Quantum and Classical Machine Learning Algorithm Comparisons for Monitoring PV Array Faults with Emphasis to Shading Detection

Kaden McGuffie
SenSIP IRES UCy
Arizona State University
Tempe, Arizona
ktmcguff@asu.edu

Glen Uehara
SenSIP Researcher
Arizona State University
Tempe, Arizona
guehara@asu.edu

Sameeksha Katoch
SenSIP Researcher
Arizona State University
Tempe, Arizona
skatoch1@asu.edu

Andreas Spanias
SenSIP Director
Arizona State University
Tempe, Arizona
spanias@asu.edu

***Abstract*—Solar panels are becoming more common for everyday energy production, however there are drawbacks that inhibit them from reaching their optimal potential. One major drawback comes from a variety of faults within utility scale solar panel arrays. Using machine learning algorithms, engineers are able to drastically reduce the time for detecting and repairing solar panels faults. The focus of this project is to use machine learning to detect shading effects on utility scale PV arrays. The first phase of this SenSIP-KIOS collaborative IRES project included training at ASU on a variety of machine learning algorithms and software. In this phase of the project, it became evident that there are challenges in using the machine learning (ML) models for detecting shading. I began my studies on ML by considering three classical models, namely, used: Logistic Regression (LR), Support Vector Machine (SVM), and Neural Networks (NN). The three methods were compared using solar fault data and results were produced in terms of prediction accuracy. I then explored quantum machine learning models and trained and tested the quantum model. The accuracy of the classical and quantum models are compared to see which model is optimal for classifying shaded solar array faults.**

***Index Terms*—machine learning, quantum, neural networks, solar panels, optimization,**

I. Introduction

The transition to renewable energy is imperative to our society's future. Solar energy holds a promising future in renewable energy sector, but still has ways to go before taking over our current, harmful energy sources. In order to fast-track the transition, solar energy needs to prove itself as a more viable energy source. There are many aspects to look at when trying to reach the maximum energy output. This research project does not look at the inner workings and structure of the solar panels, but instead looks at the type of faults that occur in real time which prevent the solar panels from reaching their maximum energy potential.

To detect faults, smart monitoring devices (SMD) periodically send data of each solar panel's voltage, current, irradiance, and temperature. With this data, machine learning algorithms are trained to predict the type of fault, and then tested to see the accuracy of the model. There are three subsections of machine learning: supervised learning, unsupervised learning, and reinforcement learning. This project uses supervised learning. When the training data points are labeled as their correct class, supervised learning is used to train models to classify the test data. The data provided for this project is labeled with its corresponding features, hence supervised learning is used [1]. Within the code, there are hyperparameters used that can alter the efficiency of the system. The type of solver, type of kernel, number of hidden layers, and number of nodes are examples of hyperparameters. Different models of machine learning are presented resulting in different hyperparameters being used. Playing with these hyperparameters allows models to minimize their bias and variance. Having low bias underfits the data, which makes the data too generalized. Having high variance overfits the data, which means there is too much noise the algorithm uses for fitting the data [2].

There are five classifications of faults looked at within the SenSIP IRES: soiled, degraded, shaded, short circuited, and standard testing conditions (STC). Each fault holds different values in terms of voltage, current, irradiance, and temperature. For example, if the irradiance and temperature are near STC but the current is very low, it is highly likely that the fault is a short circuit [1]. This report holds emphasis toward the shaded fault.

Fig. 1. Shaded Solar Panel

.

For the first part of this project, three classical models are used to classify shaded faults: LR, SVM, and NN. The three are then compared to see which classical model is best suited for shaded fault detection.

For the second part of the project, a quantum support vector machine (QSVM) and quantum neural network (QNN) are trained and tested to classify the shaded fault. This is done with the Qiskit library. Qiskit allows users to run quantum simulations without the need for expensive quantum hardware. These results are compared with their classical model counterpart.

To show the results of each model, a confusion matrix is generated. A confusion matrix shows how many times a class was predicted by the model and if the class prediction is true or false. The confusion matrix gives the reader an understanding on how accurate the model is. It is also useful for its abilities to show where the model is strong or weak. These can t seen with four categories: true positives, false positives, tru negatives, false negatives. In terms of shaded fault detection, true positive occurs when the model correctly predicts a shade fault. A false positive occurs when the model predicts th there is a shaded fault when in actuality there is no fault. A tru negative is when the model correctly predicts there is no faul and a false negative is when there is a shaded fault and th model predicted that there is no fault [3]. Fig. 2 is an examp of a confusion matrix from the Machine Learning for Sol Array Monitoring, Optimization, and Control publication.

		Actual Class		
		Positive	Negative	
Predicted Class	Positive	65	38	103
	Negative	74	123	197
		139	161	

Fig. 2. Confusion Matrix Example

.

Quantum computing offers superior capabilities over classical computing. They hold much greater storage, and easily process and store data. Quantum computing is great for optimization problems, making it attractive to use for this project. However, this study does not use a massive database, which is better suited for quantum. Still, quantum algorithms have outperformed classical models in many areas [4]. This project will see how quantum and classical compare when classifying fault detection.

Currently, quantum systems have work to be done in order to maximize their proficiency in prediction models. Still, this research gives insight to how the two methods compare as of present day technology.

II. Classical Models

All of the classical models used 70% of the data for training and the remainder for testing.

A. Logistic Regression

The logistic regression machine learning algorithm is the first method tested for fault detection classification. This model uses the sigmoid function (1) for binary classification. The purpose of the function is to create a threshold that if crossed can determine that the data now in a certain class [5].

$$f(x) = \frac{1}{1 + e^{-\theta x}} \tag{1}$$

The Cyprus IRES team optimized hyperparameters for a specific fault and found the probability for correctly classifying their chosen fault. In this report, the shaded class was optimized and found unimpressive results when compared to the other optimized classes from the IRES team.

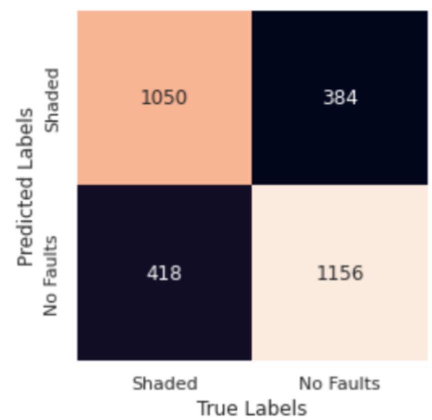

Fig. 3. Confusion Matrix for Shaded Logistic Regression

.

The average accuracy of the model is 76%. The hyperparameters barely changed the accuracy of the results. Still, the gradient descent solver 'lbfgs' produced the best results by two tenths of a percent. The solver took around a hundred iterations to converge in its gradient descent, and did not need any penalty applied. Penalties change the coefficients of the features, which can help optimize the results.

B. Support Vector Machine

The second model developed and optimized was the support vector machine. Essentially, a SVM creates a line, plane, or hyperplane to seperate the data into their classes [6]. The model performed optimally with the 'rbf' kernel, and on average, correctly predicted the shaded class 75% of the time. This is slightly lower than the LR model.

.

The confusion matrix shows a higher amount of false negatives than false positives. This is also the case in the first model. Most likely, the Shaded features can be indistinguishable from the STC in certain instances, making it difficult for the model to classify the two.

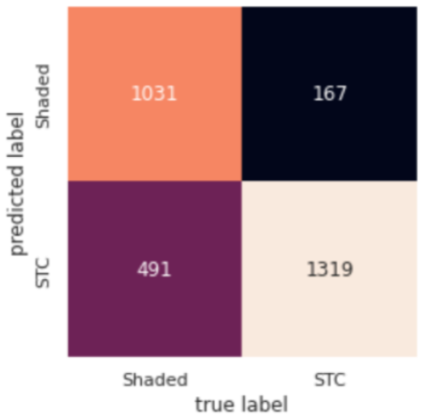

Fig. 4. Confusion Matrix for Shaded Support Vector Machine

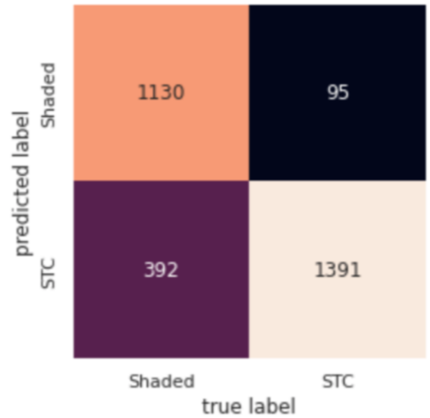

Fig. 6. Confusion Matrix for Shaded Neural Network

C. Neural Network

Neural networks are generally more complex compared to the LR and SVM models. A NN has hidden layers with a specified amount of nodes that will activate when certain conditions are met. The NN goes through a multitude of forward and backwards propagation to readjust the weights and bias of the nodes to optimize the classification [7]. Fig. shows an example of what the inner workings of a multi class neural network fault classifier looks like.

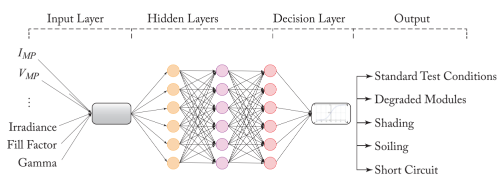

Fig. 5. Neural Network Architecture used for Fault Detection and Classi cation

.

The Neural Network model takes more time to train du to the multitude of hyperparameters with different values t try. The hyperparameteres used to train the model in this study are the activation function, solver, number of hidden layers, and number of nodes [8]. After many trials, the optimal hyperparameters are the 'ReLU' activation function, the 'adam' solver, and two hidden layers with 150 nodes each. The model performed at an average accuracy of 83%, the highest of the three classical models. The NN model has more false negatives compared to false positives.

.

III. Quantum Models

A. Quantum Support Vector Machine

The quantum support vector machine is created with a quantum circuit and simulated with Qiskit libraries. The QSVM model has different hyperparameters such as the number of shots, backend simulator, and entanglement. The QSVM outperformed its classical counterpart by 2%

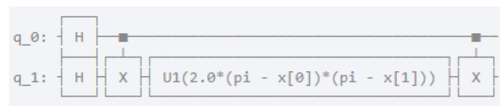

Fig. 7. Quantum Support Vector Machine Circuit

.

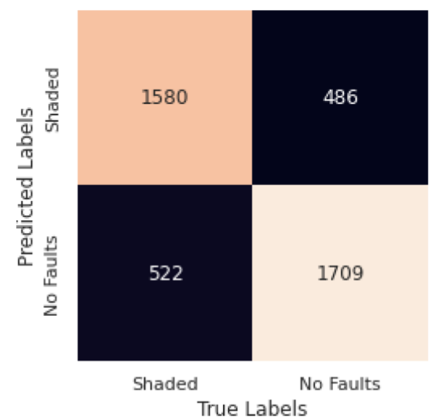

Fig. 8. Confusion Matrix for Shaded Quantum Support Vector Machine

.

B. Quantum Neural Network

The QNN works by converting classical nodes into quantum nodes, which can be used with Qiskit libraries [9]. After extensive work with the code, the algorithm still performed inconsistently. The QNN still will be worked on in the future as it is likely the model could perform very effectively if developed correctly.

.

IV. Conclusion

There are four key takeaways from this study. Firstly, every model developed had a greater number of false negatives

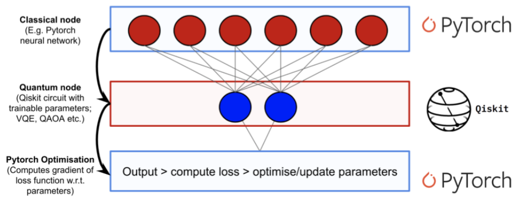

Fig. 9. Confusion Matrix for Shaded Quantum Support Vector Machine

compared to false positves. This shows that machine learning algorithms has a harder time detecting a shaded fault because the features can be very similar to STC. Secondly, the classical neural network model performed the best out of all the working models. Thirdly, the quantum support vector machine model was able to outperform its classical counterpart, showing there can be an advantage using quantum for this classification problem. Lastly, if the QNN were to work, it can be inferred that the QNN can outperform every model. Seeing that the classical NN model performed the best and that the QSVM boosted the accuracy by 2% compared to its classical counterpart, the QNN has the potential to be the best suited model for this problem.

V. Acknowledgements

This work was supported in part by NSF Grant No. 1854273. Special thanks to Dr. Andreas Spanias, Glen Uehara, and Sameeksha Katoch for their instruction and guidance throughout this project.

VI. References

[1] Rao, S., Katoch, S., Narayanaswamy, V., Muniraju, G., Tepedelenlioglu, C., Spanias, A., ... Srinivasan, D. (2020). Machine learning for solar array monitoring, optimization, and control. Synthesis Lectures on Power Electronics, 7(1), 1-91.

[2] Mehta, P., Bukov, M., Wang, C. H., Day, A. G., Richardson, C., Fisher, C. K., Schwab, D. J. (2019). A high-bias, low-variance introduction to machine learning for physicists. Physics reports, 810, 1-124.

[3] Beauxis-Aussalet, E., Hardman, L. (2014). Visualization of confusion matrix for non-expert users. In IEEE Conference on Visual Analytics Science and Technology (VAST)-Poster Proceedings.

[4] Divya, R., Peter, J. D. (2021, November). Quantum Machine Learning: A comprehensive review on optimization of machine learning algorithms. In 2021 Fourth International Conference on Microelectronics, Signals Systems (ICMSS) (pp. 1-6). IEEE.

[5] Dreiseitl, S., Ohno-Machado, L. (2002). Logistic regression and artificial neural network classification models: a methodology review. Journal of biomedical informatics, 35(5-6), 352-359.

[6] Noble, W. S. (2006). What is a support vector machine?. Nature biotechnology, 24(12), 1565-1567.

[7] Cilimkovic, M. (2015). Neural networks and back propagation algorithm. Institute of Technology Blanchardstown, Blanchardstown Road North Dublin, 15(1).

[8] Smith, L. N. (2018). A disciplined approach to neural network hyper-parameters: Part 1–learning rate, batch size, momentum, and weight decay. arXiv preprint arXiv:1803.09820.

[9] The Qiskit Team. “Hybrid Quantum-Classical Neural Networks with Pytorch and Qiskit.” Qiskit.org, Data 100 at UC Berkeley, 6 July 2022, https://qiskit.org/textbook/ch-machine-learning/machine-learning-qiskit-pytorch.html.