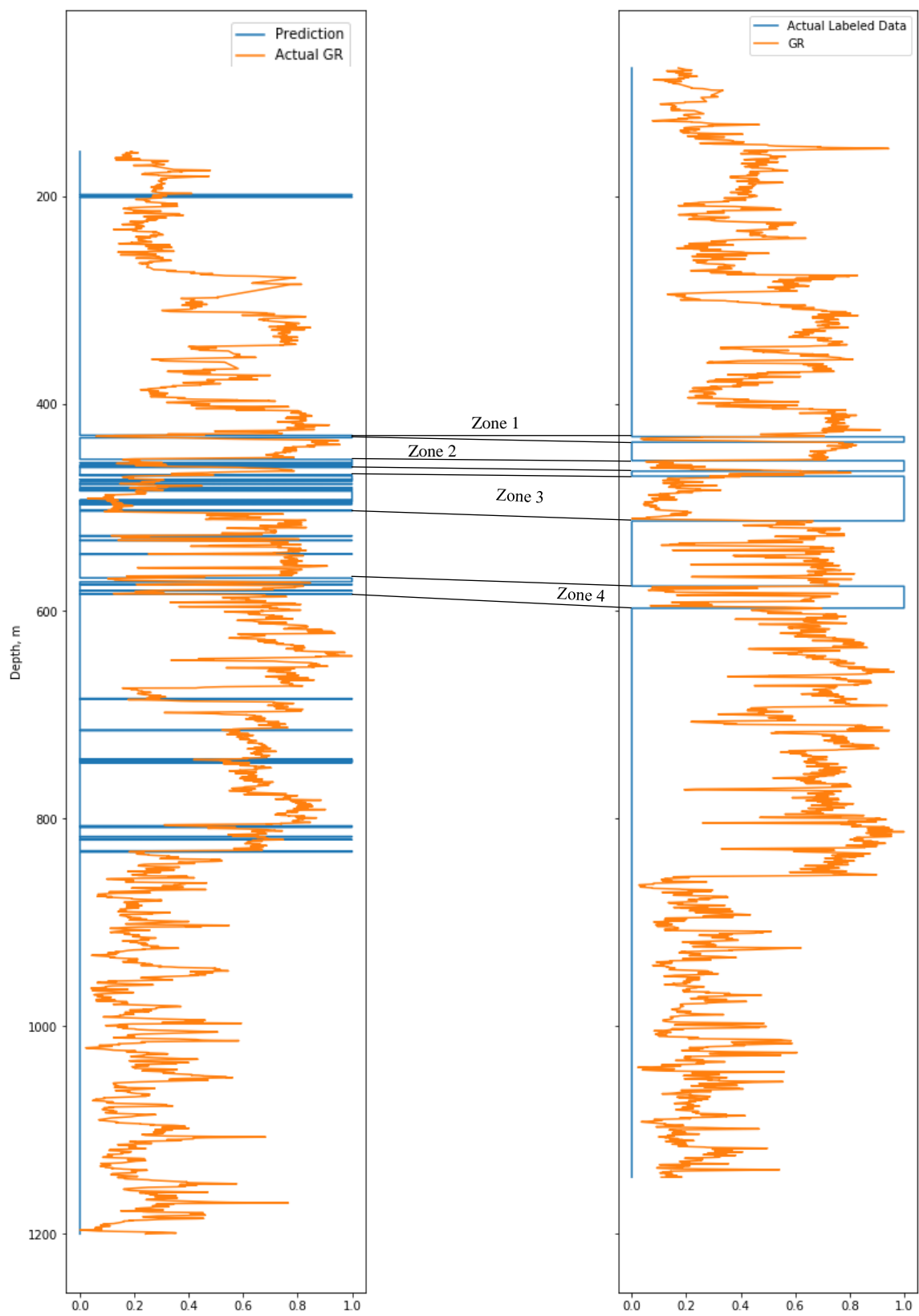


Figure 17. Well to Well Correlation Example on J-2 (Right) and J-1 (Left)