# Building machine learning models to predict solar power generation based on location

Laila Shakoor
Mentored by Mr. Jarick Cammarato

## Introduction

Solar power is one of the most promising solutions to climate change. However, solar power generation is often variable and unreliable as it is susceptible to weather conditions. Being able to predict solar power generation would increase confidence in the capability of solar power and allow it to be introduced into more communities.

Both weather conditions, such as solar irradiance, cloud cover, and temperatures, and societal/human factors such as operation, maintenance, and access to materials, affect solar power generation (Sharma et al., 2011). Instead of attempting to consider all these factors, this project aimed to build machine-learning models to predict solar power generation based solely on location.

Linear regression models and artificial neural networks (ANNs) were built to determine which would be more accurate. Linear regression models detect linear relationships from data and use them to predict unknown data, while ANNs can also work with non-linear relationships (Kumar & Kalavathi, 2011). The features, or input, of the models, were latitude, longitude, and the installed capacity of a hypothetical solar plant. The label, or output, was the yearly solar power generation in megawatt-hours (MWh).

## Methods and Materials

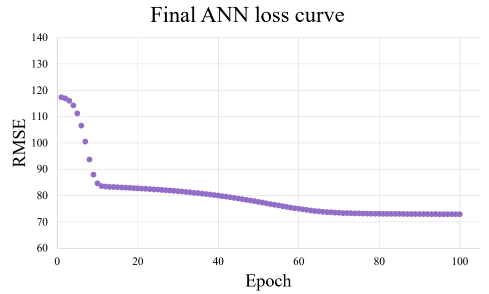
The dataset used to build the machine learning models was from the National Renewable Energy Laboratory (NREL). It gave the latitude, longitude, installed capacity, and a year's worth of solar power generation values for about 6,000 hypothetical solar plants, each in a separate Excel file. The features, latitude, longitude, and installed capacity, were in the names of the files downloaded from the NREL. These were extracted using a VBA script that parsed each file name for the features. Each file contained a year's worth of generation values for a specific location in five-minute intervals. To extract the label, the yearly outputs, a Python script was written to iterate through each location's file and find the sum of all generation values, yielding the total generation for the year. Then, the features and labels were combined into a Pandas DataFrame, allowing it to be used when training the model.

The dataset was shuffled and then split into training data (80%) and testing data (20%). Outliers were determined and removed from the dataset using a Python script. A data point was considered an outlier if its *z*-score was above three. The linear regression models was built in Google Colab, using the Python language and the TensorFlow library. Numerous combinations of hyperparameters such as batch size, number of epochs, and learning rate were tested, along with multiple different feature representations such as bucketized features and feature crosses.

## Methods and Materials (continued)

This was done until the model had achieved its lowest possible root mean squared error (RMSE).

An ANN was built using the same language and library. Various hyperparameters were tested, along with the number of hidden layers and neurons involved in training. Once the ANN had achieved it's lowest possible RMSE, as shown in Graph 1, it was determined that the ANN was more accurate than the linear regression model.

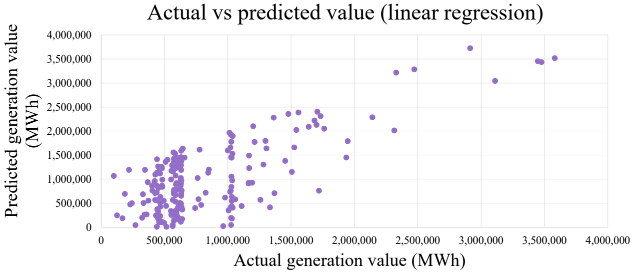


Graph 1(left): A curve showing the RMSE over the training time of the final ANN.

## Results

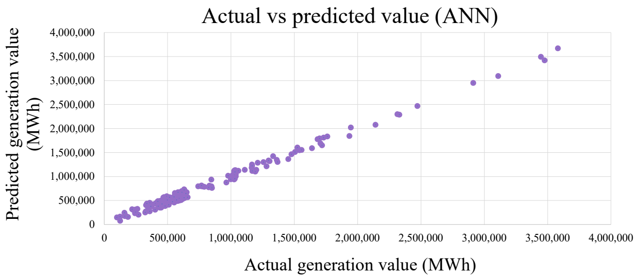
| Model Type | Linear Regression | ANN |
|---|---|---|
| RMSE | 5,742.00 | 77.20 |

Table 1 (above): The RMSE of the optimal version of each model type after tuning of features and hyperparameters.



Graph 2 (above): The true generation values and the values predicted by the linear regression model.

## Results (continued)



Graph 3 (above): The true generation values and the values predicted by the ANN.

It was determined that the ANN, shown in Graph 3, was more accurate, with an RMSE of 77.20 compared to 5,742.00, the RMSE of the linear regression model, shown in Graph 2. Each model was trained multiple times with various feature representations and hyperparameters, so the RMSE values shown in Table 1 are the lowest values obtained throughout the training. The ANN was then tested using observed data from the Global Power Plant Database, as opposed to the hypothesized data used during training. The resulting RMSE was 261.36, so the ANN was not as accurate when tested on a different dataset.

## Conclusion

The purpose of the project was met. It was determined that the ANN outperformed the linear regression model with an RMSE of 77.20. This implies that the relationship between location and solar power generation is not linear. This project has significant potential applications. When solar power consumption exceeds generation, supplemental energy, often from nonrenewable sources, is required. However, if solar power generation can be predicted, consumption can be tailored to this prediction, decreasing the need for supplemental energy. Additionally, government organizations could use solar power generation predictions to determine the best possible location for a solar plant.

## References

Kumar, K. R., & Kalavathi, M. S. (2018). Artificial intelligence based forecast models for predicting solar power generation. *Materials Today: Proceedings*, *5*(1), 796–802. https://doi.org/10.1016/j.matpr.2017.11.149

Sharma, N., Sharma, P., Irwin, D., & Shenoy, P. (2011). *Predicting solar generation from weather forecasts using machine learning* [Conference session]. IEEE International Conference on Smart Grid Communications, Brussels, Belgium. https://ieeexplore.ieee.org/document/6102379