MACHINE LEARNING AIDED PRODUCTION DATA ANALYSIS FOR ESTIMATED ULTIMATE RECOVERY FORECASTING

A Thesis

by

LIUYI JIN

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Chair of Committee,	John Lee
Committee Members,	Duane McVay
	Yoonsuck Choe
Head of Department,	Duane McVay

May 2018

Major Subject: Petroleum Engineering

ABSTRACT

Estimated ultimate recovery (EUR) predictions are important in the petroleum industry. Many researchers have worked on implementing accurate EUR predictions. In this study, we used machine learning techniques to help predict the EUR range. We analyzed 200 Barnett shale wells with less than 170 months production history. We forecasted the production profile for each well using the modified Arps hyperbolic decline model. With the EUR values for 200 wells available, we forecasted the EUR of wells with limited production history by using three machine learning techniques, neural networks (NNet), support vector machine (SVM) and random forest (RF).

The results show that the 200 sorted EUR values predicted with the commercial decline analysis software, ValNav, follows a lognormal distribution as indicated on a log-probability paper plot. The P90, P50 and P10 EUR values were identified and the low P10/P90 value of 2.3 shows a low variance of EUR in this geologic area.

The production data were separated into eight groups and processed before being fed into the 3 machine learning algorithms. A four-fold cross-validation technique was employed to reduce the generalization error of the trained classifiers. The details of these 3 algorithms were also introduced. NNet performed best with highest test accuracy of 0.97 among the three machine learning algorithms employed with wells of 170 months' production history. In addition, we also tested the EUR prediction performance with 24, 48, 96, and 170 months' production history. The result shows that when we predict the

wells' EUR with increasing production history, we could achieve more accurate forecasting performance.

The results in this project can be used to help oil and gas companies make financial decisions based on available production data in the same geologic area. Also, this project can also help provide a basis for researchers who are interested in this direction.

Robustness analysis was implemented. The robustness of the algorithm is defined as the total distance of misclassified types to the correct types. Less total distance corresponds to more reliable and more stable performance for each individual algorithm. The NNet gives more robust performance with 100% misclassified samples classified into the types within one type distance to the correct types. RF is least robust. As the production history increases, the robustness of the three algorithms increases.

DEDICATION

To my father Changyi Jin, my mother Diping Zhou, and my sister Ying Jin for their support during the past 2 years.

ACKNOWLEDGEMENTS

I would like to thank my parents and my sister. They gave me strong support both financially and mentally in the period of pursuing this degree.

I would also like to thank my committee chair, Dr. John Lee. He is always very professional in providing me with academic help. I would like to give my thanks to my committee members, Dr. McVay and Dr. Choe, for their guidance and support throughout the course of this research.

Thanks also go to my friends and colleagues and the department faculty and staff for making my time at Texas A&M University a great experience.

Finally, thanks everyone in this university who gave me support.

CONTRIBUTORS AND FUNDING SOURCES

This work was supervised by a dissertation committee consisting of Professor John Lee and Professor Duane McVay of the Department of Petroleum Engineering and Professor Yoonsuch Choe of Department of Computer Science.

Graduate study was supported by an in-state tuition fellowship from Department of Petroleum Engineering at Texas A&M University.

The data analyzed for the Results and Discussion section was extracted from DrillingInfo. The software ValNav usage depicted in Results and Discussion were helped by Chief Research Engineer Randy Freeborn from 3esi-Enersight.

All other work conducted for the thesis was completed by the student independently.

TABLE OF CONTENTS

Page

LIST OF FIGURES

Page

1 INTRODUCTION

Unconventional resources have become dominant in the oil and gas world in recent years. As a carry-over from the era of conventional resources, decline curve analysis (DCA), which is dominated by the Arps decline model (Arps 1945), is the main method used to predict production of these resources, and therefore the main method used to predict estimated ultimate recovery (EUR).

Unconventional resources, including shale gas, shale oil, and coalbed methane, are gaining increasing attention from many researchers in the petroleum industry. As defined by Holditch (Holditch 2003), an unconventional reservoir is one "that cannot be produced at economic flow rates or that does not produce economic volumes of oil and gas without assistance from mass stimulation treatments or special recovery processes and technologies". This definition manifests the difficulties of unconventional resources extraction. Unlike extraction of conventional resources, the lengthy transient flow production period in unconventional resources limits the applicability of the Arps decline model, which was designed and validated in the conventional resources era.

Fig. 1 below used viscosity and permeability to distinguish conventional and unconventional resources. Besides the characteristics of the unconventional resources themselves, unconventional reservoirs themselves are relatively more heterogeneous. There are more geologic uncertainties as well (e.g., different geologic layering of reservoirs).

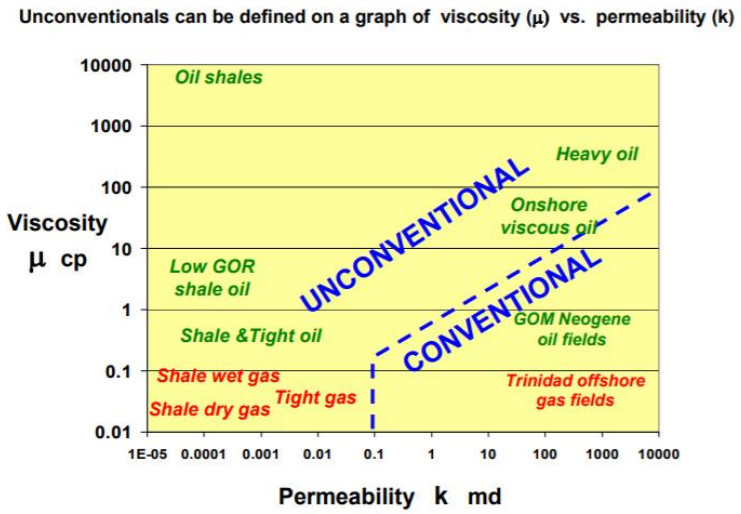


Fig. 1— Conventional and unconventional resources (adapted from BP 2012)

These conditions have caused unexpected errors when applying conventional reservoir production forecasting methods to unconventional reservoirs. Some researchers have proposed alternatives to the conventional Arps decline models (Arps 1945) to generate more accurate production data predictions. These modified models include Duong's method (Duong, 2011), and stretched exponential (Valko and Lee, 2010) and power law models (IIK et al., 2008), which can work well under certain constrained conditions for unconventional resources. Unfortunately, all these approaches can still be in error due to the many uncertainties introduced by unconventional resources. These uncertainties have limited the effective application of decline models in predicting EUR values for wells.

1.1 EUR Prediction

This study focuses on predicting EUR with the aid of machine learning methods, in contrast to methods others have used in the past to predict EUR. Xie (Xie et al. 2012) proposed a method to predict EUR for Haynesville shale gas wells. Xie normalized the actual rate to the rate corresponding to a constant operating pressure, followed by traditional Arps decline analysis to forecast EUR on the basis of normalized rate. Both field data and simulation results confirmed the advantages of this method. Sharma (Sharma and Lee, 2016) prepared a comprehensive and improved workflow for EUR prediction in unconventional reservoirs to address the problems of forecasts in ultra-low permeability reservoirs. Sharma validated and analyzed the workflow results and the conventional Arps decline methods. His results showed that the proposed workflow works better in most situations and lends perspective regarding the magnitude and scale of errors possible. In research more related to this study Crnkovic-Friis (Crnkovic-Friis and Erlandson, 2015) developed a novel approach adopting a deep neural network (DNN). In Crnkovic-Friis's work, more than 200,000 geological data points and more than 800 wells were used to train the DNN which correlates geology and average EUR. They found that the deep learning method was significantly more accurate than type curve region averages since the DNN model had captured the complex relationships between geological parameters and the expected EUR.

1.2 Machine Learning Applications in Petroleum Industry

Recent researchers in the oil and gas industry have applied AI, including machine learning techniques, to industry problems.

Ali (Ali 1994) reviewed the application of neural networks, a major machine learning technique, in the petroleum industry at that time. He pointed out that the capabilities of neural networks include "pattern recognition, classification of noisy data, nonlinear feature detection, market forecasting and process modeling," which makes them well suited to solve problems in petroleum industry. Ramgulam (Ramgulam et al. 2007) developed a trained artificial neural network to specifically address the history matching problem. Liu (Liu and Horne 2011) adopted data-mining techniques to analyze permanent downhole gauges. Liu exploited the least-mean-squares method and used the stochastic gradient descent method to train the parameters in a given polynomial equation to evaluate the relationship between pressure and flow rate. In his paper (Liu and Horne 2013b), Liu continued to approach the problems in interpreting pressure and flow rate data from permanent downhole equipment using data mining and machine learning methods, respectively.

Neural Network (NNet)

More recently, Jia (Jia and Zhang 2016) used neural networks to forecast production from the Barnett Shale. In his research work, Jia preprocessed the data by assigning different weights to each individual point in order to avoid the influence of outlier data. In the prediction section, he gave the history matching results of both time series and bottom hole pressure. The final prediction results from neural network model were compared with the results from exponential, harmonic and hyperbolic models, and neural network achieved much more accurate predictions than with the other three

conventional empirical models. **Fig. 2** illustrates the simple neural network structure that Jia employed (Jia and Zhang 2016).

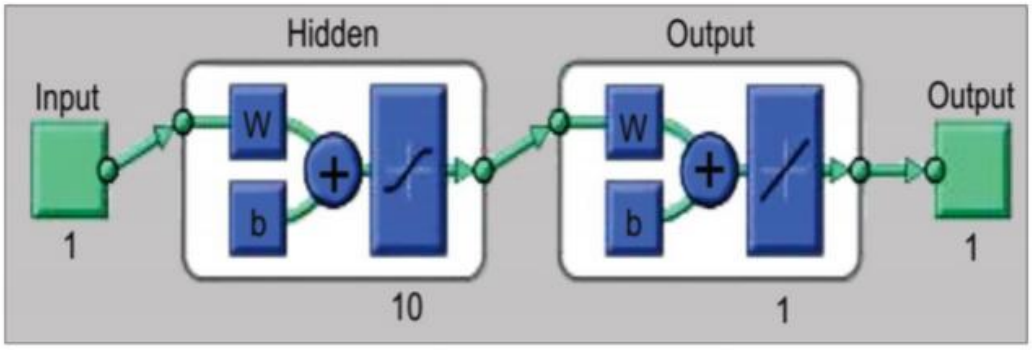


Fig. 2— Simple neural network structure (adapted from Jia and Zhang 2016)

Random Forest (RF)

Aulia (Aulia et al. 2014) states that a subset of the bottom-hole pressure (BHP) can be a contributing factor to the oil recovery factor in the field, so he combined Latin Hypercube Monte Carlo (LHMC) and random forest (RF) to identify such subsets. Aulia also pointed out that due to the fact that RF can identify the importance of RF's independent variables based on a collection of uncertainty runs, LHMC and RF combined can be a global approach to sensitivity analysis. Hedge (Hedge et al. 2015) showed the possibilities of using statistical learning methods like trees, bagging, and random forest to predict rate of penetration during real-time operations. Hedges also anticipated that random forest and bagging techniques could be employed to determine the relative importance of input parameters, which would further provide sound information for drilling parameter predictions and optimize on-the-fly rig floor changes.

Support Vector Machine (SVM)

Zhao (Zhao et al. 2006) trained an ε-intensive support vector machine (SVM) to implement the regression of water saturation from seismic data by using a water saturation curve calculated from density and resistivity logs of a gas well at the Gulf of Mexico. This provides a way to estimate the water saturation away from wellbore, which can be applied to distinguish between commercial and low saturation gas. Zhao (Zhao et al. 2014) gave an example of using proximal SVM (PSVM) to classify lithofacies; specifically, Zhao used PSVM to differentiate limestone from shale in a Barnett Shale gas play. Zhao's result was based on the two applications of PSVM, one for waveform classification and the other for the classification of well data. The promising results in both seismic and well log data demonstrated the validity of the PSVM classifier in binary classification. In Anifowose's paper (Anifowose et al. 2012), artificial neural networks (ANN) and SVM were both employed to predict porosity and permeability of oil and gas reservoirs with carbonate platforms. The results show that SVM performs better than ANN. In Anifowose's implementation, six datasets were used through the stratified sampling rather than the conventional static method of data division. SVM algorithm was expected to assist petroleum exploration engineers to estimate various properties with better accuracy, leading to reduced exploration time and increased production.

1.3 Problems and Objectives

As unconventional resources have been playing a more and more important role in the international energy market, effective and accurate methods to predict EURs for each well in unconventional reservoirs are in great need.

Machine learning techniques are powerful in solving classification problems. For different specific problems, the performance of different machine learning algorithms varies. In this study, we used three machine learning models to deal with the highly nonlinear relationship between variables. Thus, the performance of the machine learning algorithms also needs to be evaluated.

One of the main objectives of my research was to explore the advantages of machine learning in forecasting EURs for unconventional gas wells. In particular, neural networks (NNet), support vector machine (SVM), and random forest (RF) were adopted in this project to aid in the application of predictions of EURs. In brief, we have the following three objectives for this project:

(1) Generate EUR values for available wells with forecasted production profiles using the modified Arps hyperbolic model;

(2) Forecast EUR values for wells with only short production histories with the assistance of machine learning techniques;

(3) Evaluate the performance of different machine learning techniques in EUR forecasting and determine the technique most likely to produce accurate production forecasts.

1.4 Significance to Industry

This is a trial in petroleum engineering to use machine learning techniques to forecast EURs for wells with limited production histories. Machine learning techniques involved in this project aim to reliably classify wells with only short production histories into different EUR ranges; then, the EUR for a particular well with production history of limited length can be easily estimated. The results of this project should oil and gas companies to make financial decisions on the production histories of a limited number of production wells.

1.5 Outline of thesis

In this thesis, following the introduction, we first discuss the preprocessing of production data from 200 wells in Barnett Shale. The details of forecasting each well's EUR and the results are given. Next, we explain the data processing techniques we adopted before implementing the machine learning algorithms.

We discuss three machine learning algorithms: neural networks, support vector machine, and random forest. We implemented the three algorithms to classify the wells with only short production histories to one of eight types. We then discuss the details of the implementation and the performance of the three algorithms.

2 RESULTS AND DISCUSSION

In this section, I will discuss (1) details of how I implemented this project, (2) the results I found in the implementations, and (3) the conclusions I drew from the results.

2.1 Data Preprocessing

In petroleum production, production data are recorded as time series data; more specifically, they are reported as rate versus time (usually in months). The data used in this project were extracted from DrillingInfo (DrillingInfo 2017), which is a commercial data service specifically focusing on providing nation-wide oil and gas production data. I selected 200 gas wells from Barnett Shale reservoir that are actively producing. Selecting wells from the Barnett Shale helps that the wells are from a geologically similar area.

The 200 wells have different starting production times, but share the same ending production time of May 2017, which results in different productive lives. In our first step, we labeled these 200 samples.

A typical production profile is shown in **Fig. 3**. The production data in Fig. 3 is from the well with API number 4212132044, one of the 200 wells that we used in this project. The red dots show the trend of declining rate of production. This well produced for 157 months to May 1, 2017. With increasing time, the production rate will decrease below a threshold, the “abandonment rate,” which is the economic limit rate for the well.

Normally, the life of one particular unconventional gas well would be more than 360 months before it decreased to its abandonment rate. However, in this project, we forecasted each well's production to 360 months from the start of each well's production using ValNav. The 360 months well life and the economic limit, whichever comes first, would be assumed to be the end of the production, and the cumulative production at that time will be the EUR. In our project, the rate at 360 months was always greater than the economic limit rate, hence we used the cumulative production at 360 months as the EUR values for all wells. The forecasting process used the modified Arps hyperbolic model. Since we have well production data for only 170 months or less, our forecasting by implement the extrapolation first which is performed through switching from a transient flow decline model to a boundary dominated flow regime (BDF) at a specified switch point. The details and relevant concept explanations will be discussed in 2.2 Forecasting.

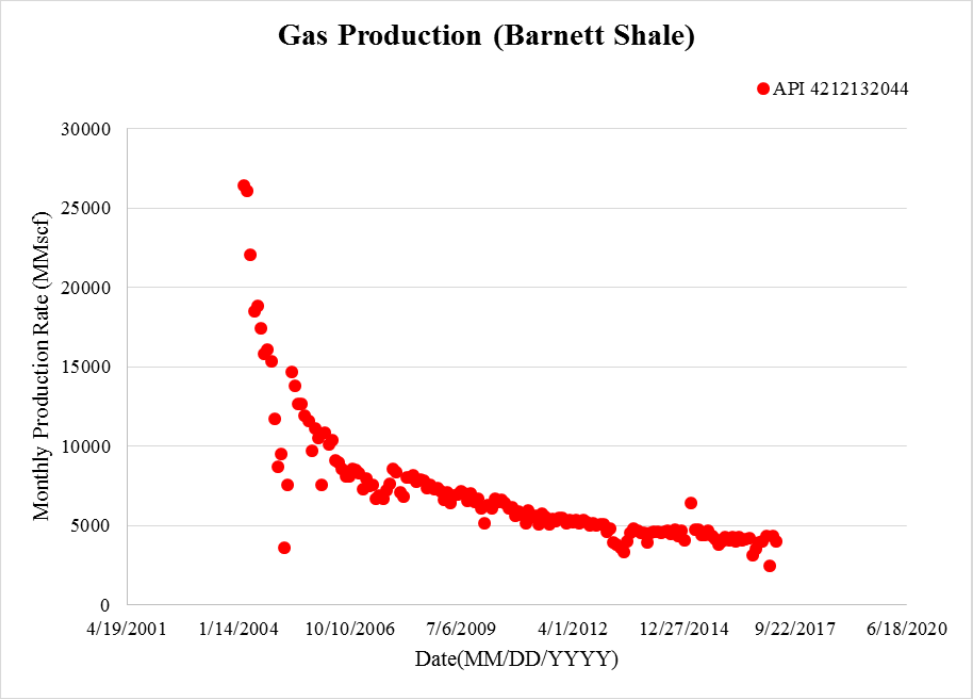


Fig. 3— Typical well production profile

2.2 Forecasting

As we see from Fig. 3, we have less than 170 months of production history. For the sake of reliably classifying the available dataset, we used ValNav to implement the production forecasting for all 200 wells. The mathematical models adopted in this process is modified hyperbolic model (Long and Davis 1987) shown in Eq.1 and Eq.2. Eq.2 indicates that decline rate decreases as time increases, while Eq. 1 illustrates that a production decline rate limit is imposed at the decline rate in the modified hyperbolic model. Before the decline rate decreases to the specified limit D_s, the production is assumed to be in transient flow, which displays a hyperbolic relationship between production rate and time in Eq. 1. When the decline rate decreases to the limit D_s, the production is assumed to be in boundary dominated flow regime (BDF). Boundary

dominated flow is defined as the period when pressure transient has reached all of the boundaries and the static pressure is declining at the boundary. In BDF, the decline rate remains constant, and the production is in exponential decline. As discussed by Seshadri (Seshadri and Mattar 2010), the choice of decline at the limit point D_s is normally based on either experience, a best-guess or through mutual agreement between production companies and their reserves assessors.

$$q = \begin{cases} \frac{q_i}{(1+bD_i t)^{\frac{1}{b}}}, D \le D_s \\ q_i e^{-D_s t}, D > D_s \end{cases} \quad \text{.....(1)}$$

$$D = \frac{1}{\frac{1}{D_i}+bt} \quad \text{.....(2)}$$

Two wells that have not reached BDF are shown in **Fig. 4**. The two wells were among the 200 wells in our Barnett Shale data set. As shown in the two log-log plots in Fig. 4, the blue dot lines show the raw data plotted in log-log scale – rate versus material balance time. In Fig. 4, the green solid line and red solid line represent the absolute slope value of 0.5 and 1, respectively, which corresponds to transient linear flow and boundary dominated flow. It is quite obvious that the slope of the decline trend line is still 0.5 by the end of May 2017, which is indicated by the green lines. This means that the two wells have not reached BDF when we try to forecast the production profile for them using modified hyperbolic decline model, and BDF is specifically indicated by the red

lines. Thus, we cannot directly use the available data to forecast the future production profile, which further prevents us from forecasting the EUR value for each well.

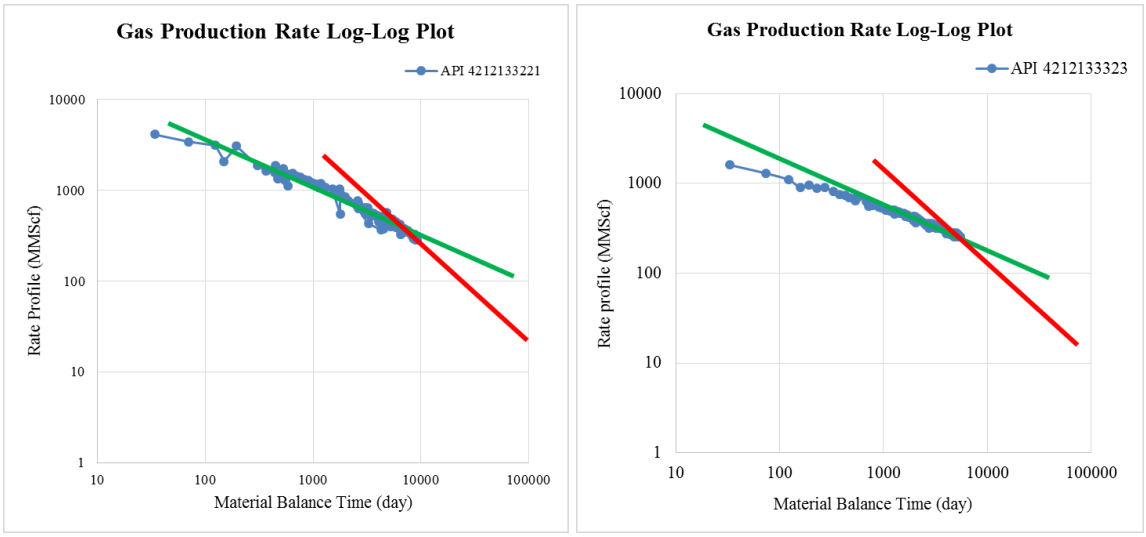


Fig. 4— Wells not reaching boundary dominanted flow

Eq. 3 gives the mathematical definition of material balance time.

$$t_m(t) = \frac{N_p(t)}{q(t)} \quad \text{.....(3)}$$

Palacio (Palacio and Blasingame 1993) proposed the concept of material balance time for variable rate and variable pressure rate analysis, which is defined as the ratio of cumulative production to flow rate. Before the material balance time was introduced, analysis of rate transient data was popular but wells are rarely produced at either constant rate or constant bottom-hole pressure for their entire history (Jha and Lee

2017), and material balance time was introduced to address this problem. According to Palacio's early work (Palacio and Blasingame 1993), we would expect a -0.5 slope during transient linear flow and -1 slope during BDF on a production rate versus material balance time plot.

In ValNav (ValNav 2017), we have three options to specify the switch point from linear flow to BDF: specifying a fixed decline slope (%/yr), specifying months after the start of first declining forecast segment, and specifying the months after start of history. In this project, we simply chose to specify the switch point by specifying the fixed decline slope as 6.5 %/year as shown in **Fig. 5**.

With the parameters set as shown in Fig. 5, ValNav will automatically transfer into BDF mode after the decline rate (%/yr) of transient flow at 6.5% is identified. When ValNav is implementing the forecasting with the switch point specified, it will fit the given data into the modified hyperbolic model (Eqs.1 and 2), and the production histories will be extrapolated to 360 months. In this way, we can make sure the 200 wells will have reached BDF by the end of the 360 months.

When using ValNav to predict future production, we can easily choose the best fit for each well to achieve the forecast. **Fig. 6** gives our parameter setting in ValNav for BestFit. The BestFit automatically uses the modified hyperbolic model to fit data first, and it will then forecast to the abandonment rate (i.e., 10 Mscf/d which we specified in this project) or 360 months, whichever occurs first.

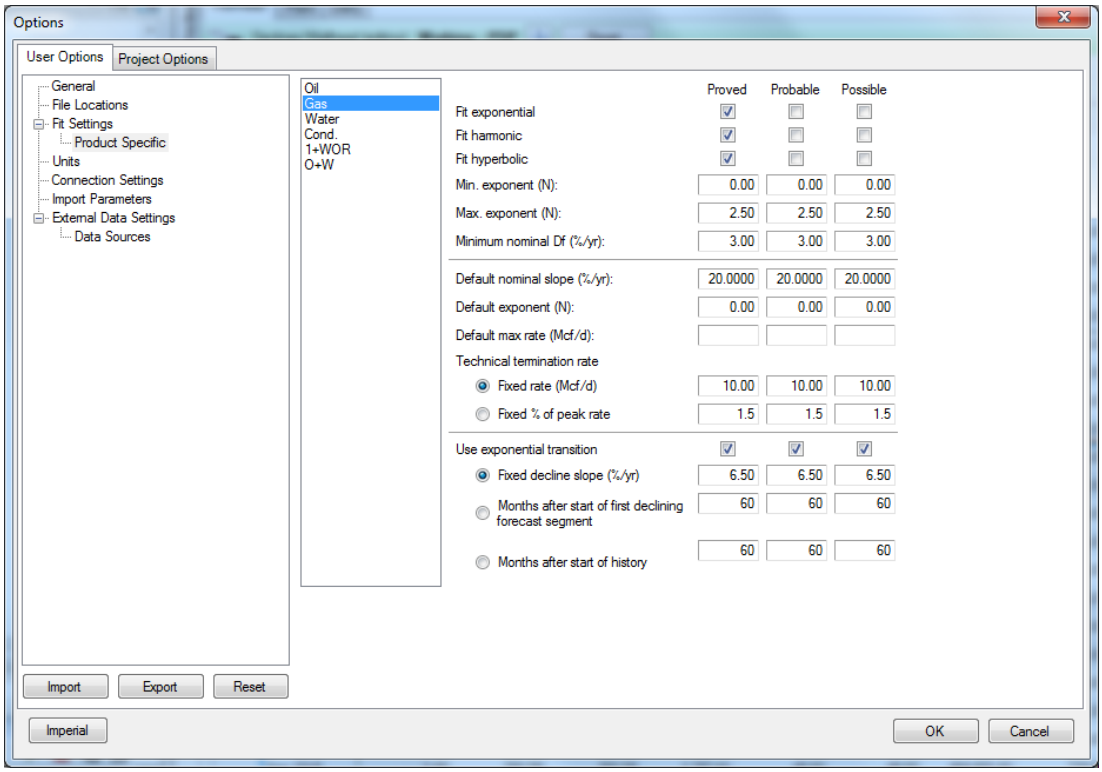


Fig. 5— Specifying the switch point in ValNav

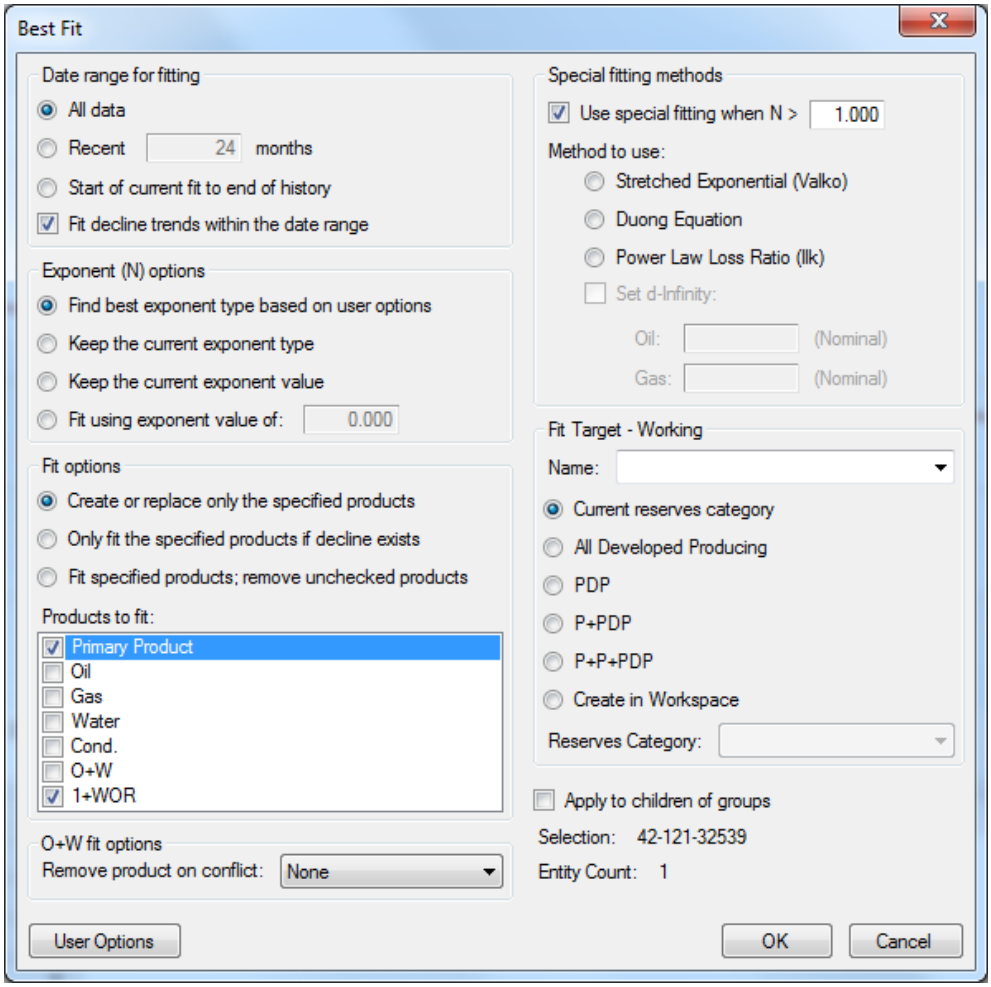


Fig. 6— Best fit parameter setting in ValNav

After ValNav has completed the BestFit procedure, we can generate a production profile for each well. The profile, as mentioned before, may have a producing life of more than 360 months. However, we use only the first 360 months' data in our analysis; i.e., the EUR of each well is assumed to be the cumulative production at the end of 360 months.

2.3 Lognormal Distribution of EUR

We sorted the 200 EUR values in descending order. The sorted EUR values for each well correspond to a "less than probability." We plotted the EUR distribution on log probability paper as shown in **Fig. 7**. The horizontal and vertical axis in Fig. 7 are EUR values and "less than probability," respectively. The majority of the EUR values are approximately located on a straight line, indicating a lognormal distribution of EUR values of wells from this geologic area.

With the insurance of BDF and the EUR value for each well, we can easily find the P90, P50, and P10 EUR values from the plot in Fig. 7; they are 1538 MMscf, 2448 MMscf, and 3759 MMScf, respectively. The P90 value is the EUR value corresponding to the less than probability of 0.1 in Fig. 7, and P50 and P10 values correspond to 0.5 and 0.9 respectively. The P10/P90 ratio at approximately 2.32, indicating a low distribution variance, which in turn manifest a low uncertainty in the distribution.

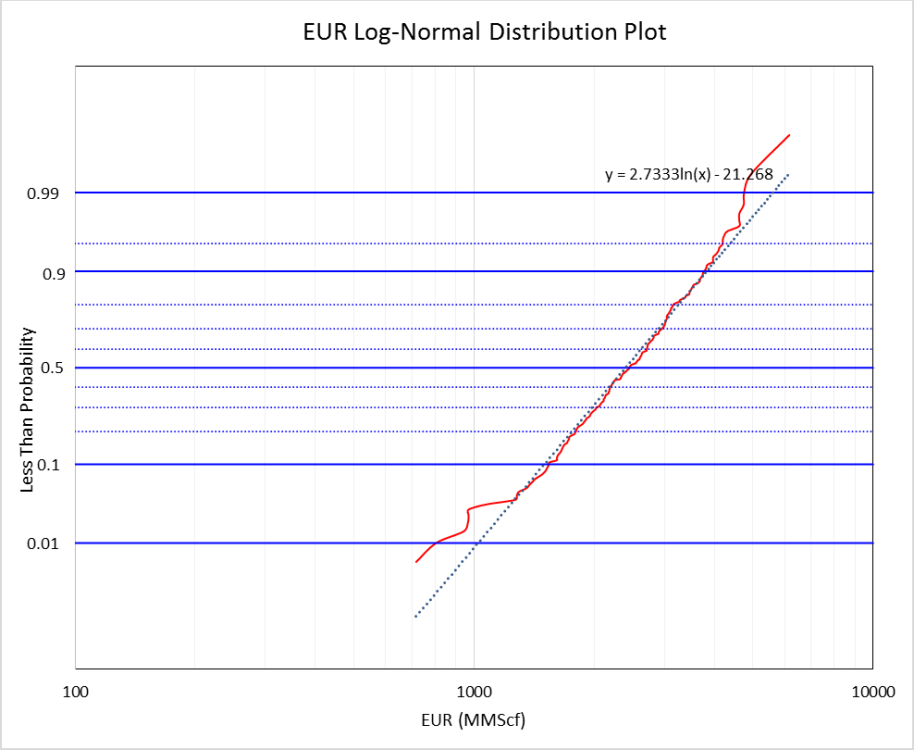


Fig. 7— EUR lognormal distribution

2.5 Labeling

With the aid of ValNav, we obtained the EUR value for each well. Since an assumed well life of 360 months was reached before the economic limit for the 200 wells in our dataset, we directly chose the forecasted production at 360 months data as the EUR for each well. In our machine learning application process, we classified each well as a given type such that they could be used to train the machine learning algorithms. The labels given to the well samples are called types. We separated the well samples into eight types (The corresponding Less Than Probability is denoted as P(Less Than)):

Type 1 – EUR $<$ 1642 MMScf (P(Less Than) $<$ 0.125)

Type 2 – 1642 MMScf $\leq$ EUR $<$ 1927 MMScf ($0.125 <$ P(Less Than) ≤ 0.25)

Type 3 – 1927 MMScf $\leq$ EUR $<$ 2181 MMScf ($0.25 <$ P(Less Than) ≤ 0.375)

Type 4 – 2181 MMScf $\leq$ EUR $<$ 2450 MMScf ($0.375 <$ P(Less Than) ≤ 0.5)

Type 5 – 2450 MMScf $\leq$ EUR $<$ 2745 MMScf ($0.5 <$ P(Less Than) ≤ 0.625)

Type 6 – 2745 MMScf $\leq$ EUR $<$ 3040 MMScf ($0.625 <$ P(Less Than) ≤ 0.75)

Type 7 – 3040 MMScf $\leq$ EUR $<$ 3680 MMScf ($0.75 <$ P(Less Than) ≤ 0.875)

Type 8 – 3680 MMScf $\leq$ EUR ($0.875 \leq$ P(Less Than))

Our objective is to classify the short production history data of a new well into one of the eight types. The EUR value range is determined for each well upon the identification of the well's type even though we do not have production data available for the full well life.

2.6 Uniform Input Dimensionality

In section 2.5 Labelling, we used each well's cumulative production at the end of 360 months (i.e., EUR) as the criterion to label well samples with one of eight different types mentioned above. In reality, when we classify the "relatively new" well samples with short history as one of the eight types, we are dealing with "relatively new" wells that don't have a production history of 360 months. In this project, the longest production history for any well is 170 months.

To ensure that each input sample to the machine learning algorithms has the same feature dimensions, we extrapolated each well's raw data to 170 months. This procedure

was performed simply by extracting the first 170 months' production data from the 360 months' production data that were used in the labelling section. The 170 months' production data was the actual input used to train the classifiers using machine learning algorithms.

Since we want to implement the EUR forecasting with only limited production history, the limited production history will be less than 170 months. It is obvious that accurate forecasts for wells with shorter production history will be more meaningful. To make our results more meaningful, we also added the options of extrapolating each well's production data beyond 24, 48 and 96 months into our considerations. For each well with available production longer than 24, 48 and 96 months, we chose the corresponding production period directly without any need of extrapolating the production history.

In the following discussion, the 200 well production data set will be seen as 200 samples, each with 24, 48, 96 and 170 features, respectively. This terminology is appropriate in machine learning fields.

2.7 Four-fold Cross Validation and Early Stopping Technique

A cross validation technique as shown in **Fig. 8** is commonly used to select the appropriate model parameter, and thus reduce the generalization error of machine learning algorithms. Before we implement the machine learning algorithms to classify the wells with only short production histories into one of the eight types as mentioned above, I separated the 200 samples into five groups. Four of them were used for cross-

validation, and the fifth group was used as the test data set. For each individual algorithm, I used three of the four groups as the training set, and one group was left to be the validation set. We repeated this training process four times with different data groups as the test set. After this training, the model parameters were fixed. The overall training accuracy of this algorithm is the average of the accuracies obtained from the four repeated training and testing cases. In this way, the generalization error (i.e., the risk of overfitting) of the algorithm will be minimized by not relying on any specific group of data that might not be representative of the whole dataset.

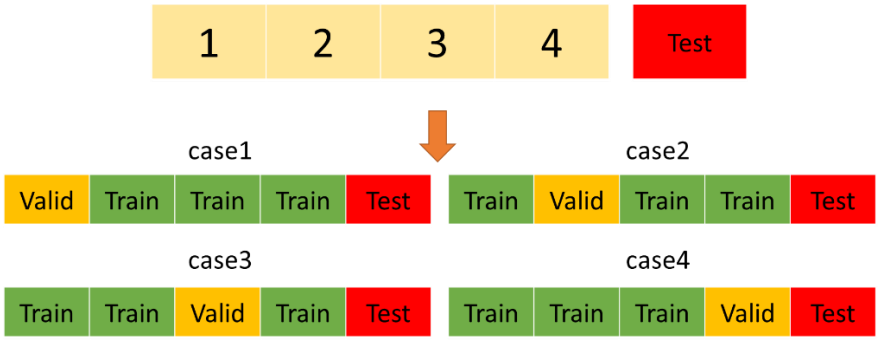


Fig. 8— Cross validation technique

Also, in this project, the early stopping technique is applied to avoid overfitting of our model. As explained in Wikipedia (Wikipedia 2018), in machine learning early stopping is a form of regularization used to avoid overfitting when training a learner with an iterative method, such as gradient descent. Such methods update the learner so as to make it better fit the training data with each iteration. Up to a point, this improves the learner's performance on data outside of the training set. Past that point, however,

improving the learner's fit to the training data comes at the expense of increased generalization error. Early stopping rules provide guidance as to how many iterations can be run before the learner begins to over fit. Early stopping rules have been employed in many different machine learning methods, with varying amounts of theoretical foundation.

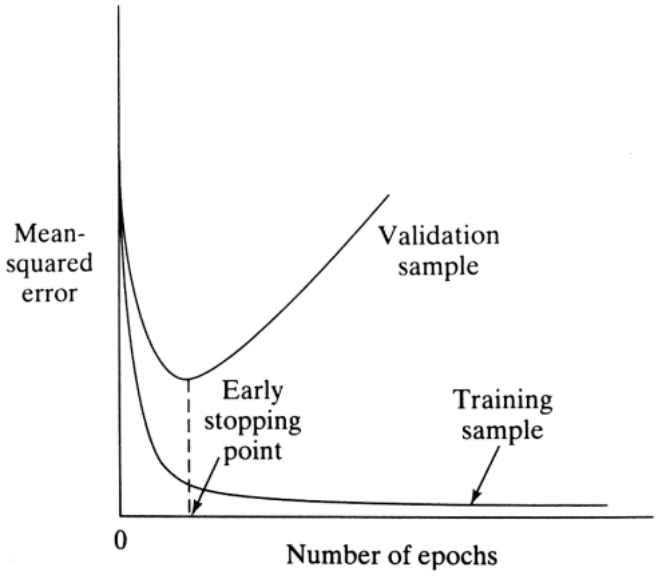


Fig. 9— Validation - based early stopping

Since we have the validation set mentioned above, we used validation-based early stopping. As shown in **Fig. 9**, for each iteration in the training process, as validation error begins to increase, we stop the training.

2.8 Machine Learning Algorithm Applications

The machine learning algorithms I used include neural networks (NNet), support vector machine (SVM) and Random Forest (RF); the pros and cons of each algorithm are discussed individually. In this section, I will introduce in detail the principles of the three algorithms, how I implemented the algorithms, and the results and corresponding performance evaluations.

2.8.1 NNet

Neural network algorithms are becoming very popular in solving many regression and classification problems. There are some variants of neural networks: convolutional neural networks (CNN), which are commonly used in image recognition; recurrent neural networks (RNN), which are widely used in natural language processing (NLP); multi-layer perceptron (MLP), and others. MLP can deal with high-nonlinearity problems appropriately with suitable parameter settings. In addition, we do not need to consider the dependence between each pair of variables nor the number of parameters or hyper-parameters when using MLP to solve our problems.

In this project, we chose the MLP as our training algorithm because it is simple to implement. The basic MLP architecture used in this project is constructed as shown as **Fig. 10**.

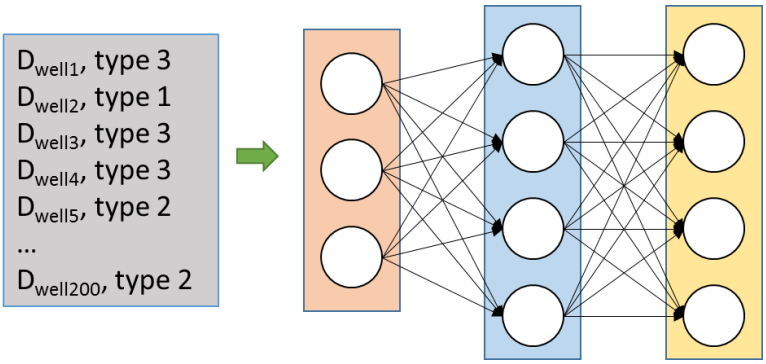


Fig. 10— Neural networks training architecture

In Fig. 10, there is only one hidden layer in the NNet architecture. The number of input layer neurons is set to be 24, 48, 96 and 170, respectively. The number of hidden layer neurons and output layer neurons are 167 and 4, respectively. The determination of hidden layer neurons numbers was based on the cross-validation process which will be discussed later. The classical logistic function was used as forward activation function. In the backpropagation process, the weights were updated using *lbgfs* solver, which is one of the quasi-newton options. In addition, the initial learning rate and momentum parameters were set to be 0.1 and 0.5, respectively, to improve the overall performance of this architecture.

Before the data training, I normalized 200 well samples. I used the min-max normalization for all the 200 samples to avoid abnormal feature values (e.g. some feature values are too low/high compared to others). As shown in **Fig. 11**, each column represents a particular feature of all samples, and each row represents all features of a particular sample. In the min-max normalization, each column is selected, and the

maximum and minimum values are identified. The normalized value is computed using Eq. 4. In this way, all the values are converted into the value range between [0, 1], and the risks of abnormal features are minimized.

$$val = \frac{value - Min}{Max - Min} \quad \text{.....(4)}$$

	Feature1	Feature2	Feature3	.....	.....	Feature170
Sample1	S1_F1	S1_F2	S1_F3	.....	.....	S1_F170
Sample2	S2_F1	S2_F2	S2_F3	.....	.....	S2_F170
Sample3	S3_F1	S3_F2	S3_F3	.....	.....	S3_F170
.....	...	...	...	...	...	...
.....	...	...	...	...	...	...
Sample199	S199_F1	S199_F2	S199_F3	.....	.....	S199_F170
Sample200	S200_F1	S200_F2	S200_F3	.....	.....	S200_F170

Fig. 11— Min-Max normalization

The backpropagation algorithm is the core of our neural network algorithm. Algorithm 1 illustrates the back propagation algorithm. It is based on stochastic gradient descent.

First, we initialize all weighs of all neuron connections to small random numbers near 0. Those numbers may vary between (-1, 1). Second, we enter into repeated iterations. In each iteration, 160 samples in the validation set are input into Algorithm 1 to be fed forward. The feed forward process produces an output, which is the type of each

individual sample input. We then compute the error at the output layer. Given the error from the output layer, we can back-propagate the error from output layer to input layer to update the weights accordingly until the stopping criterion is satisfied.

Algorithm 1 Back Propagation

Initialize all weights to small random numbers.
Until satisfied, Do
    For each training example, Do
        1. Input the training example to the network and compute the network outputs
        2. For each output unit j
           $\delta_j \leftarrow y_j(1-y_j)(d_j-y_j)$
        3. For each hidden unit h
           $\delta_h \leftarrow y_h(1-y_h)\sum_{j\in outputs} w_{jh}\delta_j$
        4. Update each network weight $w_{i,j}$
           $w_{ji} \leftarrow w_{ji} + \Delta w_{ji}$ where $\Delta w_{ji} = \eta\delta_j x_i$

Note: w_{ji} is the weight from i to j (i.e., $w_{j\leftarrow i}$)

Algorithm 1 was implemented using python package scikit-learn (MLPClassifier, scikit-learn 2017). The results for input sample of 170 feature dimensions are shown in **Fig. 12**.

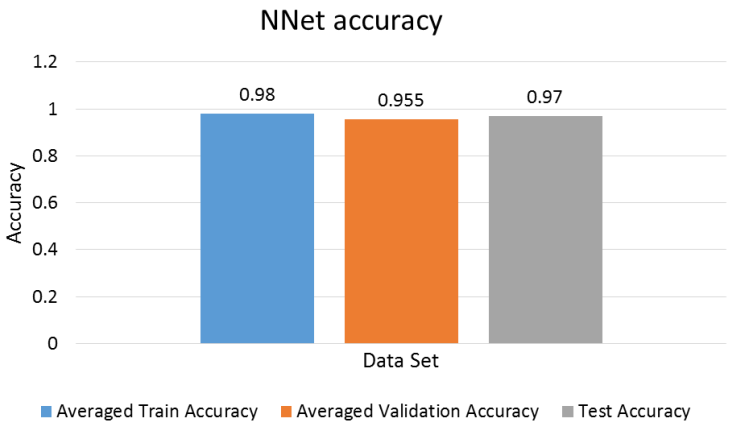


Fig. 12— Neural network algorithm accuracy

In Fig. 12, we can see that MLP can achieve 98% average training accuracy in the four-fold cross validation techniques. The average validation accuracy is 0.955. The test accuracy of 0.97 was evaluated as the overall criterion for the MLP algorithm. With the aid of NNet algorithm and the architecture parameter settings mentioned above, we have 97% confidence to correctly classify a well given the sample with a production history of 170 months. In addition, we also tested the cases of 24, 48 and 96 months of production. The test accuracy result is shown in **Fig. 13**.

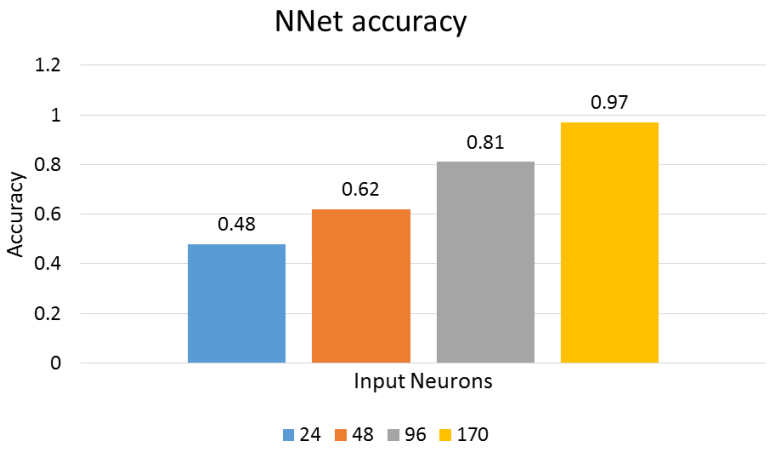


Fig. 13— Different input neuron number accuracy comparison

Fig. 13 shows that fewer input neurons in the neural network architecture would induce lower accuracies. When the well has only two years' production history (i.e., 24 months), the accuracy will be less than 0.5; when the production history is 170 months, the accuracy is more than 0.95. As the number of input neurons increases, the network architecture has more prior information, and thus it will be more accurate in predicting the EUR ranges.

However, the main limitation of MLP is that it cannot guarantee the globally optimal solution. It may "stick" in a locally optimal solution and then stop updating the weights for neuron connections. A good option to mitigate this is to have a random start position (e.g., randomly generate the weights for MLP each time) as we did initially in Algorithm 1 to initialize all weights to small random numbers. At the same time, we also added a momentum parameter, as we did when implementing Algorithm 1. This is another option that can help us reach the global optimum.

The performance of MLP can also be limited by the number of hidden layer neurons. Too many hidden neurons would cause overfitting problems, and too few may lead to underfitting. This dilemma is resolved by choosing the most suitable value between the number of neurons in the input layer and that in the output. As mentioned earlier, each well's production history was extrapolated to 170 months for the sake of uniform input dimensionality. Thus, the input layer has 170 neurons, while the output layer has only four neurons. Thus the potential number of hidden layer neurons candidate can be then chosen from the interval [4, 170]. **Fig. 14** gives the relationship between the number of hidden neurons and accuracy for validation sets in the cross validation process (i.e., the average four-fold cross validation test accuracy).

Fig. 14 shows that, the accuracy varies rapidly as the number of hidden neurons changes. The highest average validation accuracy was 0.97 when the number of hidden neurons was 167. Although the accuracy oscillates, we observe a generally increasing accuracy as the number of hidden neurons increases. This might be due to the fact that, the more hidden neurons in MLP, the more non-linear relationships it can understand. As we add more than 167 hidden neurons into the neural network architecture, in theory, the model will become more complex, which will further decrease the accuracies of the validation dataset. This is what we commonly call the overfitting problem.

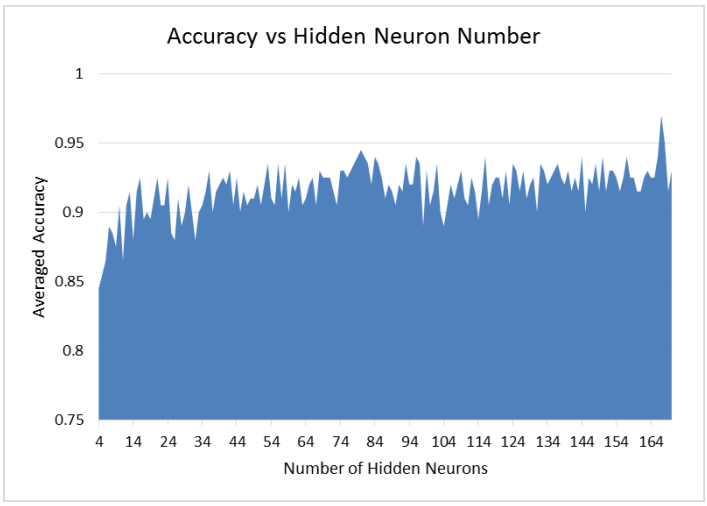


Fig. 14— Hidden neuron number determination through cross validation

Since the weights are initialized with random small numbers, each time we start a new training process, the number of hidden neurons that lead to the highest accuracy may vary. The results shown in Fig. 14 reflect only the determination that corresponds to the weights initialization in which 167 hidden neurons can achieve the best performance. Readers may obtain different results as they implement their own network architectures.

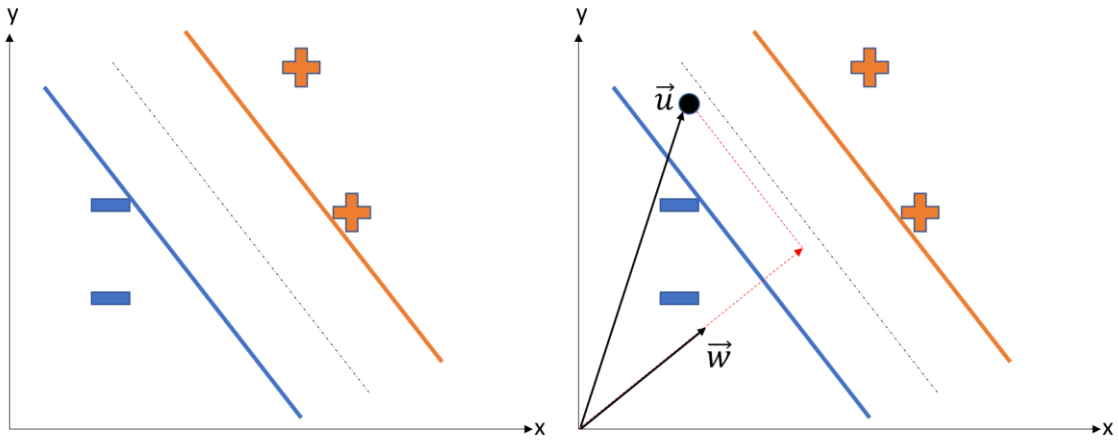
2.8.2 SVM

SVM is a classical machine learning algorithm. Its primary objective is to find a plane that can separate the samples with largest margin, i.e., maximize the margin. The margin is defined as the distance of closest samples from the separating hyperplane.

To simplify explanations of the SVM algorithm, assume that we are dealing with a binary classification problem. The possible label involved can only be either 1 or -1. Sometimes the label will be called simply either plus or minus. The multiple class

classification can be implemented simply by simply generating multiple SVM classifiers.

Suppose we have several samples that are labelled with minus and plus signs which represent two classes. We separate the samples with a plane that is represented by the dotted black line in **Fig. 15(a)**. The orientation of this plane can be different as long as the plane can separate the two kinds of samples into two different classes. The distance from the plane to the blue line and to the orange line is the same, and the points that are exactly located on the blue lines and orange lines are called support vectors. This is also the source of the classifier name "Support Vector Machine." The objective of the algorithm is to find such support vectors to maximize the margin (i.e., to minimize the generalization error).



(a) (b)

Fig. 15— Support vector machine principles (hard margin)

In **Fig. 15(b)**, we add a vector $\vec{w}$ perpendicular to the plane (dotted black line). This vector can be a unit vector. In the coordinate system as shown in Fig. 15(b), the dotted black line can also be represented by a vector $\vec{w}$. When we need to classify the new sample $\vec{u}$, we need to compute the length in the direction of $\vec{w}$, which is $\vec{w} \cdot \vec{u}$. If the length exceeds a certain constant c, then this new sample will be classified as a plus sign, otherwise, it would be defined as minus. To formally describe this situation, if the new point satisfies Eq. 5,

$$\vec{w}^T\vec{u} + b > 0 \quad \text{.....(5)}$$

then the point would be labelled a plus. Otherwise, the point will be a minus. Eq. 5 is also called the decision rule. We find a suitable vector $\vec{w}$ and b value to maximize the margin. To achieve this, we define

$$y_i\left(\vec{x_i}^T\vec{w} + b\right) - 1 = 0 \quad \text{..... (6)}$$

for each sample i between the blue line and the orange line. From Eq. 6, we can derive the width of the margin as in Eq. 7.

$$width = \left(\overrightarrow{x^+} - \overrightarrow{x^-}\right) \cdot \frac{\vec{w}}{\|\vec{w}\|} = \frac{2}{\|\vec{w}\|} \quad \text{.....(7)}$$

Given Eq. 7, to find vector $\vec{w}$ and the b value that maximize the margin, we can simply define the objective function as

$$L = \frac{1}{2}\|\vec{w}\|^2 \quad \text{(8)}$$

With the constraints from Eq. 6, we need to introduce Lagrange multipliers to solve this convex programming problem. In this way, the objective function becomes

$$L = \frac{1}{2}\|\vec{w}\|^2 - \sum_i \alpha_i \left[y_i\left(\vec{x_i}^T\vec{w} + b\right) - 1\right] \quad \text{(9)}$$

Eq. 9 is also called the Lagrange primal function, $\alpha_i \geq 0\ \forall i$, and this becomes a dual problem. We set the partial derivative to be 0, and obtain

$$\vec{w} = \sum_i \alpha_i \vec{x_i} y_i \quad \text{(10)}$$

$$\sum_i \alpha_i y_i = 0 \quad \text{(11)}$$

By substituting Eqs. 10 and 11 into Eq. 9, we obtain the Lagrangian dual function

$$L = \sum_i \alpha_i - \frac{1}{2}\sum\sum \alpha_i \alpha_j y_i y_j \vec{x_i}^T \cdot \vec{x_j} \quad \text{(12)}$$

subject to $\alpha_i \geq 0\ \forall i, \sum_i \alpha_i y_i = 0$

Eq. 12 indicates that the Lagrangian function depends solely on the pair of sample points when Lagrange multipliers are fixed. This gives us the general algorithm for a SVM classifier:

Algorithm 2 Support Vector Machine

Repeat till convergence {

1. Select some pair α_i and α_j to update next (using a heuristic that tries to pick the two that will allow us to make the biggest progress towards the global maximum)
2. Reoptimize L(α) with respect to α_i and α_j, while holding all the other α_k's $(k \neq i, j)$ fixed.

}

Using Algorithm 2, we can specify the suitable α_i for each sample i. Once the Lagrange multipliers are fixed, the plane with maximal margin is identified. Then we can use Eqs. 5 and 10 together to determine the label of an unknown point $\vec{u}$ as shown in Eq. 13.

$$\sum_i \alpha_i \overrightarrow{x_i} y_i \cdot \vec{u} + b > 0 \quad \text{.....(13)}$$

The procedure we have outlined is the process of determining a separating plane for a separable sample set, which is also called hard margin. As for a non-separable sample set, we will create a soft margin that can tolerate misclassified samples. **Fig. 16** illustrates the differences between hard margin and the soft margin.

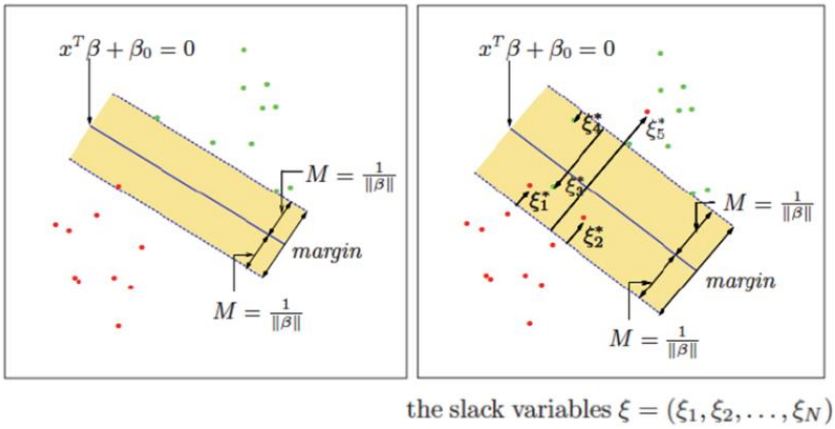


Fig. 16— Hard margin and soft margin

In Fig. 16, the left side shows the hard margin schematically, while the right side indicates the situation of soft margin where the data sets are non-separable. Several samples are misclassified in soft margin (e.g., the green dots and red dots in the margin area in the graph on the right of in **Fig. 16**).

The process of equation deduction for the soft margin is similar, and I briefly introduce it below. When we add a penalty parameter C of the error term to the Lagrange primal function, the Lagrange primal function becomes

$$\boldsymbol{L = \frac{1}{2}\|\vec{w}\|^2 + C\sum_{i=1}^{N}\xi_i - \sum_{i=1}^{N}\alpha_i\left[y_i\left(\vec{x_i}^T\vec{w} + b\right) - (1 - \xi_i)\right] -}$$

$$\boldsymbol{\sum_{i=1}^{N}\mu_i\xi_i}\ldots\ldots\ldots\ldots(14)$$

subject to $\xi_i \geq 0, y_i\left(\vec{x_i}^T\vec{w} + b\right) \geq 1 - \xi_i\ \forall i.$

As we solve this quadratic problem, we can easily obtain the Lagrange dual function

$$L = \sum_i \alpha_i - \frac{1}{2}\sum\sum \alpha_i \alpha_j y_i y_j \overrightarrow{x_i}^T \cdot \overrightarrow{x_j} \quad (15)$$

subject to $\alpha_i \geq 0 \; \forall i, \sum_i \alpha_i y_i = 0$

The KKT conditions for the soft margin are

$$\alpha_i \left[y_i \left(\overrightarrow{x_i}^T \vec{w} + b \right) - (1 - \xi_i) \right] = 0$$

$$\mu_i \xi_i = 0$$

$$y_i \left(\overrightarrow{x_i}^T \vec{w} + b \right) - (1 - \xi_i) \geq 0$$

We can see that the penalty term C limits the range of α_i; this is how the C value affects the final accuracy. The SVM classifier was implemented using the SVC classifier from the scikit-learn python package (SVC, scikit-learn 2017).

In this project, we have eight types at total. The data sets in this project are non-separable. This is indicated in **Fig. 17**. In Fig 17, the C value is set to be 1 and we still cannot achieve the 100% accuracy for any one of the four cases. This is the indication of the non-separable dataset. We could also see the accuracy increasing trend as the data sample has more production history. In NNet, we also observed a similar increasing trend.

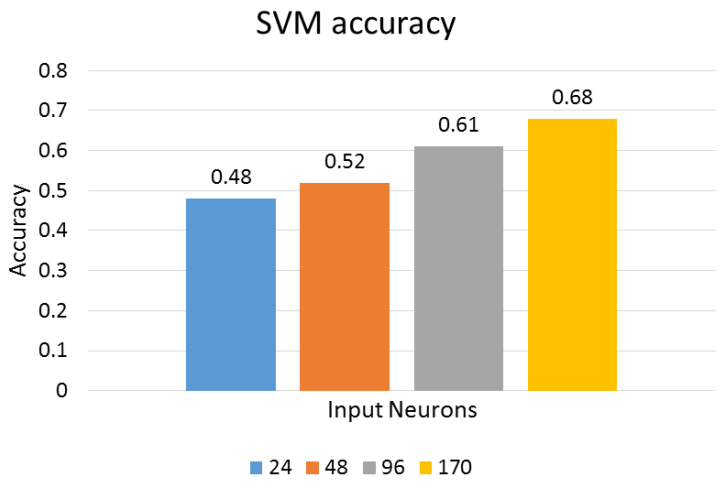


Fig. 17— Different input sample dimension comparison

Fig. 17 shows the accuracy for the test dataset. The accuracy for the test dataset with 170 months' production history is 68%. Thus, we can infer that given a new well production data with a production history of 170 months, the confidence of classifying this well into correct type is 68% when using support vector machine with parameters setting mentioned above. Compared to the test accuracy of 97% for NNet, we can see a disadvantage of SVM. The low accuracy may be induced by the fact that this SVM classifier needs to deal with a multi-type classification problem with a non-separable dataset in this application.

While the accuracy for the wells with only 24 months' production is the same for the two algorithms, the NNet is superior to SVM for 48 and 96 months of production history. As we have stated in section 2.5 Labeling, we matched the eight types with a corresponding EUR range. Thus, once the type is determined, further EUR range

estimation is obvious and oil and gas companies can make appropriate financial decisions.

2.8.3 RF

RF is an ensemble approach that adopts the idea of divide-and-conquer to improve performance. The guiding principle of RF is that a group of "weak learners" can come together to form a "strong learner". RF starts with a standard machine learning technique, classification and regression tree (CART), which, in ensemble terms, corresponds to the weak learner. **Fig. 18** (Simafore 2017) shows the different types of trees at each level and their usages.

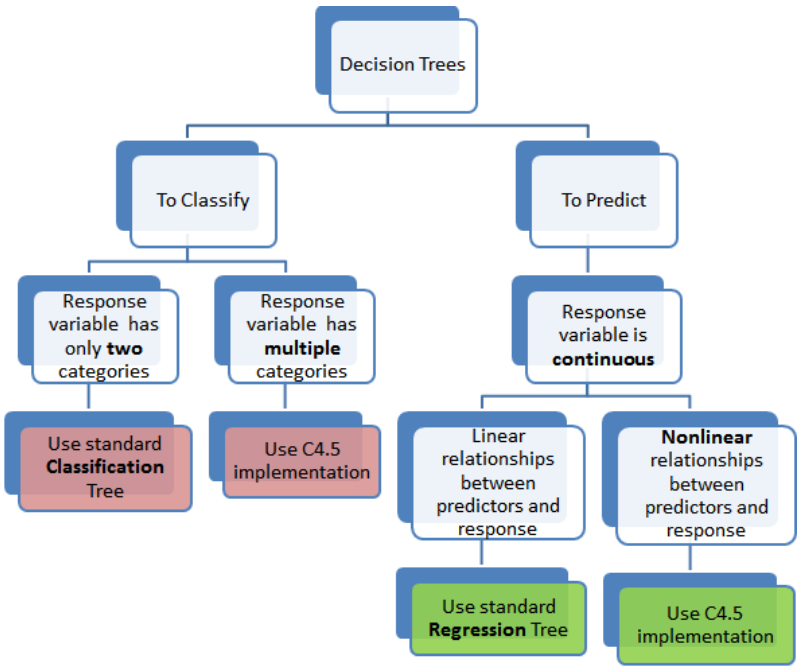


Fig. 18— Classification and regression tree

In our classification problem, we compared two criterion to split the nodes: gini index and entropy gain in the cross-validation process. The comparison is based on 170 months of production history. The results are shown in **Fig. 19**. In Fig. 19, we can see that the test accuracy difference of the two cases varies. The accuracy of the Gini index splitting criterion is always higher than that of entropy gain in both cases. So in our project, we implemented the RF algorithm based on the Gini index to split the nodes.

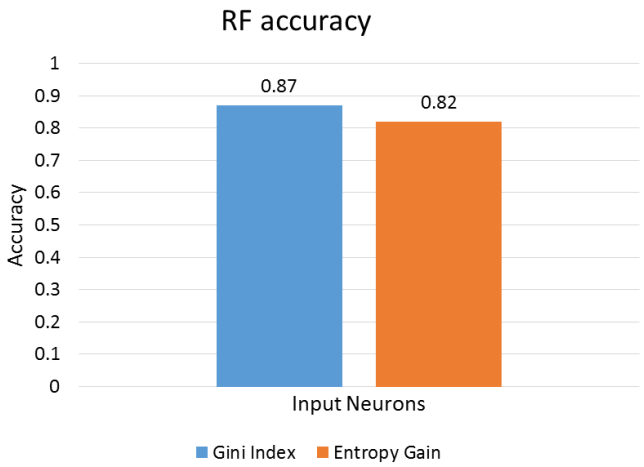


Fig. 19— RF accuracy comparison: Gini index and entropy gain

The decision trees are recursively built following a top-down approach by repeated splits of the training dataset. When decision trees work with continuous numerical values, the binary splits are usually performed by choosing the threshold which minimize the impurity measure used as splitting criterion (Berzal et. al 2003). **Fig. 20** (Saedsayad 2018) is an example of leveraging the decision tree to predict whether or not to play golf.

The core algorithm in this graph is ID3 which uses entropy and information gain as the criterion to split the tree nodes when constructing the decision tree.

Fig. 20— Example of decision tree

The RF algorithm addresses the overfitting problem that often arises in the decision tree algorithm. The algorithm generates a number of decision trees. Each decision tree is constructed based on a subset of training samples (randomly selected with replacement) and a subset of features (randomly selected without replacement). These combined decision trees decide the final classification type through majority vote.

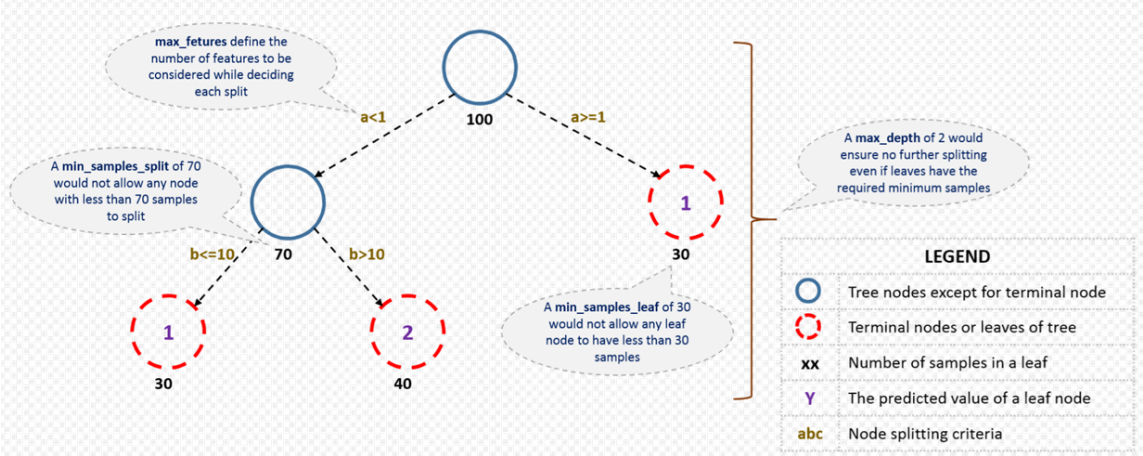


Fig. 21— Structure terms of random forest

RF can be used both in classification and regression problems (CART). In classification, we use entropy gain as the criterion to split the nodes. The number of decision trees, the maximum tree depth and maximum features are set to be 100, 20 and 50, respectively. Readers can refer to **Fig. 21** (Analytics Vidhya 2018) and scikit-learn (RandomForestClassifier, scikit-learn 2017) for the definition of these parameters.

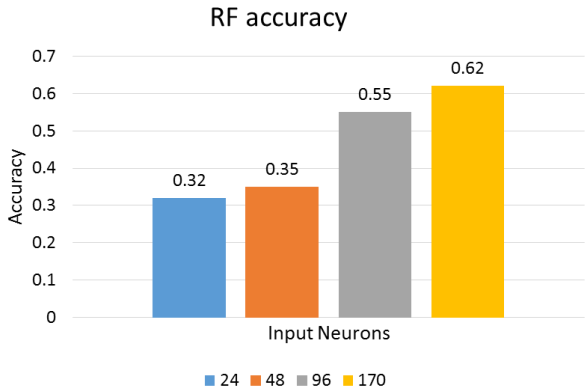


Fig. 22— Random Forest algorithm application results

The RF classification results are shown in **Fig. 22**. In Fig. 22, we see that the test accuracy in the cross validation process can be 62% for the data with 170 months of production history. Recall the results that we obtained from our neural networks (NNet) and support vector machine (SVM) applications: the test accuracy of NNet is 0.97, and the test accuracy of SVM is 0.68. The accuracy of RF is less than the two other algorithms. As one well with a short production history is classified into one of the 4 types by our RF algorithm, we have only 62% percent confidence that this classification was implemented correctly.

2.9 Machine Algorithm Robustness Analysis

As shown in section 2.8 Machine Learning Algorithm Applications, the three machine learning algorithms did not achieve 100% test accuracies. We dug deeper into the failed cases, and looked at the robustness for the three algorithms. The robustness is defined as the total distance of misclassified types to the correct types. For each failed case, we measured distance of the predicted type to the correct type, and use the overall distance as the criterion to evaluate the robustness of the algorithms. **Fig. 23**, **Fig. 24** and **Fig. 25** showed the robustness analysis for the three algorithms. In the three figures below, the horizontal axis is the distance of the predicted types to the correct types. +1/-1 Missed Type means that the algorithm misclassified sample as the type that is either one type above or one type below. This principle is similar for +2/-2 Missed Type and +3/-3 Missed Type. The vertical axis is the percentage for each missed type range. The columns for each color belongs to the same length of production history (e.g., blue columns correspond to the case with only 24 months' production history). The total

percentage for each color columns is 1. It worth noting that the total number of misclassified samples may not be the same for each color, we are measuring the percentage of misclassified samples that are misclassified into the types with different distance to the correct type.

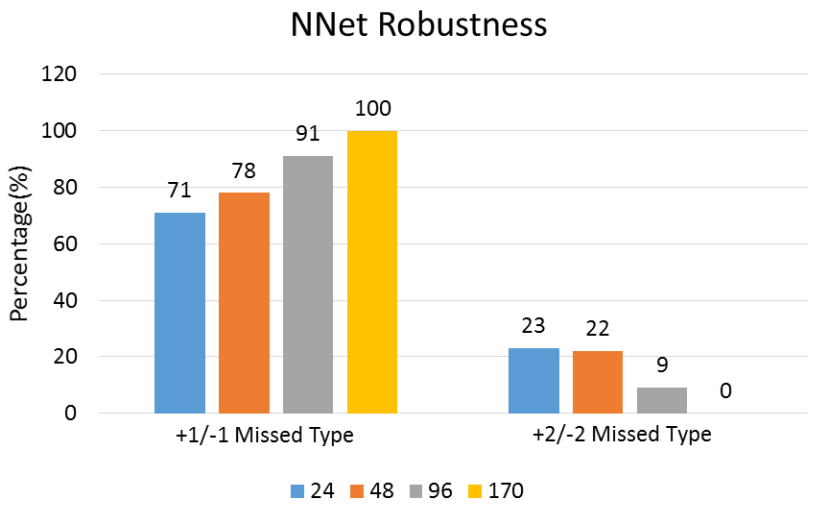


Fig. 23— NNet Robustness Analysis Result

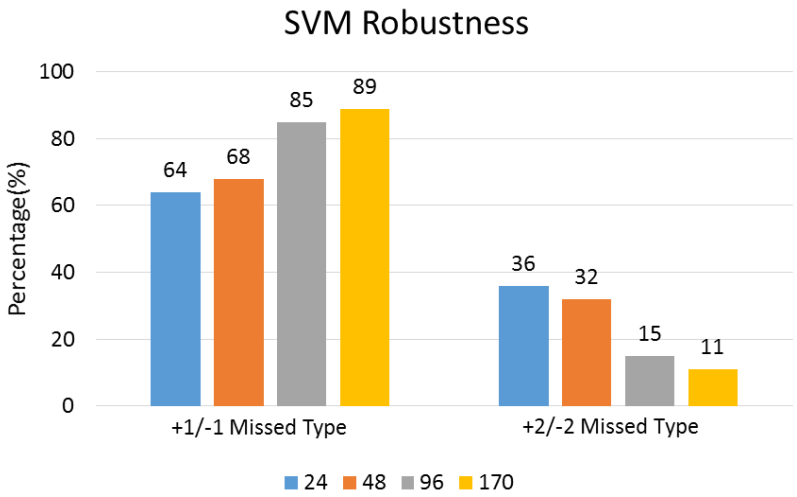


Fig. 24— SVM Robustness Analysis Result

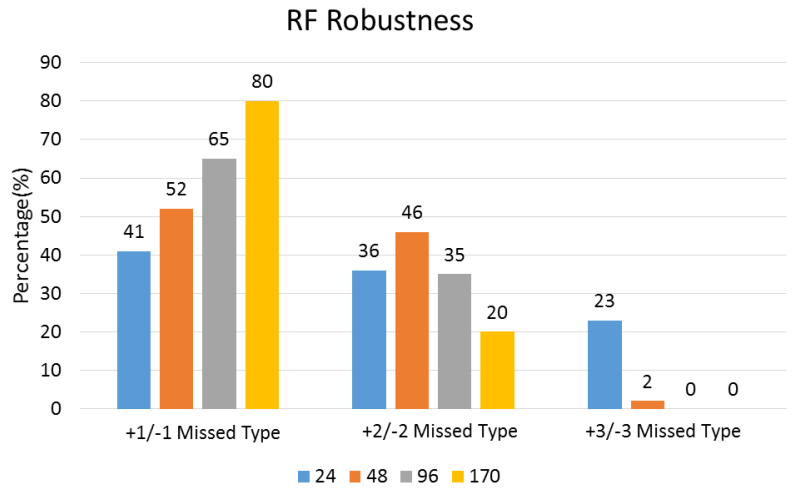


Fig. 25— RF Robustness Analysis Result

From Fig. 23, Fig. 24, and Fig. 25, we could easily see that for the +1/-1 Missed Type, as the production history increases, more misclassified samples will be classified into the types that are one type away from the correct one. This means that the performance of

the algorithm would be more reliable since more portion of the misclassified would be within one type distance to the correct one.

Among the three algorithms, NNet is the most reliable one since in NNet more misclassified samples are classified into the +1/-1 Missed Type rather than +2/-2 Missed Type or even +3/-3 Missed Type. RF performs worst in robustness. +1/-1 Missed Type percentage is the lowest in RF. In addition, there are some samples misclassified into the +3/-3 Missed Type category.

2.10 Summary

The Results and Discussion section explained the implementation details and corresponding results in this project. Initially, we preprocessed production data to obtain production profiles for 200 wells from the same geologic area, the Barnett Shale. We used BestFit in ValNav to forecast production for each well, and chose the cumulative production at 360 months to represent the EUR for each well. From our plot of the "Less Than Probability" versus sorted EUR on log probability paper, we identified a lognormal distribution of the 200 EUR values. Since the EUR values for all 200 wells were known, the P90, P50 and P10 values were found to be 1538 MMscf, 2448 MMscf, and 3759 MMScf, respectively.

We separated the 200 samples into 8 types, and the labels for all wells are their corresponding types. After maintaining the input sample feature dimensions of 24, 48, 96 and 170, we used the data to train classifiers with three machine learning algorithms:

neural networks (NNet), support vector machine (SVM), and random forest (RF). The cross-validation technique was also used to reduce the generalization error of the trained classifiers. The details of the three algorithms were also introduced.

In NNet, the min-max normalization technique was adopted to preprocess the input into neural networks. The test accuracy of NNet with 170 months' data was 0.97, which is the highest among the three. The number of neurons in the hidden layer was determined to be 167 to achieve the highest accuracy. For SVM, the non-separable property for our problem was identified by using the test results of C value at 1.0. Given the data with 170 months' production, the test accuracy of SVM was found to be 0.68, and the test accuracy of RF was 0.62, which is the lowest among the three algorithms. In RF, the results of two splitting criterion (Gini Index and Entropy Gain) were compared. To summarize, the NNet achieved the greatest accuracy, and it is thus the most suitable algorithm for application in this project.

Robustness analysis was implemented. The robustness of the algorithm is defined as the total distance of misclassified types to the correct types. Less total distance corresponds to more reliable and more stable performance for each individual algorithm. The NNet gives more robust performance with 100% misclassified samples classified into the types within one type distance to the correct types. RF is least robust. As the production history increases, the robustness of the three algorithms increases.

3 CONCLUSIONS

We forecasted production profile for 200 Barnett Shale gas wells using the modified Arps hyperbolic decline model, and found that the EUR values for wells in this data set followed a lognormal distribution, with a variability (P10/P90 ratio) of 2.32, indicating a highly consistent data set with minimal dispersion.

We successfully used three machine learning algorithms, Neural Networks, Support Vector Machine, and Random Forest to forecast EURs for wells with only limited production histories, following training the algorithms with the EUR values information.

After the model training process on the training dataset and the validation dataset using the cross validation technique, we get the test dataset accuracy at 0.97 for neural network algorithm provided data with 170 months' production, which indicates a 97% confidence when classify a well into one of the eight EUR range types we have got. As we have longer production history, the test accuracy increases from 0.48 for 24 months' available production history to 0.97 for 170 months' production history. Longer production history indicates more prior production information, which is helpful for the machine learning algorithms to know more about the data before they are implementing the classifications. The performance of NNet was compared with that of SVM and RF, and NNet was found to achieve the highest accuracy.

The robustness of the algorithm is defined as the total distance of misclassified types to the correct types. Less total distance corresponds to more reliable and more stable performance for each individual algorithm. The NNet gives more robust performance with 100% misclassified samples classified into the types within one type distance to the correct types. RF is least robust. As the production history increases, the robustness of the three algorithms increases.

NOMENCLATURE

a	Production Rate Parameter, Mscf/month$^{1/2}$
b	Arp's b parameter
C	Penalty Parameter
d	Displacement of the Hyperplane
D	Decline Rate, %/year
D_i	Initial Decline Rate, %/year
D_s	Decline Rate limit, %/year
L	Objective Function in SVM Problem
m	Slope of Rate Versus Material Balance Time Plot, Mscf
max	Maximum Feature Value
min	Minimum Feature Value
N_p	Cumulative Production, Mscf
q	Production Rate, Mscf/month
q_i	Initial Production Rate, Mscf/month
t	Production time, month
t_m	Material Balance Time, month
$\vec{u}$	Unknown Point in SVM Problem
val	Normalized Percentage of Feature Value
$value$	Feature Value Before Normalization
$\vec{w}$	Weight Vector

$width$	Margin of the Hyperplane
$\overrightarrow{x_i}$	Sample Point in SVM Problem
$\overrightarrow{x^+}$	Sample Point with Positive Target Value
$\overrightarrow{x^=}$	Sample Point with Negative Target Value
y_i	Target Value of the Sample Point
α_i	Lagrange multiplier
μ_i	Lagrange Primal Function Parameter
ξ_i	Lagrange Primal Function Parameter

REFERENCES

Ali, J.K. 1994. Neural Networks: A New Tool for the Petroleum Industry? . Presented at the European Petroleum Computer Conference, Aberdeen, United Kingdom, 15—17 March. SPE-27561-MS. http://dx.doi.org/10.2118/27561-MS.

Analytics Vidhya 2018 https://www.analyticsvidhya.com/blog/2016/04/complete-tutorial-tree-based-modeling-scratch-in-python/

Anifowose, F., Ewenla, A., and Eludiora, S. 2012. Prediction of Oil and Gas Reservoir Properties Using Support Vector Machines. Presented at the International Petroleum Technology Conference, Bangkok, Thailand, 7—9 February. IPTC-14514. http://dx.doi.org/10.2118/IPTC-14514.

Arps, J.J. 1945. Analysis of Decline Curves. SPE Journal 160 (1). SPE-45228. http://dx.doi.org/10.2118/45228.

Aulia, A., Rahman, A., and Velasco, J.J.Q. 2014. Strategic Well Test Planning Using Random Forest. Presented at the SPE Intelligent Energy Conference and Exhibition, Utrecht, Netherlands, 1—3 April. SPE-167827. http://dx.doi.org/10.2118/167827.

Berzal F., Cubero JC., Marín N., Sánchez D. (2003) Numerical Attributes in Decision Trees: A Hierarchical Approach. In: R. Berthold M., Lenz HJ., Bradley E., Kruse R., Borgelt C. (eds) Advances in Intelligent Data Analysis V. IDA 2003. Lecture Notes in Computer Science, vol 2810. Springer, Berlin, Heidelberg

BP 2012. AAPG, http://www.searchanddiscovery.com/documents/2012/80217cander/ndx_cander.

Crnkovic-Friis, L. and Erlandson, M. 2015. Geology Driven EUR Prediction Using Deep Learning. Presented at the SPE Annual Technical Conference and Exhibition, Houston, Texas, 28—30 September. SPE-174799. http://dx.doi.org/10.2118/174799.

DrillingInfo 2017 https://info.drillinginfo.com

Duong, A.N., 2011, Rate-decline analysis for fracture-dominated shale reservoirs, SPE Reserves Evaluation Engineering, 14(03) (2011), pp.377-387, 10.2118/137748-PA

Hedge, C., Wallace, S., and Gray, K. 2015. Using Trees, Bagging, and Random Forest to Predict Rate of Penetration During Drilling. Presented at the SPE Middle East Intelligent Oil&Gas Conference & Exhibition, Abu Dhabi, United Arab Emirates, 15—16 September. SPE-176792. http://dx.doi.org/10.2118/176792.

Holditch, S.A. 2003. The Increasing Role of Unconventional Reservoirs in the Future of the Oil and Gas Business. Journal of Petroleum Technology 55 (11): 34—79.

IIK, D., Rushing, J.A., Perego, A.D., Blasingame, T.A., Exponential vs. Hyperbolic decline in tight gas sands: understanding the origin and implications for reserve estimate using Arp's decline curves, Presented at the SPE Annual Technical Conference and Exhibition, 21-24 September (2008, January 1), 10.2118/116731-MS, Denver, Colorado, USA

Jha, H.S. and Lee, W.J. 2017. Problems with Application of Material Balance Time to Transient Flow Data in Diagnostic Plots. Presented at the Unconventional Resources Technology Conference, Austin, Texas, 24—26 July. SPE-2697627. http://dx.doi.org/10.2118/2697627.

Jia, X. and Zhang, F. 2016. Applying Data-Driven Method to Production Decline Analysis and Forecasting. Presented at the SPE Annual Technical Conference and

Exhibition, Dubai, United Arab Emirates, 26—28 September. SPE-181616-MS. http://dx.doi.org/10.2118/181616-MS.

Liu, Y. and Horne, R.N. 2011. Interpreting Pressure and Flow Rate Data from Permanent Downhole Gauges Using Data Mining Approaches. Presented at the SPE Annual Technical Conference and Exhibition, Denver, Colorado, 30 October—2 November. SPE-147298-MS. http://dx.doi.org/10.2118/147298-MS.

Liu, Y. and Horne, R.N. 2013. Interpreting Pressure and Flow Rate Data from Permanent Downhole Gauges with Convolution-Kernel-Based Data Mining. Presented at the SPE Annual Technical Conference and Exhibition, New Orleans, Louisiana, 30 September—2 October. SPE-166440-MS. http://dx.doi.org/10.2118/166440-MS.

Long, D.R. and Davis, M.J. 1987. A New Approach To the Hyperbolic Curve. Presented at the SPE Production Operations Symposium, Oklahoma City, Oklahoma, 8—10 March. SPE-16237. http://dx.doi.org/10.2118/16237.

MLPClassifier, scikit-learn 2017 http://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html

Palacio, J.C. and Blasingame, T.A. 1993. Decline-Curve Analysis with Type Curves - Analysis of Gas Well Production Data. Presented at the Low Permeability Reservoir Symposium, Denver, Colorado, 26—28 April. SPE-25909. http://dx.doi.org/10.2118/25909.

Ramgulam, A., Ertekin, T., and Flemings, P.B. 2007. An Artificial Neural Network Utility for the Optimization of History Matching Process. Presented at the Latin American&Caribbean Petroleum Engineering Conference, Buenos Aires, Argentina, 15—18 April. SPE-107468-MS. http://dx.doi.org/10.2118/107468-MS.

RandomForestClassifier, scikit-learn 2017 http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

Saedsayad 2018 http://www.saedsayad.com/decision_tree.htm

Seshadri, J. and Mattar, L. 2010. Comparison of Power Law and Modified Hyperbolic Decline Methods. Presented at the Canadian Unconventional Resources&International Petroleum Conference, Alberta, Canada, 19—21 October. SPE-137320. http://dx.doi.org/10.2118/137320.

Sharma, A. and Lee, W.J. 2016. Improved Workflow for EUR Prediction in Unconventional Reservoirs. Presented at the Unconventional Resources Technology Conference, San Antonio, Texas, 1—3 August. SPE-2444280. http://dx.doi.org/10.2118/2444280.

SimaFore 2017 http://www.simafore.com/blog/bid/62482/2-main-differences-between-classification-and-regression-trees

SVC, scikit-learn 2017 http://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html

Valko, P. P., & Lee, W. J. (2010, January 1). A Better Way To Forecast Production From Unconventional Gas Wells. Society of Petroleum Engineers. doi:10.2118/134231-MS

ValNav 2017 http://www.energynavigator.com/value-navigator

Wikipedia 2018 https://en.wikipedia.org/wiki/Early_stopping

Xie, X., Fairbanks, M.D., Fox, K.S. et al. 2012. A New Method for Earlier and More Accurate EUR Prediction of Haynesville Shale Gas Wells. Presented at the SPE Annual

Technical Conference and Exhibition held in San Antonio, San Antonio, Texas, 8—10 October. SPE-159273. http://dx.doi.org/10.2118/159273.

Zhao, B., Zhou H., Li, X., and Han, D., 2006. Water Saturation Estimation Using Support Vector Machine. SEG Annual Meeting, New Orleans, Louisiana. SEG 2006-1693.

Zhao, T., Jayaram, V., Marfurt, K., and Zhou., H. 2014. Lithofacies classification in Barnett Shale Using Proximal Support Vector Machines, Danver, Colorado. SEG 2014-1491