Proceedings World Geothermal Congress 2020+1
Reykjavik, Iceland, April - October 2021

Inferring Interwell Connectivity in Fractured Geothermal Reservoirs Using Neural Networks

Halldora Gudmundsdottir and Roland N. Horne

Department of Energy Resources Engineering, Stanford University, Stanford, California, USA

halldora@stanford.edu

Keywords: connectivity, reservoir characterization, neural networks, tracer

ABSTRACT

The goal of this study was to use machine learning methods to characterize the fractured nature of geothermal reservoirs, specifically neural networks. A series of simulations were performed on discrete fracture networks with different fracture characters. For each of the fracture networks, a neural network was trained using injection rates from two injectors as input and produced tracer concentration at one producer as output. A sensitivity analysis was applied on the trained models to quantify the significance that each injector has on the producer which was then translated to a connectivity index describing the injector-producer relationship. The connectivity indices from the neural network were compared to connectivity indices obtained using the ACE algorithm as well as simulated peak arrival time of the tracer. In many cases the neural network connectivity was in agreement with the connectivity from peak arrival times, but the results were not conclusive in determining if neural networks could in fact quantify the strength of connection between wells.

1. INTRODUCTION

Fractures and faults are the main conduits for flow in geothermal reservoirs and thus control the mass and heat transport within the geothermal systems. To maintain the pressure and fluid volume in the reservoir, reinjection of waste water is performed is a standard practice. Although the purpose of reinjection is to improve the reservoir's performance, it can also have detrimental effects. Highly conductive fractures can quickly carry cold water to production wells without effective thermal sweep of the rock, lowering temperature locally and causing thermal breakthrough at the wells. Therefore, understanding the layout of fractures and how reservoirs respond to stimulation, such as injection and production, is necessary for proper design of reinjection schedules. Such an understanding is obtained through predictive analysis, which is a challenging but very important reservoir engineering task for optimal management of geothermal wells.

Reservoir characterization is the process of determining the reservoir architecture and characteristics, establishing fluid-flow trends, constructing reservoir models, and identifying reserve growth potential. Characterization of fractured reservoirs has been the subject of numerous papers. Production and injection rates, which are among the most readily available data, have been used to infer well-to-well connectivity to optimize injection schedules. Heffer et al. (1997) calculated the Spearman's rank correlation coefficient between oil production rates, reflecting the well-to-well connections throughout the reservoir. Refunjol and Lake (1999) applied the Spearman's analysis in combination with geological features and tracer data to determine preferential flow paths in oil reservoirs. Tian and Horne (2016) proposed a new method, the modified Pearson's correlation coefficient, to calculate the correlation between injection and production wells, and compared the results to the Spearman's analysis, highlighting the method's advantages in identifying faults within the reservoir. The capacitance-resistance model, described by Yousef et al. (2006) is another approach for estimating well connectivity. The model is based on governing material-balance equations at reservoir conditions and used to estimate two coefficients, one for the degree of connectivity and one for the amount of fluid stored between the wells. The main limitation of the method is that the modelling parameters are constant while rapid changes in the number of active producers require them to vary. Recently, because of their nonlinear capabilities, attention has been given to Artificial Neural Networks (ANN). Demiryurek et al. (2008) generated an ANN with injection rates and production data and applied sensitivity analysis to quantify interwell connectivity. Sarkheil et al. (2009) presented a method of predicting natural fracture distribution between two wells using ANN, based on image logs and core measurements. Furthermore, Alizadeh et al. (2015) used ANNs to predict the fractures dip inclination degree of a well based on image log and other geological log data from two other wells nearby.

The geothermal industry has sought inspiration to the oil and gas industry by applying similar methods for characterization of geothermal reservoirs. On example of inspiration is the research by Sullera (1998). She isolated short-term variations in produced chloride concentration and injection rate using Wavelet transformations. These short-term variations were then used in multiple regressions to quantify connections between injectors and producers. Horne and Szucs (2007) further developed this idea by using nonparametric regression on the same data set without making assumptions about the underlying form of the relationships. Another example is demonstrated in Juliusson (2012)), but he developed a novel method for characterizing fractured reservoirs using flow rate and tracer data. The flow rate data were used to calculate an interwell connectivity matrix, describing how injected fluid is divided between producers. A method referred to as M-ARX was used to compute the interwell connectivity, but the main advantage of the method was that producer-producer interactions could be estimated as well as injector-producer interactions.

The objective of this study is to investigate the capabilities of machine learning methods to characterize geothermal reservoirs. Specifically, we explore neural networks and their ability to quantify well-to-well connectivity within the reservoir. Our proposed method is compared to the ACE algorithm, an earlier form of machine learning, and simulated peak arrival time of tracers. We perform this study on a simulated synthetic dataset of fractured geothermal reservoirs.

2. METHODOLOGY

For this study we used two machine learning algorithms, Alternating Conditional Expectations (ACE) and neural networks. Here we will briefly describe each method.

2.1 Alternating Conditional Expectations (ACE)

The Alternating Conditional Expectations (ACE) algorithm is a nonparametric regression method that seeks to build a model, fitting the data without assuming an underlying form of the relationship between variables. The ACE algorithm was developed by Breiman and Friedman (1985) and its objective is to find transformations $\varphi_1, \dots, \varphi_p$ of the predictor variables $X_1, \dots, X_p$ and a transformation θ of the response variable Y that maximizes the correlation between $\theta(Y)$ and $\sum_{i=1}^{p} \varphi_i(X_i)$. The general form of the ACE regression model is written as:

$$\theta(Y) = \sum_{i=1}^{p} \varphi_i(X_i) + \epsilon \tag{1}$$

where ϵ is the error that is not explained by the regression. The error is related to the maximum correlation coefficient by $\epsilon^2 = 1 - \rho^2$. Therefore, maximizing the correlation is equivalent to minimizing the error variance (under the constraint that $E[\theta^2(Y)] = 1$):

$$\epsilon^2\big(\theta, \varphi_1, \dots, \varphi_p\big) = E[(\theta(Y) - \sum_{i=1}^{p} \varphi_i(X_i))^2] \tag{2}$$

In order to minimize the error, optimal transformations θ^* and φ^* are defined. The ACE algorithm gives estimates of these transformations by the following iterative process.

1. Set $\theta(Y) = Y/\|Y\|$
2. Compute new φ with $\varphi(X) = E[\theta(Y)|X]$. By defining $\varphi(X)$ as the conditional expectation of $\theta(Y)$ given X, $\varphi(X)$ becomes the function of X most correlated with $\theta(Y)$.
3. Compute new θ with $\theta(Y) = E[\varphi(X)|Y]/\|E[\varphi(X)|y]\|$. To fulfill the constraint of $E[\theta(Y)] = 1$, the new $\theta(Y)$ needs to be divided with the standard deviation.
4. Repeat Step 2 and 3 until ϵ^2 fails to decrease.
5. Now, θ and φ are the solutions to θ^* and φ^*.

As can be seen, the procedure involves *alternating conditional expectations,* hence the name of the algorithm (ACE). For the multiple predictor case in Eq. (2), the transformations of $\varphi(X)$ and $\theta(Y)$ can be expressed as:

$$\varphi_i(X_i) = E[\theta(Y) - \sum_{j \neq i}^{p} \varphi_j\big(X_j\big)|X_i] \tag{3}$$

$$\theta(Y) = E[\sum_{i=1}^{p} \varphi_i(X_i)|Y]/||E[\sum_{i=1}^{p} \varphi_i(X_i)|Y]|| \tag{4}$$

After the iterations, the final transformation functions $\varphi_1(X_1), \dots, \varphi_p(X_p)$ and $\theta(Y)$ are assumed to be the optimal transformations $\varphi_1^*(X_1), \dots, \varphi_p^*(X_p)$ and $\theta^*(Y)$. Finally, the predictor and response variables are related in the transformation space through:

$$\theta^*(Y) = \sum_{i=1}^{p} \varphi_i^*(X_i) + \epsilon^* \tag{5}$$

where ϵ^* is the minimization error not captured by the ACE transformations and is assumed to have a normal distribution with zero mean (Nguyen-Cong and Rode, 1995; Wang and Murphy, 2004).

Based on the optimal transformations, a measure of connectivity between injector *i* and a given producer can be established with the following formula (Horne & Szucs, 2007)

$$I_i = \frac{1}{m}\sum_{j=1}^{m}\left|\varphi_i^*(X_i(t_j))\right| \tag{6}$$

2.2 Neural Networks

In its simplest form a standard neural network consists of an input layer, a hidden layer and an output layer. For a neural network to be considered a deep learning model it must contain more than one hidden layer (Goodfellow, Bengio, & Courville, 2016; Tian & Horne, 2015). An example of a feedforward neural network with one hidden layer is shown in Figure 1. Notice that we denote this neural network as a 2-layer neural network, that is we do not count the input layer. The layers in the network consist of neurons, that perform the computations for the output. Most neural networks are fully connected, meaning that each neuron is connected to every other neuron in the layers on either side, but neurons within a single layer share no connections. The strength of the connections between neurons are represented by weights that can either be positive (amplifying effects) or negative (dampening effects). The higher the weight, the more influence a neuron has on the other neurons it is connected to.

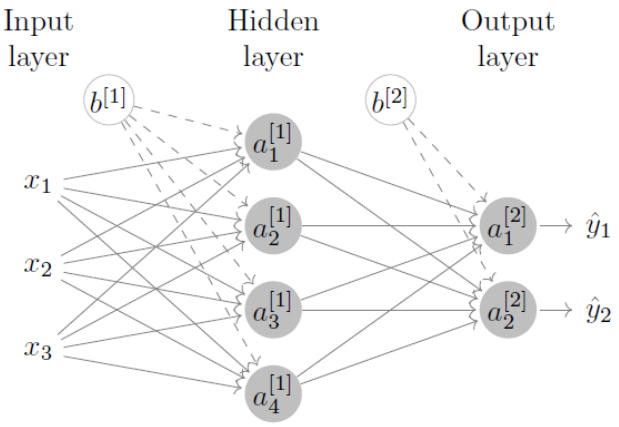


Figure 1: A 2-layer neural network with one hidden layer. Nodes represent neurons and arrows the connection between neurons.

The mapping from $\boldsymbol{x} = [x_1, x_2, x_3]$ to $\hat{\boldsymbol{y}} = [\hat{y}_1, \hat{y}_2]$ can be written as:

$$\boldsymbol{z}^{[1]} = \boldsymbol{W}^{[1]}\boldsymbol{x} + \boldsymbol{b}^{[1]} \tag{7}$$

$$\boldsymbol{a}^{[1]} = g^{[1]}\left(\boldsymbol{z}^{[1]}\right) \tag{8}$$

$$\boldsymbol{z}^{[2]} = \boldsymbol{W}^{[2]}\boldsymbol{a}^{[1]} + \boldsymbol{b}^{[2]} \tag{9}$$

$$\boldsymbol{a}^{[2]} = g^{[2]}\left(\boldsymbol{z}^{[2]}\right) = \hat{\boldsymbol{y}} \tag{10}$$

where $\boldsymbol{W}^{[l]}$ is the weight matrix of the connection between neurons, $\boldsymbol{b}^{[l]}$ the bias term and $g^{[l]}$ the activation function, for each layer $l = 1, \ldots, L$. When we train our neural network, we aim to optimize the model parameters $\boldsymbol{W}^{[l]}$ and $\boldsymbol{b}^{[l]}$ such that the predictions closely match the actual outputs for all training samples $i = 1, \ldots, m$. To quantify how we are achieving this goal we define a cost function discrepancy between the predicted value and the desired output. The objective of a neural network is to minimize the cost function which is usually accomplished through an iterative optimization algorithm such as gradient descent.

2.2.1 Sensitivity Analysis

Different methods have been used to interpret what has been learned by a trained neural network. One such method is an analytical sensitivity analysis. The sensitivity study quantifies the effect of a small perturbation of one of the network's parameter, for example an input or hidden neuron value or a weight, on the output of the neural network (Montano & Palmer, 2003). In our study, we were interested in estimating how much each injector impacts the geothermal production at the producer, or essentially measuring the relative influence on the output, due to a change in the input. We used a sensitivity analysis proposed by Zurada et al. (1993) for this purpose that involves calculating partial derivatives of the output $\hat{y}_k$ with respect to the input x_i, forming a Jacobian or a sensitivity matrix.

$$S_{jk} = \frac{\partial \hat{y}_k}{\partial x_j} \tag{11}$$

Notice that the output layer $\hat{\boldsymbol{y}}$ can also be symbolized as $\boldsymbol{a}^{[L]}$ for a L-layer neural network as can be seen in Eq. (11). For the 2-layer neural network in Figure 1, by applying the chain rule, the partial derivatives can be written in terms of the network's weights

$$\frac{\partial \hat{y}_k}{\partial x_j} = g^{[2]\prime}(z_k^{[2]}) \sum_{n=1}^{n_h^{[1]}} w_{kn}^{[2]} g^{[1]\prime}(z_n^{[1]}) w_{nj}^{[1]} \tag{12}$$

where $n_h^{[1]}$is the number of neurons for the hidden layer, $w_{nj}^{[1]}$ is the weight between neuron j in the input layer and neuron n in the hidden layer, $w_{kn}^{[2]}$ the weight between neuron n in the hidden layer and neuron k in the output layer, and $g_n^{[1]\prime}$and $g_k^{[2]\prime}$ are the derivatives of neurons n and k in the hidden and the output layer, respectively. There are many activation functions available, such as the sigmoid function, tanh function and the reLU function. As an example, the derivative of the sigmoid function is written as

$$g'(x) = \frac{1}{1+e^{-x}}\left(1 - \frac{1}{1+e^{-x}}\right) \tag{13}$$

Writing Eq. (12) in matrix form instead of summation form provides us with a more succinct definition

$$\boldsymbol{S} = \boldsymbol{G}^{[2]\prime}\boldsymbol{W}^{[2]}\boldsymbol{G}^{[1]\prime}\boldsymbol{W}^{[1]} \tag{14}$$

where $\boldsymbol{W}^{[1]}_{n_h^{[1]}\times n_h^{[0]}}$ and $\boldsymbol{W}^{[2]}_{n_h^{[2]}\times n_h^{[1]}}$ are the hidden layer and output layer weight matrices, and $\boldsymbol{G}^{[1]\prime}_{n_h^{[1]}\times n_h^{[1]}}$ and $\boldsymbol{G}^{[2]\prime}_{n_h^{[2]}\times n_h^{[2]}}$ the diagonal activation derivative matrices. The dimensions of the matrices in Eq. (14) are denoted as subscripts where $n_h^{[0]}$, $n_h^{[1]}$ and $n_h^{[2]}$ is the number of neurons in the input, hidden and output layer, respectively. For a L-layer neural network, Eq. (14) can be written as

$$\boldsymbol{S} = \prod_{l=1}^{L} \boldsymbol{G}^{[L+1-l]\prime}\boldsymbol{W}^{[L+1-l]} \tag{15}$$

When training a neural network, numerous training samples are used to calculate the error needed to update the weights and biases. Eq. (15) defines a sensitivity matrix for a single training sample; however, each training pair $[\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)}], i = 1,\dots,m$, produces a different sensitivity matrix $\boldsymbol{S}^{(i)}$. The reason lies with the activation functions for each layer, whose values change across the set of training samples, which in turn produces distinct diagonal matrices of the derivatives $G^{[l]\prime}, l = 1,\dots,L$. Since, different input values or training samples give new sensitivity matrices, the overall sensitivity, evaluated over the entire training set, is found as the mean square average

$$\boldsymbol{S}_{avg} = \sqrt{\frac{\sum_{i=1}^{m}(\boldsymbol{S}^{(i)})^2}{m}} \tag{16}$$

From the sensitivity matrix, inquiries about the influence of each injector on the producer are made. The input values, associated with specific injectors, that produce high sensitivity are considered to be in better connection to the producer than input values that give low sensitivity. Studying specific sensitivity values enables us to rank the impact of each injector on the producer and estimate well-to-well connectivity that is valuable for designing reinjection-production schedules (Demiryurek, Banaei-Kashani, Shahabi, & Wilkinson, 2008; Engelbrecht, Cloete, & Zurada, 1995; Zurada, Malinowski, & Cloete, 1994).

3. RESULTS AND ANALYSES

3.1 Experimental Setup

In order to evaluate our proposed method for quantifying interwell connectivity we performed numerous simulation experiments. A series of 100 discrete fracture networks (DFNs) were generated and simulated under geothermal conditions using the reservoir simulator ADGPRS, developed at Stanford University. The DFNs were generated randomly with varying degree of fractures and orientation and thus with quite different character (see one example in Figure 2). All fracture networks contained two injection wells, located at the boundary of the reservoirs, and one producer in the middle of the reservoir. For the simulations, we used varying injection schedules, where we injected increasing amount of tracer over the simulation time. This was done to mimic a situation where natural tracers (e.g. chloride produced in brine) are constantly being reinjected into the reservoir. For both the neural network and the ACE algorithm, injection rates were used as inputs and produced tracer concentration as output.

Before we could quantitatively compare one method to the next, we needed to establish a point of reference. For our synthetic library of fractured reservoirs, we simulated two tracer tests, one for Injector 1 and another for Injector 2. The tracer tests were conducted by continuously injecting a fixed concentration of tracer into an injector while monitoring the produced concentration at the producer. An example of tracer return curves (and the associated fracture network) is shown in Figure 2. The peak arrival time (t_p) of the tracer was used to quantify the connection between the injectors and the producer. The less time it takes for the peak of tracer concentration to return, the stronger the connection is between the wells. For intuitive comparison, we created a reference interwell connectivity index by taking the inverse of the peak arrival time, with high value corresponding to strong connection and low value to poor connection.

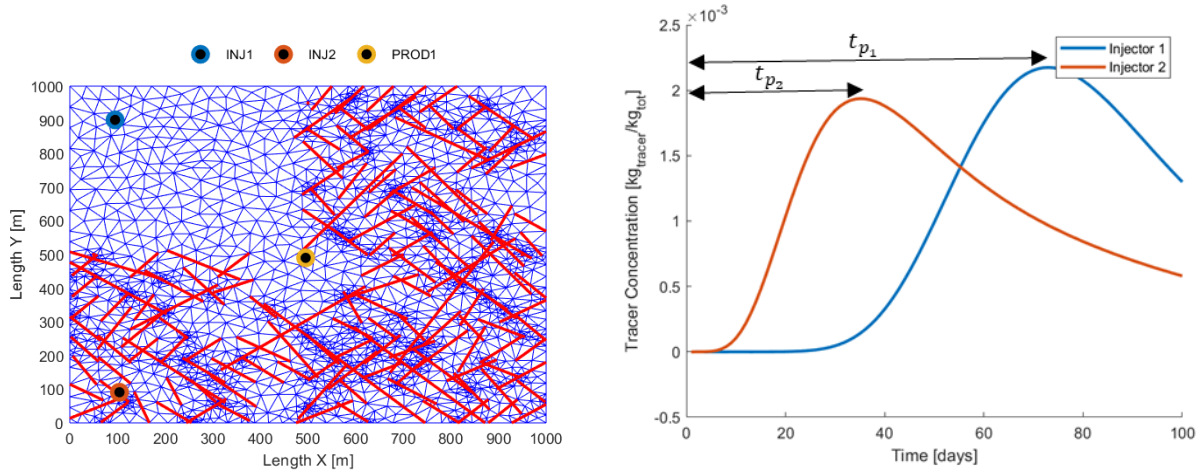


Figure 2: Tracer return curves (right) for a producer in a fractured geothermal (left) from two separate tracer tests. The peak arrival time of the tracer is indicated as t_p for the tracer return curves.

3.2 Training the Neural Network and Comparative Study

We used a standard feedforward neural network often referred to as Multilayer Perceptrons (MLP) on the synthetic library of fracture networks. Each fracture network or dataset contained 440 data points, one for each day over the course of 440 days. A neural network was trained for each fracture network with injection rates as inputs and produced tracer concentration as output. We used a 70% of the datasets as training data and 30% as testing data. Hyperparameter tuning was used to find the optimal configuration for the neural network algorithm. A subset of randomly selected parameters such as number of layers, number of neurons, number of time lags and type of activation functions was generated and the configuration with the lowest test error selected. The optimal model configuration found included: six layers, 170 neurons, five time lags and reLU as the activation function.

Smooth and well-behaved signals are generally not found in reality, and noise is inherently a part of measurements in geothermal measurements. Therefore, we both explored training our algorithms with smooth signals as well as adding 3% Gaussian noise. An example of noisy input data and output data (for the fracture network in Figure 2), is shown in Figure 3. After training the model, the same noisy input data was fed into the algorithm. The fitted concentration is much smoother than the noisy training data, indicating that the model was able to learn the pattern behind the noise rather than fitting the noise itself.

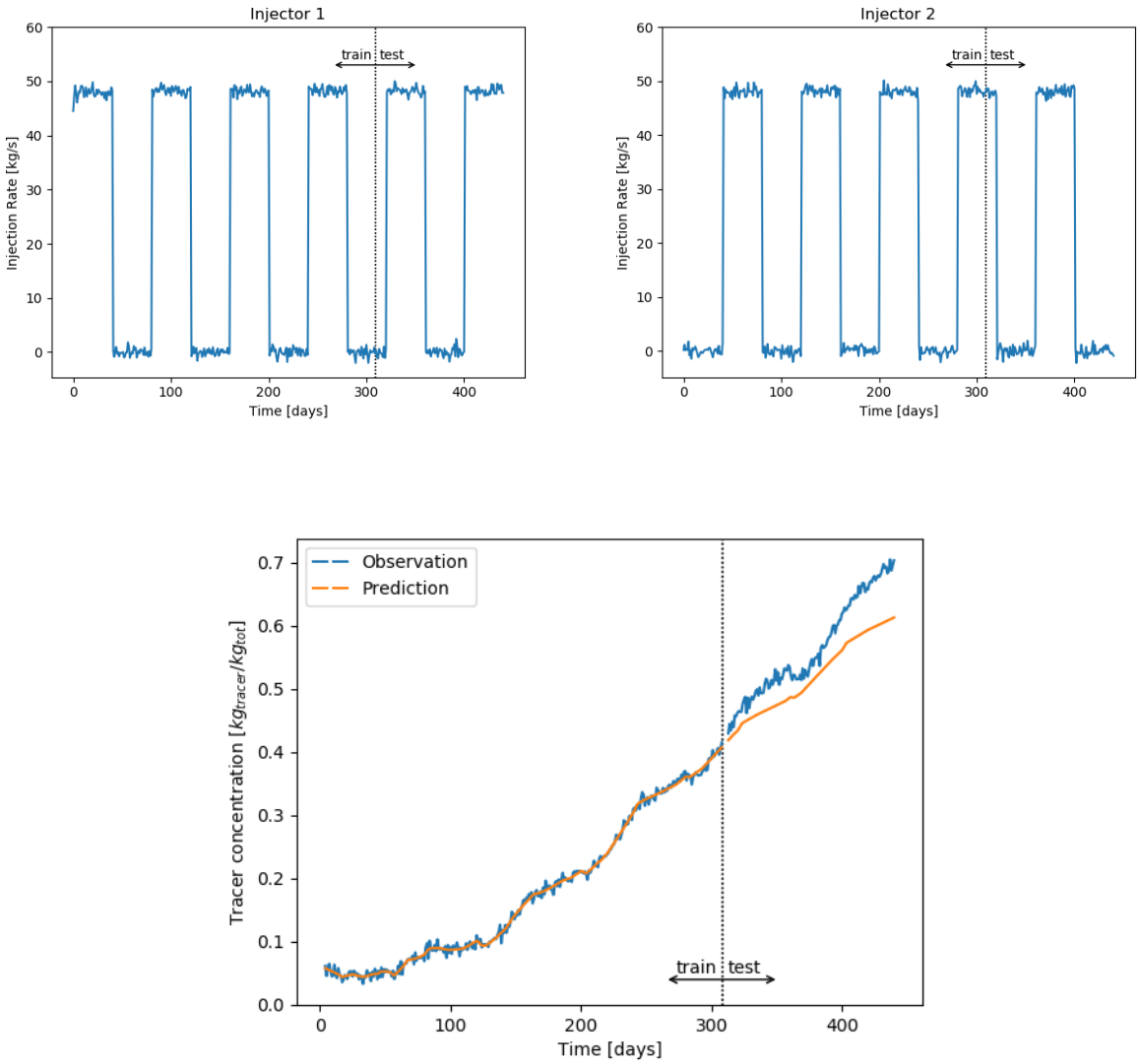


Figure 3: Input data for the neural network from Injector 1 (top left) and Injector 2 (top right). Observed and predicted tracer concentration at producer (bottom).

Once the model is sufficiently trained, it can be used for predictive modeling of tracer return curves based on injection rates. However, the goal for this study was to estimate the strength of the connection between the injector-producer pairs. The sensitivity study, described in the previous chapter, was used to find the influence of each of the injectors on the producer. For the fracture network in Figure 2, the connectivity indices from the sensitivity study on the trained neural network were 0.342 for the Injector 1 – Producer 1 pair and 0.658 for the Injector 2 – Producer 1 pair. For comparison we applied the ACE algorithm on the same inputs and output and obtained connectivity indices of 0.306 for the Injector 1 – Producer 1 pair and 0.694 for the Injector 2 – Producer 1 pair. The actual connectivity from peak arrival time for this fracture network was 0.324 for the Injector 1 – Producer 1 pair and 0.676 for the Injector 2 – Producer 1 pair. Both the neural network and the ACE algorithm were able to estimate the connectivity between the injectors and producer reasonably well for this specific fracture network.

In order generalize the method for a wider range of applications we applied both neural networks and ACE on a total of 100 different fracture networks. Figure 4 illustrates how well connectivity indices from the neural network compares to connectivity indices from peak arrival times of the tracer for Injector 1 and Injector 2, respectively. Similarly, Figure 5 shows how well the connectivity indices from the ACE algorithm compares to connectivity indices from peak arrival times of the tracer for the injectors. From these results we can see that in many cases the connectivity indices from neural networks and the ACE algorithm are similar to the ones from peak arrival times. However, there are also many cases where the opposite is true. For a perfect match the data points would align around the upward diagonal line, but we can see that the correlation for both model approaches is not strong. Still, there is a general upward trend which is reassuring, implying that the connectivity indices from neural networks and ACE can be indicative of the connection between well pairs.

From these results we cannot conclusively determine which method is better, the standard neural network or ACE. The ACE algorithm seems to perform better overall quantitively estimating the connectivity between well-pairs with a coefficient of determination of 0.4570 (R^2) compared to 0.2853 (R^2) when using neural networks. However, qualitatively the neural network performs slightly better. Simply ranking the connection of the injectors, which of the two injector has a stronger connection to the producer, the neural network got it right in 77% of the cases (compared to peak arrival time) compared to 73% for the ACE algorithm.

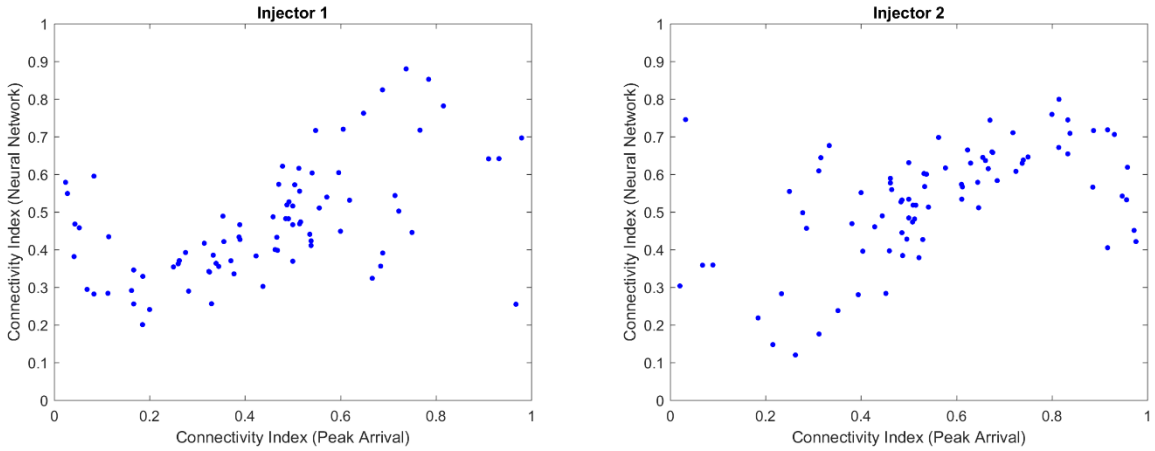


Figure 4: Comparison of connectivity indices from the neural network and peak arrival time for Injector 1 (left) and Injector 2 (right).

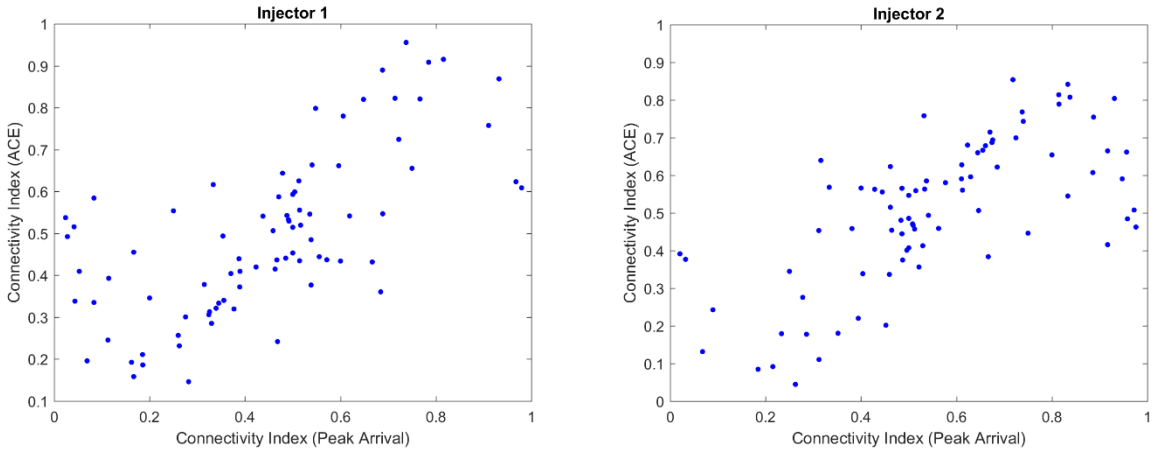


Figure 5: Comparison of connectivity indices from the ACE algorithm and peak arrival time for Injector 1 (left) and Injector 2 (right).

4. DISCUSSION

In this paper we discussed the possibility of using neural networks to estimate the strength of well-to-well connectivity in geothermal reservoirs. Having methods that can help determine the relationship between wells is very valuable, especially under varying well controls with injectors being opened and shut through time. During this study we found that neural networks can be beneficial in determining the connectivity within the fractured reservoirs. In many cases the strength of the connection using neural networks agreed with the connectivity found from peak arrival times. Also, the neural network was more often than not able to determine which of the two wells was better connected to the producer. However, the results at this point were not conclusive and more research is needed to determine if in fact neural networks can be used to characterize fractured reservoirs under geothermal conditions.

Future work includes investigating different types of neural network algorithms, specifically recurrent neural networks because of their ability to learn temporal dependence structures for challenging time series predictions. Furthermore, a more comprehensive hyperparameter tuning will be performed to ensure optimal configuration of the neural networks. Other well-variables will also be

considered for training the models. Here we used injection rates and produced tracer concentration, but other variables will be tested such as bottomhole pressure, produced temperature and electric potential difference between wells.

ACKNOWLEDGEMENTS

We would like to thank the Energy Resources Engineering Department at Stanford University and Landsvirkjun, National Power Company of Iceland for the financial support that made this research possible.

REFERENCES

Alizadeh, M., Junin, D., Mohsin, R., Movahed, Z., Alizadeh, M., & Alizadeh, M. (2015). Application of Artificial Neural Networks in Fracture Characterization and Modeling Techniques. In *Proceedings of the 17th International Conference on Mathematical and Computational Methods in Science and Engineering, Kuala Lumpur, Malaysia* (pp. 162–169).

Breiman, L., & Friedman, J. H. (1982). *Estimating Optimal Transformations for Multiple Regression and Correlation*. *Department of Statistics, Stanford University*. https://doi.org/10.1080/01621459.1985.10478157

Demiryurek, U., Banaei-Kashani, F., Shahabi, C., & Wilkinson, F. (2008). Neural-Network Based Sensitivity Analysis for Injector-Producer Relationship Identification. In *SPE Intelligent Energy Conference and Exhibition, Amsterdam, The Netherlands*.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press. Retrieved from http://www.deeplearningbook.org

Heffer, K. J., Fox, R. J., McGill, C. A., & Koutsabeloulis, N. C. (1997). Novel Techniques Show Links between Reservoir Flow Directionality, Earth Stress, Fault Structure and Geomechanical Changes in Mature Waterfloods. In *SPE Journal* (Vol. 2, pp. 91–98). https://doi.org/10.2118/30711-PA

Horne, R. N., & Szucs, P. (2007). Inferring Well-to-Well Connectivity using Nonparametric Regression on Well Histories. In *Proceedings, 32nd Workshop on Geothermal Reservoir Engineering, Stanford University*. Stanford, California.

Juliusson, E. (2012). *Characterization of Fractured Geothermal Reservoirs Based on Production Data*. *Stanford University*. Stanford University.

Montano, J. J., & Palmer, A. (2003). Numeric sensitivity analysis applied to feedforward neural networks. *Neural Computing & Applications*, *12*(2), 119–125. https://doi.org/10.1007/s00521-003-0377-9

Nguyen-Cong, V., & Rode, B. M. (1995). Application of Alternating Conditional Expectations Method to Quantitative Electronic Structure-Activity Relationships (QESAR). *Quantitative Structure-Activity Relationships*, *14*, 512–517. https://doi.org/10.1002/qsar.19950140604

Refunjol, B. T., & Lake, L. W. (1999). Reservoir Characterization Based on Tracer Response and Rank Analysis of Production and Injection Rates. *Reservoir Characterization - Recent Advances, AAPG Memoir 71*, 209–218.

Sarkheil, H., Hassani, H., & Alinia, F. (2009). The Fracture Network Modeling in Naturally Fractured Reservoirs Using Artificial Neural Network Based on Image Logs and Core Measurements. *Australian Journal of Basic and Applied Sciences*, *3*.

Sullera, M. (1998). *Inferring Injection Returns from Chloride Monitoring Data*. Stanford Univeristy.

Tian, C., & Horne, R. N. (2015). Applying Machine Learning Techniques to Interpret Flow Rate, Pressure and Temperature Data From Permanent Downhole Gauges. In *SPE Western Regional Meeting, California, USA, April 27-30, 2015*. https://doi.org/10.2118/174034-MS

Tian, C., & Horne, R. N. (2016). Inferring Interwell Connectivity Using Production Data. In *SPE Annual Technical Conference and Exhibition, Dubai, UEA*. https://doi.org/10.2118/181556-MS

Wang, D., & Murphy, M. (2004). Estimating Optimal Transformations for Multiple Regression Using the ACE Algorithm. *Journal of Data Science*, *2*, 329–346. https://doi.org/10.1177/0049124189018002005

Yousef, A. A., Gentil, P., Jensen, J. L., & Lake, L. W. (2006). A Capacitance Model To Infer Interwell Connectivity From Production and Injection Rate Fluctuations. *SPE Reservoir Evaluation & Engineering*, *9*(06). https://doi.org/10.2118/95322-PA