

VIETNAM OIL AND GAS GROUP
PETROVIETNAM UNIVERSITY



GRADUATE THESIS

**APPLIED MACHINE LEARNING AND DEEP
LEARNING TO PREDICT PRODUCTION RATE**

Student name : Luong Khanh Loc
Major : Petroleum Engineering
Student ID : 04PET110009
Class : K4KKT
Course : 2014-2019
Supervisor : Dr. Nguyen Van Hung
Co-supervisor : Msc. Luong Hai Linh

Ba Ria - Vung Tau, August 5, 2019

**THE THESIS IS COMPLETED AT
PETROVIETNAM UNIVERSITY**

Supervisor:

Co-supervisor:.....

Thesis examiner:.....

The thesis is defended at:

**THESIS COUNCIL
PETROVIETNAM UNIVERSITY**

Month Date Year

THESIS ASSIGNMENT FORM

Student name: Luong Khanh Loc

Student ID: 04PET110009

Major: Petroleum engineering.

Class: K4KKT

1. Thesis title: Applied machine learning and deep learning to predict production rate.

2. Assignment:

- Study on the theory of production forecasting from conventional decline curve analysis, type curve analysis to reservoir modeling.
- Study on the data cleaning and analysis of production well.
- Building machine learning and deep learning models in predicting production rate.
- Implementation of both conventional and modern methods in excel and python.
- Optimizing several hyperparameters on machine learning and deep learning models in finding the best combination for prediction.

3. Starting date: 20/03/2019

4. Finishing date: 17/07/2019

5. Supervisor's name:

- Dr. Nguyen Van Hung
- MSc. Luong Hai Linh

Ba Ria - Vung Tau, August 5, 2019

RECTOR

**HEAD OF OFFICE OF
ACADEMIC AFFAIRS**

**DEAN OF PETROLEUM
FACULTY**

THESIS EVALUATION FORM

Thesis title: Applied machine learning and deep learning to predict production rate.

Student name: Luong Khanh Loc.

Major: Petroleum engineering

Course: 2014 - 2019.

Thesis examiner's name:

1. Comment on the student's attitude of work and study:

.....
.....
.....
.....
.....
.....
.....

2. Comment on the result:

.....
.....
.....
.....
.....
.....

3. Defects (If any):

.....
.....
.....

.....
.....

4. Grade:

Ba Ria - Vung Tau, August 5, 2019

SUPERVISOR

THESIS EVALUATION FORM

Thesis title: Applied machine learning and deep learning to predict production rate.

Student name: Luong Khanh Loc.

Major: Petroleum engineering

Course: 2014 - 2019.

Thesis examiner's name:

I. COMMENT SECTION

1. Form and structure of the thesis:

.....
.....
.....
.....
.....
.....

2. Contents:

2.1 Comment on references:

.....
.....
.....
.....
.....

2.2 Comment on the research method:

.....

.....

.....

.....

.....

2.3 Comment on the result:

.....

.....

.....

.....

.....

2.4 Comment on the conclusion:

.....

.....

.....

.....

.....

2.5 Defects of the thesis:

.....

.....

.....

.....

.....

II. GRADE:.....(In words)

Ba Ria - Vung Tau, August 5, 2019

THESIS EXAMINE

STATUTORY DECLARATION

I herewith formally declare that I myself wrote the submitted thesis independently and used plenty of sources mentioned at the end of this paper. I did not violate the intellectual property law and Vietnam law. I take full responsibility before the law of them..

AUTHOR

Luong Khanh Loc

ABSTRACT

APPLIED MACHINE LEARNING AND DEEP LEARNING TO PREDICT PRODUCTION RATE

Student name: Luong Khanh Loc

Course: 2014 - 2019

Major: Petroleum engineering.

Supervisor: Dr. Nguyen Van Hung

Class: K4KKT

Co-supervisor: MSc. Luong Hai Linh

Recently, artificial intelligent (AI) has gained a great deal of attention in many areas, especially the oil and gas industry. There was a significant development of AI that enables oil and gas companies to reduce a huge amount of cost and make more accurate decisions. In this paper, a new method based on Artificial Intelligence including machine learning and deep learning techniques has applied to petroleum engineering to predict oil and gas production.

Production rates are predicted using various algorithms based on AI technique. This study proposed a novel approach using regression algorithms and ensemble methods to leverage the results. A novel method of using recurrent neural networks include long short term memory (LSTM), gated recurrent unit (GRU) and simple recurrent neural networks (Simple RNN) has been developed to predict oil and gas production.

Long short term memory has been chosen to compare with conventional decline curve analysis and reservoir modeling method. The outcome showed that the development of long short term memory has outperformed DCA and give better result than eclipse model in some cases. This research focuses on figuring out the potential and efficiency of machine learning and deep learning algorithms to predict production in the future. This study explored twelve different machine/deep learning algorithms and their performance evaluation. Python is chosen for the implementation because of its useful and robust machine/deep learning libraries.

ACKNOWLEDGMENTS

First of all, I would like to express my gratitude and appreciate to all the professors and teachers and of Drilling and Production Department of PetroVietnam University who have supported me during my five-year learning at PetroVietnam university. I also thank Vietsovpetro Joint Venture, particularly Production Operations Department for giving me a precious internship which brought me a lot of experience and orientation for my graduate thesis.

My thanks especially go to Dr. Hung Nguyen Van who has given me immense and thoughtful knowledge and also detailed guidance. His kindly support and thorough advice went through the process of conducting my thesis had significantly enriched and improved my work. My co-supervisor, MSc. Luong Hai Linh has been extremely supportive and available whenever needed. She is always open to talk, very understanding and willing to help. I also want to thank MSc. Nguyen Viet Khoi Nguyen for the help with eclipse simulator and clear explanation of Eclipse files.

Additionally, I would like to express my sincere thanks to all the teachers and my friends at PVU who have giving their support during my study here. With them, my university life would be easier and meaningful. Finally, I am honored to have been a student at Petro-Vietnam University.

Once again, I would like to acknowledge and thank all of them who have helped me complete my graduate thesis.

Lastly, I want to dedicate this work for my families who have fully supported me during my university time.

Luong Khanh Loc

Contents

STATUTORY DECLARATION	i
ABSTRACT	ii
ACKNOWLEDGMENTS	iii
TABLE OF CONTENTS	vii
LIST OF TABLES	ix
LIST OF FIGURES	xiii
ABBREVIATIONS	xv
PREFACE	xvi
CHAPTER 1 Overview of production forecasting	2
1.1 History of production analysis[1]	2
1.2 Traditional decline curve methods[2]	4
1.3 Type curve methods.[2]	9
1.4 Reservoir modelling and forecast methods	11
1.4.1 Reservoir modelling [3]	12
1.4.2 History matching	13
1.4.2.1 Manual history matching [6]	14
1.4.2.2 Automatic history matching[3]	15

1.4.3	Prediction	16
CHAPTER 2	Overview of Volve field	18
2.1	Field description	18
2.1.1	Reservoir	19
2.2	Production profile	19
CHAPTER 3	Artificial Intelligence	23
3.1	Machine learning	24
3.1.1	Linear Regression	24
3.1.2	Ridge	25
3.1.3	KNeighbors Regression	25
3.1.4	Support Vector Regression	26
3.1.5	Decision Tree Regression	26
3.1.6	Random Forest Regression	27
3.1.7	Extra Trees Regression	27
3.1.8	Ada Boost Regression	28
3.1.9	Gradient Boosting Regression	28
3.1.10	Extreme Gradient Boosting Regression	29
3.1.11	Light Gradient Boosting Regression	29
3.2	Deep learning	30

3.2.1	Artificial neural networks	30
3.2.2	Deep feed-foward networks	31
3.2.3	Simple Recurrent neural networks	32
3.2.4	Long short term memory	33
3.2.5	Gated Recurrent Unit	33
CHAPTER 4	Methodology	35
4.1	Input data	35
4.2	Work-flow	36
4.2.1	Gather data	37
4.2.2	Data preprocessing	37
4.2.3	Cross validation strategies	38
4.2.4	Feature scaling	38
4.2.5	Machine learning and deep learning algorithms	39
CHAPTER 5	Result and Discussion	40
5.1	Machine learning	40
5.2	Deep learning	44
5.2.1	Neural networks	44
5.2.2	Recurrent Neural Networks	46
5.3	Applied long short term memory as the best algorithms	47

5.3.1	Comparison with DCA	48
5.3.2	Comparison with Eclipse	51
5.3.2.1	Well F1	51
5.3.2.2	Well F11	52
5.3.2.3	Well F12	53
5.3.2.4	Well F14	54
5.3.2.5	Well F15	55
5.3.2.6	Total cumulative oil production in comparison	59
CHAPTER 6	Conclusion	61
CHAPTER A	Preprocessing and data used in thesis	67
CHAPTER B	Hyperparametes optimization	71
CHAPTER C	Machine learning result	73
CHAPTER D	Results of LSTM on five well	78
CHAPTER E	Results of LSTM cumulative gas and water	81

List of Tables

1.1	Different values b versus three kinds of decline curve	7
1.2	Recomedation of b values	8
2.1	Production profile of Volvo filed.	21
4.1	Description of data set	35
5.1	Score and error of machine learning algorithms on oil	40
5.2	Pearson correlation of best five algorithms	41
5.3	Score and error on oil prediction	44
5.4	Score and error on gas prediction	45
5.5	Score and error on water prediction	45
5.6	Pearson correlation of neural networks	45
5.7	Decline (D) values and time periods	49
5.8	Cumulative oil production comparison	49
5.9	Cumulative oil production by well	58
5.10	Total cumulative oil production and errors	60
A.1	Sample data for applying machine learning and deep learning	70
B.1	Hyperparametes to optimize machine learning algorithms	71

B.2	Hyperparameters to optimize deep learning algorithms	72
C.1	Score and error of machine learning algorithms on gas production	73
C.2	Score and error of machine learning algorithms on water production . . .	73

List of Figures

1.1	Classification of production decline curves	6
1.2	Processing of using reservoir modeling	17
2.1	Location of Volve field	18
2.2	Reservoir model of Volve field	19
2.3	Production during the period 2008 to 2017	20
3.1	Shallow neural networks with one hidden layer	30
3.2	Deep feed forward neural networks	31
3.3	Recurrent neural network simple architecture[26]	32
3.4	Simple stretch of long short term memory[26]	33
3.5	Simple stretch of Gated Recurrent Unit[26]	34
4.1	Work flow for using machine and deep learning algorithms	36
4.2	Detailed work flow for using machine learning and deep learning algorithms	37
5.1	Production rate prediction versus actual values	42
5.2	Oil prediction using time series cross-validation method.	43
5.3	Gas prediction using time series cross-validation method.	43
5.4	Production rate prediction using shallow neural networks.	45

5.5	Production rate prediction using deep neural networks.	46
5.6	Production rate prediction using long short term memory.	47
5.7	Production rate prediction using gated recurrent unit.	47
5.8	Production rate prediction using simple recurrent neural networks.	47
5.9	History production of well F14	48
5.10	Comparison of actual values, LSTM and DCA	49
5.11	Production prediction on well F1	51
5.12	Production prediction on well F11	52
5.13	Production prediction on well F12	53
5.14	Production prediction on well F14	54
5.15	Production prediction on well F15	55
5.16	Cumulative oil production on F1	56
5.17	Cumulative oil production on F11	57
5.18	Cumulative oil production on F12	57
5.19	Cumulative oil production on F14	58
5.20	Cumulative oil production on F15	58
5.21	Total production by day from 2008 to 2016	59
5.22	Total cumulative oil production comparison	59
A.1	Data before feature engineering	67

A.2	Data after feature engineering	67
A.3	Box plot of input	67
A.4	Box plot of production rate	68
A.5	Correlation maxtrix of variables	69
C.1	Best five model on oil prediction	74
C.2	Best five model on gas prediction	74
C.3	Best five model on water predict	74
C.4	Water prediction using time series cross-validation method.	75
D.1	Production prediction on well F1	78
D.2	Production prediction on well F11	79
D.3	Production prediction on well F12	79
D.4	Production prediction on well F14	80
D.5	Production prediction on well F15	80
E.1	Cumulative gas prediction F1	81
E.2	Cumulative water prediction F1	81
E.3	Cumulative gas prediction F11	81
E.4	Cumulative water prediction F11	81
E.5	Cumulative gas prediction F12	81
E.6	Cumulative water prediction F12	81

E.7	Cumulative gas prediction F14	82
E.8	Cumulative water prediction F14	82
E.9	Cumulative gas prediction F15	82
E.10	Cumulative water prediction F15	82
E.11	Total oil production by day	82
E.12	Total cumulative oil production	83

Abbreviations

- w The learning parameters for machine learning algorithms
- X The input data for machine learning algorithms
- A The area of the needle point
- a The number of angels per unit area
- b Arps' decline curve exponent
- D Decline rate
- N The number of angels per needle point
- y Actual value of production rate
- AI Artificial Intelligent
- BHP Bottom Hole Pressure
- BHT Bottom Hole Temperature
- CS Choke Size (Percentage)
- DCA Decline Curve Analysis
- DP Different Pressure in casing
- GRU Gated Recurrent Unit
- LGBM Light Gradient Boosting Machine
- LSTM Long Short Term Memory
- MLP Multi Layers Perceptron

RNN Recurrent Neural Networks

SEPD Stretched Exponential Production Decline

THP Tubing Head Pressure

WHT Well Head Temperature

XGB Extreme Gradient Boosting

PREFACE

Rationale

Production forecasting is one of the most difficult tasks in petroleum engineering. Its purpose has been significantly important for making decision related to business development. There various way to predict future production includes traditional decline curve analysis (DCA), type curve and reservoir modeling. Recently, with the development of artificial intelligence, AI can improve their capacity in prediction. So, the potential and efficiency of AI technique have been studied in this work.

Objective

- Study on the theory of production forecasting from conventional decline curve analysis, type curve analysis to reservoir modeling.
- Study on the data cleaning and analysis of production well.
- Building machine learning and deep learning models in predicting production rate.
- Implementation of both conventional and modern methods in excel and python.
- Optimizing several hyperparameters on machine learning and deep learning models in finding the best combination for prediction.

Subject and scope

- Subject: Production prediction with decline curve analysis, reservoir modeling and artificial intelligence methods.
- Scope: This thesis is work on Volve field which is a decommissioned field located in North Sea, specifically five production wells are used to predict the future production rate.

Methodology

- Reading documents related to the oil and gas production forecasting, reservoir modeling and artificial intelligence methods applied in petroleum engineering.

- Building a work-flow for using machine learning and deep learning algorithms in production prediction and implementation in python library.
- Making comparison between three methods (conventional DCA, reservoir modeling and artificial intelligence) in order proving the potential of AI that can be an effective tools in oil and gas industry.

Significance

- Scientific significance: The thesis presents theories about production analysis, machine learning, and deep learning algorithms, especially reveal the great performance of artificial intelligence methods in term of production forecasting. Also, the thesis could possibly be a reference source for those who want to explore the potential of AI when applying to petroleum engineering.
- Practical significance: This is a piece of work that proves the novel methods of using artificial intelligence including machine and deep learning can be tools for engineering in production prediction.

Thesis structure

Chaper 1: Overview of production forecasting

- History of production analysis
- Traditional decline curve methods
- Type curve methods
- Reservoir modeling and forecast methods

Chapter 2: Overview of Volve field

- Field description
- Reservoir
- Production profile

Chapter 3: Artificial Intelligence

- Machine learning
- Deep learning

Chapter 4: Methodology

- Input data
- Work-flow

Chapter 5: Result and discussion

- Machine learning output
- Deep learning output
- Applied LSTM

Chapter 6: Conclusion

Appendix A. Preprocessing and data used in thesis

Appendix B. Hyperparameters optimization

Appendix C. Machine learning result

Appendix D. Results of LSTM on five well

Appendix E. Results of LSTM cumulative gas and water

Chapter 1. Overview of production forecasting

1.1. History of production analysis[1]

The Production Analysis started in 1920 with a pure motive to find the best decline function by which it is possible to predict the revenue of the production in the future on an empirical basis with no technical background in it. Then (Arps , 1945) formulated the constant pressure exponential, hyperbolic and harmonic rate decline. In 1960, Type-curves were first introduced, still assuming constant flowing pressure by Fetkovich. These type-curves have two families one related the transient flowing period and the other for the late boundary-dominated flow. In late 80s and early 90s, (Palacio & Blasingame, 1993) introduces the variable rate/variable pressure type-curves as a log-log plot of productivity index vs material balance time.

Currently, the production decline analysis technique consists of the conventional Arps (1945) method, classical Fetkovich (1980) type curve matching method, modern Palacio and Blasingame (1993) and Agarwal et al. (1998) type curve matching methods and FMB (1998) reservoir engineering method.

Extrapolating the characteristic trend of some variables of a well can be helpful for our jobs. As to a well, the simplest and the most easily available variable is its production. If the flow rate versus time or cumulative production curve is plotted and extrapolated, the ultimate cumulative production can be obtained. The trend or mathematical relations indicated by the entire rate history of a well can be used to forecast the production performance in the future, which is referred to as the conventional Arps (1945) decline curve analysis method. This method magnificently describes the production decline laws of well at a constant bottom hole flowing pressure (BHFP) and in the completely boundary-dominated flow period. The greatest advantage of this method is that formation parameters are not necessarily obtained. On the other hand, it is not suitable for data analysis from the

transient flow stage.

Arps type curve can only be used to analyze the data of a boundary-dominated flow period. Fetkovich (1980), on the basis of homogeneous bounded formation transient filtration theory, introduced the transient flow formula in well test analysis to the decline analysis, so that the Arps type curve is extended to the transient flow period prior to boundary-dominated flow, and the transient rate decline curve and the Arps rate decline equation are organically combined. In this way, the production decline laws and the effect of boundary are intuitively shown, and a set of relatively complete log–log production decline curve matching analysis method similar to well test analysis is developed. The greatest advantage of the method is its ability to reliably determine whether the production is in a transient flow period or in a boundary-dominated flow period.

Both Arps and Fetkovich methods assume that the BHFP is constant to analyze the production data without considering the change of gas pressure–volume–temperature (PVT) characteristics with pressure. Palacio and Blasingame (1993) introduced the pseudo-pressure normalized production ($q\Delta p_p$) and the material balance pseudo-time t_{ca} to develop the type curve, which considered the production at variable BHFP and the gas PVT changing with formation pressure.

Agarwal et al. (1998) used the relations of pseudo-pressure normalized production ($q\Delta p_p$), material balance pseudo-time t_{ca} , and dimensionless parameters in well test analysis to develop the Agarwal-Gardner production decline analysis. Owing to the different definitions of dimensionless quantity, the early part of the curve is more discrete than the Blasingame chart and thus is in favor of reducing the ambiguity of matching analysis.

Both Blasingame and Agarwal-Gardner methods used the pseudo-pressure normalized production ($q\Delta p_p$) and the material balance pseudo-time t_{ca} to create type curve, while the NPI (normalized pressure integral) method (Blasingame et al., 1989) used the production normalized pressure integral to analyze the data available, which was not affected by the scatter of data.

Palacio and Blasingame (1993) and Agarwal et al. (1998) type curve matching analysis

methods introduced pseudo-time (or material balance pseudo-time) and production normalized pseudo-pressure (pseudo-pressure normalized production) to deal with variable BHFP, variable rate, and change of gas PVT with pressure. They used the flow rate integral, flow rate integral derivative, cumulative production–time, and flow rate–cumulative production type curves as the auxiliary matching analysis curves to reduce the ambiguity of interpretation results.

Mattar et al. (1998, 2006) and Agarwal et al. (1998) suggested using the “flow (dynamic) material balance” method to analyze the production data, and conducted detailed discussion on the calculation of material balance time. This method is simple and easy. Mattar and Anderson (2003) believes that there is no one universal production data analysis method that can meet all types of reservoirs, and the best way to eliminate analysis errors is to synthetically use all analysis methods and consider flowing pressure data.

Over nearly a century, the Production Analysis technique has evolved with several advances, including target to be analyzed, that is, from purely production data to both flow rate and pressure data; analytic model, that is, from no model to both analytical model and numerical model; analytic method, that is, from the empirical Arps method to the log-log method represented by Blasingame; applicable conditions, that is, from simple constant pressure production data to variable pressure and variable rate data; and the estimation parameters, that is, from only cumulative production to many parameters such as formation permeability, skin factor, dynamic reserves and drainage area, as well as interwell communication and infill potential.

1.2. Traditional decline curve methods[2]

Decline curves are one of the most extensively used forms of data analysis employed in evaluating gas reserves and predicting future production. The decline-curve analysis technique is based on the assumption that past production trends and their controlling factors will continue in the future and, therefore, can be extrapolated and described by a mathematical expression.

The method of extrapolating a “trend” for the purpose of estimating future performance

must satisfy the condition that the factors that caused changes in past performance, for example, decline in the flow rate, will operate in the same way in the future. These decline curves are characterized by three factors:

- Initial production rate, or the rate at some particular time
- Curvature of the decline
- Rate of decline

Arps (1945) proposed that the “curvature” in the production-rate versus time curve can be expressed mathematically by a member of the hyperbolic family of equations. Arps recognized the following three types of rate-decline behavior:

- Exponential decline
- Harmonic decline
- Hyperbolic decline

Each type of decline curve has a different curvature, as shown in Figure 1-1. This figure depicts the characteristic shape of each type of decline when the flow rate is plotted versus time or versus cumulative production on Cartesian, semi log, and log-log scales. The main characteristics of these decline curves can be used to select the flow rate decline model that is appropriate for describing the rate–time relationship of the hydrocarbon system:

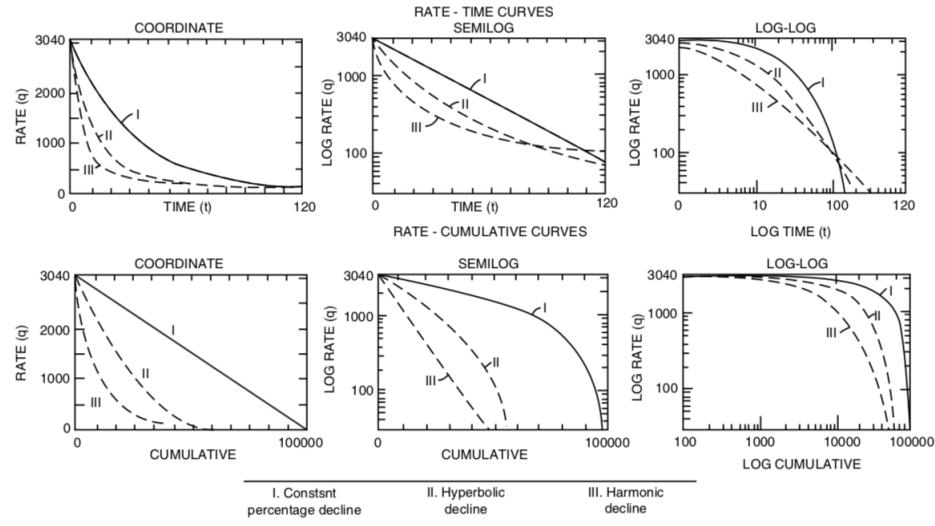


Figure 1.1. Classification of production decline curves

1. **Exponential decline:** A straight-line relationship will result when the flow rate versus time is plotted on a semi log scale and also when the flow rate versus cumulative production is plotted on a Cartesian scale.
2. **Harmonic decline:** Rate versus cumulative production is a straight line on a semi log scale; all other types of decline curves have some curvature. There are several shifting techniques that are designed to straighten out the curve that results from plotting flow rate versus time on a log-log scale.
3. **Hyperbolic decline:** None of the above plotting scales, that is, Cartesian, semi log, or log-log, will produce a straight-line relationship for a hyperbolic decline. However, if the flow rate is plotted versus time on log-log paper, the resulting curve can be straightened out with shifting techniques.

The decline-curve analysis is based on empirical relationships of production rate versus time, given by Arps (1945) as follows:

$$q_t = \frac{q_i}{(1 + bD_i t)^{\frac{1}{b}}} \quad (1.1)$$

where:

q_t is flow rate at time t, m³/day

q_i is initial flow rate, m³/day

t is time, day

D_i is initial decline rate, day⁻¹

b is Arps' decline curve exponent

The decline rate D is defined as the rate of change of the natural logarithm of the production rate, that is, $\ln(q)$, with respect to time, t

$$D = -\frac{d(\ln q)}{dt} = -\frac{1}{q} \frac{dq}{dt} \quad (1.2)$$

The minus sign has been added because dq and dt have opposite signs and it is convenient to have D always positive. Notice that the decline-rate equation, Equation 1-2, describes the instantaneous changes in the slope of the curvature, dq/dt , with the change in the flow rate, q, over time.

The parameters determined from the classical fit of the historical data, namely the decline rate, D, and the exponent, b, can be used to predict future production. This type of decline-curve analysis can be applied to individual wells or the entire reservoir. The accuracy of the entire reservoir application is sometimes even better than for individual wells due to smoothing of the rate data. Based on the type of rate-decline behavior of the hydrocarbon system, the value of b ranges from 0 to 1, and, accordingly, Arps's equation can be conveniently expressed in the following three forms:

Table 1.1. Different values b versus three kinds of decline curve

Case	b	Rate-time relationship
Exponential	$b = 0$	$q_t = q_i \exp(-D_i t)$
Hyperbolic	$0 < b < 1$	$q_t = \frac{q_i}{(1+bD_i t)^{\frac{1}{b}}}$
Harmonic	$b = 1$	$q_t = \frac{q_i}{(1+bD_i t)}$

A recommendation for b value is given by the table below:

Table 1.2. Recomedation of b values

b values	System Characterization and Identification
	Gas wells undergoing liquid loading
	Wells with high back pressure
	High-pressure gas
0	Low-pressure gas with a back-pressure curve exponent of $n \sim 0.5$
	Poor waterflood performance (oil wells)
	Gravity drainage with no solution gas (oil wells)
	Solution gas drive with unfavorable k_g/k_o (oil wells)
0.3	Typical for solution-gas-drive reservoirs
0.4 - 0.5	Typical for gas wells, $b = 0.5$ for $p_{wf} \sim 0$; $b = 0.4$ for $p_{wf} \sim 0.1 p_i$
0.5	Gravity drainage WITH solution gas and for water-drive oil reservoirs
Undeterminable	Constant-rate or increasing-rate production period Flow rates are all in transient or infinite-acting period
$0.5 < b < 0.9$	Layered or composite reservoir

It should be pointed out that these three forms of decline-curve equations are applicable only when the well/reservoir is under pseudosteady (semi steady)-state flow conditions, that is, boundary-dominated flow conditions. The following is a list of inherent assumptions that must be satisfied before performance of rate-time decline-curve analysis:

- The well is draining a constant drainage area, that is, the well is in a boundary-dominated flow condition.
- The well is produced at or near capacity.
- The well is produced at a constant bottom-hole pressure.

These three conditions must be satisfied before any of the decline-curve analysis meth-

ods is applied to describe the production performance of a reservoir. However, it can be extremely difficult to determine when a well has defined its drainage area and thus identification of the peseudo steady state flow condition. It is the main obstacle of using Arps's decline curve.

1.3. Type curve methods.[2]

The type-curve analysis approach was introduced to the petroleum industry by Agarwal et al. (1970) as a valuable tool when used in conjunction with conventional semi log plots. A type curve is a graphical representation of the theoretical solutions to flow equations. Type-curve analysis consists of finding the theoretical type curve that “matches” the actual response from a test well and the reservoir when subjected to changes in production rates or pressures. The match can be found graphically by physical superposition of a graph of actual test data on a similar graph of type curve(s) and searching for the type curve that provides the best match. Since type curves are plots of theoretical solutions to transient and pseudosteady-state flow equations, they are usually presented in terms of dimensionless variables, for example:

- dimensionless pressure, p_D
- dimensionless time, t_D
- dimensionless radius, r_D , and
- imensionless wellbore storage, C.

rather than real variables (e.g., Δp , t, r, and C). The reservoir and well parameters, such as permeability and skin, can then be calculated from the dimensionless parameters defining that type curve.

The variable can be made “dimensionless” when multiplied by a group of constants with opposite dimensions, but the choice of this group will depend on the type of problem to be solved. For example, to create the dimensionless pressure drop, p_D , the actual pressure drop Δp in psi is multiplied by group A with units of psi^{-1} or $p_\Delta = A\Delta p$. Finding a group A that makes a variable dimensionless is derived from equations that describe reservoir

fluid flow.

A well-known Darcy's equation is expressed by:

$$Q = \left[\frac{kh}{141.2B\mu [\ln(r_e/r_{wa}) - 0.5]} \right] \Delta p \quad (1.3)$$

where r_{wa} is apparent (effective) wellbore radius, defined by: $r_{wa} = r_w e^{-s}$. Group A can be then defined by rearranging Darcy's equation as:

$$\ln\left(\frac{r_e}{r_{wa}}\right) - \frac{1}{2} = \left[\frac{kh}{141.2QB\mu} \right] \Delta p \quad (1.4)$$

So, p_D is defined by:

$$p_D = \left[\frac{kh}{141.2QB\mu} \right] \Delta p \quad (1.5)$$

Similarly, t_D is defined by:

$$t_D = \left[\frac{0.0002637k}{\phi\mu c_t r_w^2} \right] t \quad (1.6)$$

By using Ei function for solving diffusivity equation in reservoir:

$$p_{r,t} = p_i + \left[\frac{70.6QB\mu}{kh} \right] Ei\left[\frac{-948\phi\mu c_t r_w^2}{kt} \right] \quad (1.7)$$

Rewrite the equation above:

$$\frac{p_i - p_{r,t}}{\left[\frac{70.6QB\mu}{kh} \right]} = -\frac{1}{2} Ei\left[\frac{-(r/r_w)^2}{4 \frac{0.0002637kt}{\phi\mu c_t r_w^2}} \right] \quad (1.8)$$

From the definition of the dimensionless variables p_D, t_D , and r_D , this relation can be expressed in terms of these dimensionless variables:

$$p_D = -\frac{1}{2} Ei\left(\frac{-(r_D)^2}{4t_D} \right) \quad (1.9)$$

By applying this similar dimensionless variables to decline curve analysis, several investigators have employed the dimensionless-variables approach to determine reserves and to describe the recovery performance of hydrocarbon systems with time, notably the following:

1. Fetkovich (1980)

2. Carter (1985)
3. Palacio and Blasingame (1993)
4. Mattar and Anderson's Flowing Material Balance (2003)
5. Anash et al. (2000)
6. Decline-curve analysis for fractured reservoirs

All the methods are based on defining a set of decline-curve dimension- less variables that includes:

1. Decline-curve dimensionless rate, q_{Dd}
2. Decline-curve dimensionless cumulative production, Q_{Dd}
3. Decline-curve dimensionless time, t_{Dd}

The aforementioned methods were developed with the objective of providing the engineer with an additional convenient tool for estimating reserves and determining other reservoir properties for oil and gas wells using the available performance data. A review of these methods and their practical applications is given in the book of Reservoir Engineering Handbook by Tarek [2].

Recently, there are a significant software that cover all of this type curve analysis such as Fekete Rate Transient Analysis, Topaze NL-RTA by KAPPA or IHS-RTA by IHS Markit. In this thesis, author has chosen traditional decline curve analysis by Arps rather than type curve analysis for making comparison with novel methods of using Artificial intelligence.

1.4. Reservoir modelling and forecast methods

Today, reservoir modeling is an essential part of every field development and assessment because of its reliability and accuracy compare with other conventional methods. In this thesis, the author does not cover the theoretical knowledge of reservoir modeling instead,

by introducing some of the core function of reservoir modeling in order to compare with machine learning and deep learning methods that will cover in the next chapter.

1.4.1. Reservoir modelling [3]

In short words, the reservoir model is a computer model based on geological model. The purpose of the reservoir model is to help facilitate numerical calculations in an effective manner. To achieve this, a huge detailed geological model must go through a reduction in parameters to reduce simulation time, usually done by dividing the geological model into grid quantitative parameters from the statics geological model in addition to dynamic parameters. This makes the reservoir model a good base for consistent analysis of the hydrocarbon presents, also giving a good basis for further economical assessment. It is also used for production optimization through well placement and tertiary recovery techniques. Dynamic parameters are parameters that change during production history. These could be well rates, grid block pressure, phase permeability, PVT characteristics of fluid or liquid and gas saturation. Some of the dynamic parameters, like grid block pressure, can vary significantly during years, or even months, of production. Matching these parameters are very challenging, as one during production does not have data specific to one grid cell, but rather data only from production and injection wells which is a result of actions from many grid-block. Static parameters, like mineralogy and net to gross ratio, do not change during production history. These kinds of parameters should thus be easier to model, as they would have the same value during the entire history matching period.

With all these parameters that needs to be defined, data is acquired from several different sources in the petroleum industry. Each type of data has a certain spatial extent associated to it. A large-scale geophysical survey, like seismic survey, could provide us with rock-types, fault placement and in some cases porosity estimates. These data does not reveal much of the heterogeneity within a cubic meter of rocks. Through well tests one can get estimates of the product of permeability and skin or near well-bore effects of the reservoir, but as with geophysical data these data are a sum of actions across a large volume of reservoir, so does not explain the actual parameters in one region or in one grid cell. Further

data collection through well logs and core samples could give data like net to gross ratio, electric resistivity, hydrocarbon content and porosity. Applying these data to a reservoir model needs to be done with consideration to both spatial variance and measurement errors. With all this data incorporated in the model, a final quality check needs to be done in order to see whether the model is geological plausible. This requires a geological interpretation, and could be performed by seeing if porosity and permeability values are consistent with rock type [4].

By using commercial reservoir simulator, we can use the reservoir model to predict future production. The accuracy of the prediction is influenced by several uncertain factors. The sources of error include, among other, the reservoir model, simulator and flow physics. This prediction is often used as a basic for major reservoir development decisions and some important decision are based on the expected recovery factor from simulation.

1.4.2. *History matching*

History matching is one of the core activities performed by petroleum engineers to decrease the uncertainty of reservoir models. By comparing real data production data gathered at the surface, with the output from a reservoir simulator, the gaps in reservoir properties of those block cells in the model is filled.

The goal of history matching is to find a plausible model variable \mathbf{m} that minimize the square norm of the data mismatch:

$$J(m) = \frac{1}{2} \| g(\mathbf{m}) - \mathbf{d}_{obs} \|_D^2 \quad (1.10)$$

where $g(\mathbf{m})$ is the output of the model (typical output from simulated model) and \mathbf{d}_{obs} is the vector observed data.

In order to perform and achieve a history match of reservoir model some sort of historical performance data must exist. These data can be production rates of different phases, pressures or any other historic data describing reservoir performance. By constraining the wells in a simulation model with a chosen type of performance data and compare the calculated model response with historical data available, the reservoir model is able to predict

future actions of reservoir, production rate and recovery factor. Because of the uncertainty of the data, in order to get a reliable model that can be further foundation of making decision, reservoir model have to be fully understanding before running any history matching. For example, we may specify the model to be simulated with well constrained by oil rate. The model will calculate the corresponding rates of gas and water according to mobility and pressure, these in turn can be compared with measured rates. The history match will be validated based on how good it matches the historical reservoir performance. There is usually some sort of deviation between the behavior of the simulation model and the actual reservoir. This established the need to predetermine a criteria used to describe a successful match.

1.4.2.1. Manual history matching [6]

Before automatic history matching of realistic reservoir models was feasible, reservoir models were history matched manually using good engineering judgment and a work flow that had been developed through years of experience. There were a wide range of ways to match the output of model with actual data. A typical method is begins with an attempt to match pressures at the large (field) scale through adjustments of a few key parameters: aquifer pore volume factors, aquifer transmissibility, permeability multipliers, rock compressibility, and the ratio of vertical to horizontal permeability. When the pressure has been matched adequately at the field level, the properties of individual flow units or layers are adjusted, followed by the adjustment of well properties or properties of cells near wells. Once the pressure matching is complete, other variables are adjusted in an attempt to match water arrival times and water cut behavior. Williams et al[5] suggest that modification of the relative permeability curves is frequently useful for saturation matching because its effect on the pressure match may be relatively small. In some cases, it may be necessary to locally increase or decrease vertical transmissibility to change the flow of water from the lower layers to the upper layers in the reservoir.

1.4.2.2. Automatic history matching[3]

Automatics history matching (or assisted history matching) routines uses optimization algorithms to mathematically describe the observed mismatch and update the reservoir model. Such routines can be divided in two groups, deterministic and stochastic models. The deterministic models require calculations of Jacobian or Hessian matrices in order to approach the true solution. Stochastic models focuses on minimizing a objective function without the use of gradients. This is generally more computationally demanding because of the model parameter space being searched can be very large, but can generally be applied to non-linear cases where gradient calculations are very hard. Major challenges for both methods are the huge parameter sets, the non-linearity of reservoir simulation and several local minimas in the objective function.

Although the objectives of history matching are relatively consistent (minimization of the square data mismatch with regularization), the methods used for the minimization and for assessing uncertainty vary widely. During a past decade, there were a various implementation of algorithms in automatics history matching:

1. Gradient descent methods
2. Design of Experiments
3. Respone surface modeling
4. Parametric methods
5. Ensemble Kalman Filters
6. Artificial Intelligent methods (include ANN, GA, fuzzy logic...)
7. Hybric methods
8. Travel time inversion
9. Multiple solution

Different algorithms yield with different problems of history matching. Part of the reason for the differences is clearly that some methods are easier to implement than others, but it is also clear that different history matching problems may require different optimization algorithms. One consequence of the “no free lunch theorem for search and optimization is that “a general-purpose universal optimization strategy is theoretically impossible, and the only way one strategy can outperform another is if it is specialized to the specific problem under consideration”[6]. So, based on the kind of history matching problem, number of parameters, computational limit, convergence speed and other aspect, in some case one algorithms can be better than the others.

1.4.3. Prediction

The objective of history matching is to provide a model than can reasonably predict the future performance of the reservoir under various operational scenarios. Different operational scenarios may include well stimulations, drilling of new wells, abandonment of existing wells, injection of different fluids, and modification of surface facilities. These actions are intended to increase or accelerate hydrocarbon recovery and most will require significant capital investment. Hence, a facility for economic analysis is required in the comparison of alternative operational scenarios.

The development planning process is an exercise in optimization, in which the objective is to maximize economic metrics such as net present value (NPV) or ultimate recovery, while honoring the external constraints, such as physical (platform) limitations, available capital, drilling rigs, or injection gas. The nature of the objective function (i.e., NPV) must be clearly defined before the optimization stage, as it provides the means to compare different scenarios.

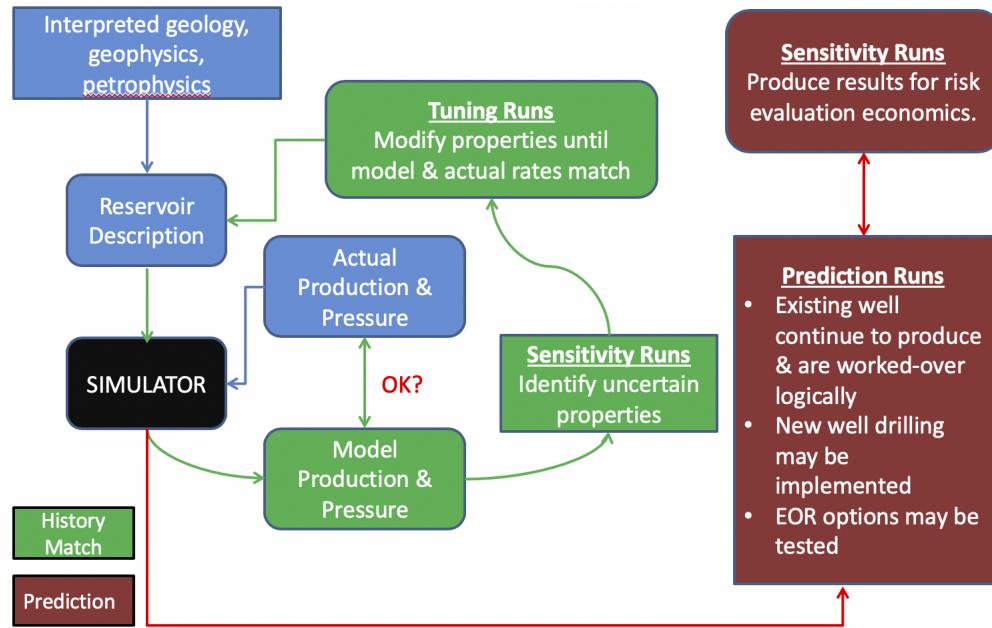


Figure 1.2. Processing of using reservoir modeling

Even when a reservoir model is calibrated (history-matched), there can be many parameters of reservoir or fluid interaction parameters that may not have been tested. So a question of whether a calibrated model can be verified to predict the future actions of reservoir?

Chapter 2. Overview of Volve field

2.1. Field description

Volve is a decommissioned field located in the North Sea which was revealed in 1993. The Field is situated 200 kilometers west of Stavanger at the southern terminal of the Norwegian division as shown in Figure 1 and five kilometers north of the Sleipner Ost field with water depths of 80 meters in the block 15/9. Drilling started in May 2007, came into production the following year, and ended in 2016 after 8.5 years in operation, more than twice as long as originally planned. New wells were being drilled up until 2012-13, which contributed to the increased recovery rate and extended the life of the field. However remaining resources were very limited and with the decrease in oil prices over recent years, new wells were no longer profitable. All possibilities to extend the life of the field were explored, which later yielded very good results.

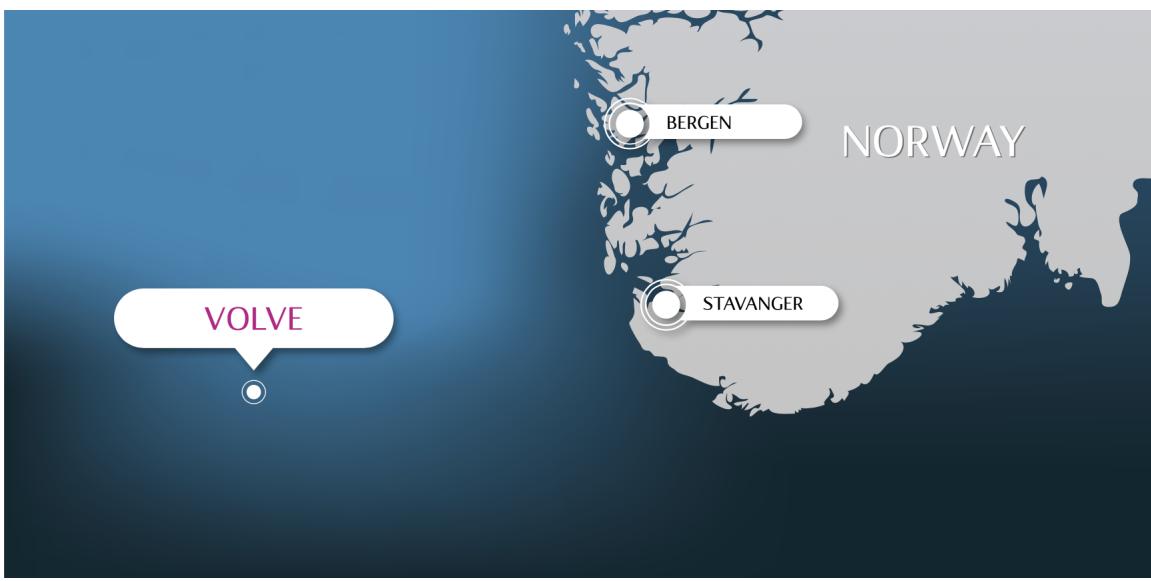


Figure 2.1. Location of Volve field

2.1.1. Reservoir

Volve produced oil from sandstone of Middle Jurassic age in the Hugin Formation. The reservoir is at a depth of 2,700-3,100 metres. The western part of the structure is heavily faulted and communication across the faults is uncertain. The field was produced with water injection for pressure support.

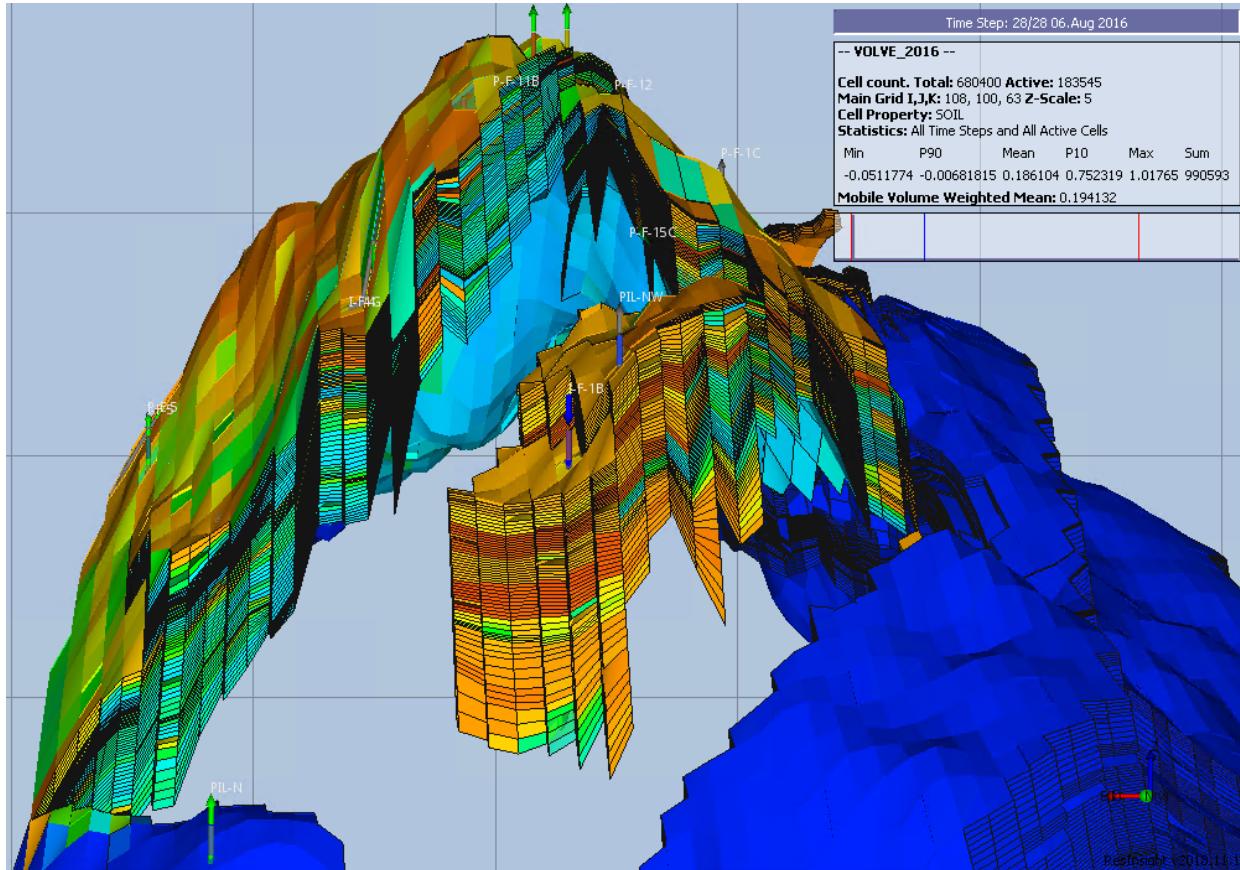


Figure 2.2. Reservoir model of Volve field

2.2. Production profile

The development was based on production from the Maersk Inspirer jack-up rig, with Navion Sga used as a storage ship to hold crude oil before export. Gas was piped to Sleipner which is a platform for final processing and export. Volve reached a recovery rate 54% and in March 2016 the license decided to shut down its production permanently. The field was originally scheduled for 3-5 years of operation.

Starting in February 2008, the Volve production lasted for about eight years. At peak, the field produced 56,000 barrels per day, and a total of 63 million barrels of oil were produced before the field was shut down in 2016. The field was developed when the oil price was low, and an unconventional concept was chosen to recover the resources in an easy and profitable manner. The field data will now have a new life for the purposes of study and research after the decommissioning. The Volve licensees were ExxonMobil and Bayergas. Information about geology, geophysics, drilling, static models, dynamic simulations and other information has provided by Equinor company in 2018[27]. Detailed production in each year has shown in the figure below:

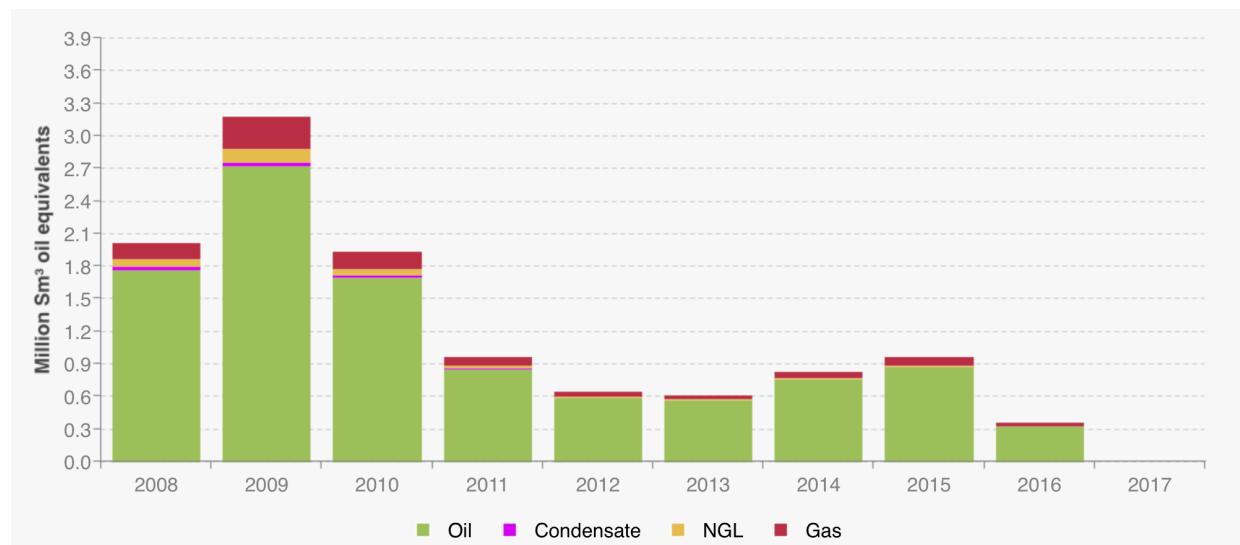


Figure 2.3. Production during the period 2008 to 2017

Table 2.1. Production profile of Volve filed.

DATE	BHP	BHT	DP	CS(%)	WHP	WHT	OIL	GAS	WAT
1/30/09	257.44	105.34	163.29	35.30	94.15	61.05	4535.43	649388.07	298.19
2/11/09	261.48	105.36	164.35	34.70	97.13	65.80	4379.88	629307.34	143.54
2/20/09	264.39	105.41	166.21	34.78	98.17	64.99	4509.07	638750.17	108.74
2/22/09	266.71	105.40	166.27	34.05	100.44	67.33	4319.02	612912.62	106.60
2/23/09	266.67	105.41	166.51	34.40	100.15	66.99	4417.66	625514.01	117.37
2/25/09	269.71	105.36	166.93	31.05	102.78	69.95	3226.61	460948.01	118.99
2/26/09	268.34	105.43	167.28	34.31	101.06	67.85	4411.90	628668.27	134.19
2/27/09	268.75	105.44	167.40	34.31	101.35	68.19	4376.91	625510.25	152.76
2/28/09	269.14	105.44	167.85	34.28	101.29	68.16	4417.91	626562.21	106.72
3/1/09	269.56	105.45	167.93	34.32	101.63	68.48	4396.74	628354.12	155.97
3/2/09	270.00	105.46	168.02	34.24	101.98	68.79	4381.09	623678.16	163.39
3/7/09	281.30	105.22	166.24	24.94	115.06	82.89	2208.96	316638.28	104.01
3/11/09	273.08	105.14	166.64	25.03	106.44	73.73	1185.05	165437.35	174.74
3/24/09	242.68	105.20	159.26	36.77	83.42	50.54	4379.47	611263.11	207.66
5/14/09	222.23	105.19	153.78	41.27	68.44	35.29	4211.25	576830.95	104.06
5/16/09	223.00	105.22	154.26	41.29	68.74	35.58	4237.94	583860.51	104.19
5/17/09	223.10	105.23	154.59	41.33	68.51	35.36	4244.23	585468.54	104.30
5/18/09	223.29	105.25	154.64	41.33	68.65	35.50	3759.11	519316.88	127.10
5/19/09	223.37	105.27	154.79	41.32	68.58	35.44	3967.32	548750.04	110.41
5/20/09	223.58	105.29	155.00	41.31	68.58	35.44	3972.97	548982.34	148.53
5/21/09	223.79	105.30	155.31	41.33	68.48	35.32	3994.95	550388.29	153.09
...
...
6/17/16	269.73	100.07	238.84	44.94	30.89	5.37	103.29	16456.74	2979.58
6/18/16	269.68	100.08	238.86	44.99	30.82	5.31	103.44	16551.02	2972.75

6/19/16	269.65	100.09	238.87	45.02	30.78	5.27	103.55	16524.52	2984.83
6/20/16	269.60	99.81	238.88	44.90	30.73	5.20	103.75	16525.42	2978.42
6/21/16	269.61	100.11	238.88	44.92	30.73	5.20	103.44	16508.09	2992.62
6/22/16	269.58	100.12	238.96	44.95	30.63	5.12	103.21	16511.03	2993.41
6/23/16	269.55	100.13	238.94	44.93	30.61	5.11	103.27	16566.18	3022.90
6/24/16	269.57	100.14	238.85	44.80	30.72	5.19	103.37	16499.88	2980.02
6/25/16	269.52	100.15	238.85	44.81	30.67	5.15	102.57	16434.67	2981.04
6/26/16	269.48	100.16	238.84	44.86	30.65	5.11	102.85	16501.91	2983.48
6/27/16	269.76	100.17	238.90	45.11	30.86	5.34	101.46	16274.31	2976.42
6/28/16	269.88	100.17	238.91	45.13	30.97	5.45	102.26	16444.37	2973.66
6/29/16	269.83	100.19	238.94	45.20	30.89	5.37	102.38	16458.56	2980.12
6/30/16	269.83	100.20	238.89	45.16	30.93	5.40	101.47	16397.18	2985.45
7/1/16	269.89	100.20	238.91	45.16	30.98	5.32	102.89	16369.86	2979.13
7/2/16	269.79	100.22	238.87	45.18	30.93	5.39	101.58	16365.91	2992.00
7/3/16	269.79	100.23	238.83	45.11	30.96	5.44	102.43	16269.74	2976.87
7/4/16	269.78	100.24	238.82	45.08	30.96	5.44	100.67	16263.02	2990.10
7/5/16	269.77	100.25	238.80	45.08	30.97	5.43	101.88	16284.48	2974.30
7/7/16	266.20	100.31	238.44	93.42	27.76	2.19	106.19	17427.78	3172.96
7/8/16	266.04	100.33	238.47	100.00	27.56	2.00	106.30	17541.20	3187.95
7/9/16	268.81	100.30	239.08	82.19	29.73	4.11	102.09	16681.29	2326.24
7/10/16	265.92	100.34	238.40	100.00	27.52	1.96	113.38	18753.12	3185.47
7/11/16	267.77	100.32	238.64	91.16	29.13	3.41	108.84	17979.28	3056.29
7/12/16	266.00	100.35	238.27	100.00	27.73	1.94	113.84	18543.76	3148.91

Chapter 3. Artificial Intelligence

Artificial intelligence is a branch of computer science that aims to create intelligent machines. It has become an essential part of the technology industry. Artificial intelligence (AI) is the simulation of human intelligence processes by machines, especially computer systems. These processes include learning (the acquisition of information and rules for using the information), reasoning (using rules to reach approximate or definite conclusions) and self-correction. Particular applications of AI include expert systems, speech recognition and machine vision.

Machine learning and deep learning are core part of AI. Learning without any kind of supervision requires an ability to identify patterns in streams of inputs, whereas learning with adequate supervision involves classification and numerical regressions. Classification determines the category an object belongs to and regression deals with obtaining a set of numerical input or output examples, thereby discovering functions enabling the generation of suitable outputs from respective inputs.

Artificial Intelligence (AI) is transforming the oil and gas industry. A study from McKinsey (2016) in 2016 mentioned that digital organization, which covers AI and Machine Learning, is one of the five big ideas that will reshape the future of the industry. Digitizing and automating work will result in better safety and productivity as fewer people will be at risk and the risk of human error is reduced. Moreover, in the low oil price environment, oil companies often have no choice but to cut their operational budget, reduce the headcount, and/or fully optimize their remaining assets. This further encourages the adoption of artificial intelligence in the industry, with the hope to simplify and/or automate inefficient processes[8].

There have been some implementations of AI in the field of oil and gas production. Boomer (1995) [9] used Neural Network to forecast oil production in Vacuum Field, Texas and the model outperformed the professionals 93% of the time. Cao et al. (2016) found

that Neural Network performed better than conventional decline curve methods (Arps, Duong, and Stretched Exponential Production Decline). Suhag et al. (2017) [10] implemented Neural Network to predict oil rate in Bakken shale wells and yielded better results than Duong's method. J. Sun, X. Ma, and M. Kazi [11] have recently propose a novel method of using Recurrent Neural Networks for production forecast and it give outperformed results compared to conventional methods like DCA, SEPD and Duong.

In this thesis, author have develop a work-flow of using machine learning and deep learning technique to predict future production rate. Ensemble methods have been built to gain better result. Author also used three kind of Recurrent Neural Network for prediction.

3.1. Machine learning

Machine learning is a statistical technology to get computers output expected results without being explicitly programmed. Machine learning is based on big-data without explicitly programming, therefore, it saves programming time and avoids various ideal scenarios restrictions, and it also improves the output prediction accuracy. Machine learning method such as Artificial Neural Network (ANN) is applied in oil and gas industry for production forecasting [7](Cao et al., 2016).

The main objective of every machine learning algorithms is to find a optimal set of parameters \mathbf{w} tailoring of the hypothesis to a specific set of data. In this paper, we have used 12 different machine learning algorithms for prediction. A brief overview of 12 algorithms we have used will be introduced below.

3.1.1. Linear Regression

The simplest linear model for regression is one that involves a linear combination of the input variables[13]

$$y(\mathbf{x}, \mathbf{w}) = w_0w_1x_1 + \dots + w_Dx_D \quad (3.1)$$

where $x = (x_1, \dots, x_D)^T$. The key property of this model is that it is a linear function of the parameters w_0, \dots, w_D . It is also, however, a linear function of the input variables x_i , and this imposes significant limitations on the model. We therefore extend the class of models

by considering linear combinations of fixed nonlinear functions of the input variables, of the form:

$$y(\mathbf{x}, \mathbf{w}) = w_0 + \sum_{j=1}^{M-1} w_j \phi_j(x) \quad (3.2)$$

where $\phi_j(x)$ are known as basis functions.

In this research, we just applied linear regression as the simplest model of machine learning algorithms for prediction.

3.1.2. Ridge

Ridge is a Linear least squares with L2 regularization. Ridge regression addresses some of the problems of Ordinary Least Squares by imposing a penalty on the size of coefficients.

The ridge coefficients minimize a penalized residual sum of squares.[22]

$$\min_w = \|Xw - y\|_2^2 + \alpha \|w\|_2^2 \quad (3.3)$$

Here, $\alpha \geq 0$ is a complexity parameter that controls the amount of shrinkage: the larger the value of α , the greater the amount of shrinkage and thus the coefficients become more robust to collinearity.

3.1.3. KNeighbors Regression

K-Nearest Neighbors uses local neighborhood points to make a prediction. In more detail, distance (Euclidean distance in this study) is calculated from a query point $x^{(i)}$ in the test set to each data point $x'^{(i)}$ in the training time:

$$D(x, x') = \sqrt{\sum_{i=1}^N (x^{(i)} - x'^{(i)})^2} \quad (3.4)$$

Continuously, sorting the distance and pick k closest data points. Define the k points closest points in the training set that are closest to x the set G. The prediction during testing time is calculated by local interpolation of the nearest neighbors in G [8].

$$\hat{y}^i = \frac{1}{K} \sum_{j \in G}^N y_j \quad (3.5)$$

where \hat{y}^i is the output of KNeighbors regressor and k is the number of nearest neighbors points.

3.1.4. Support Vector Regression

The method of Support Vector Regression is extended from SVM algorithms to solve the regression problems.

Given training vectors $x_i \in \mathbb{R}^p, i = 1, \dots, n$ and a vector $y \in \mathbb{R}^n$, SVR solves the following primal problem:

$$\min_{w, b, \zeta, \zeta^*} \frac{1}{2} w^T w + C \sum_{i=1}^n (\zeta_i + \zeta_i^*) \quad (3.6)$$

$$\text{subject to : } y_i - w^T \phi(x_i) - b \leq \varepsilon + \zeta_i^*,$$

$$w^T \phi(x_i) + b - y_i \leq \varepsilon + \zeta_i^*,$$

$$\zeta_i, \zeta_i^* \geq 0, i = 1, \dots, n.$$

3.1.5. Decision Tree Regression

Decision Tree [14] (also called Regression Tree in regression problems) is a supervised learning model that creates branches of decision nodes. Decision Tree learns a dataset by making the best split of features in the training set into subsets based on the homogeneity of each new subset. The process goes on until a stopping criterion (e.g. tree size or number of observation per node) is reached (Breiman et al. 1984).

Data $y \in \mathbb{R}^n, X \in \mathbb{R}^{n \times p}$ and observation $(y_i, x_i) \in \mathbb{R}^{p+1}, i = 1, \dots, n$. Suppose we have a partition of \mathbb{R}^p into M regions R_1, \dots, R_m . We predict the response using a constant on each R_i :

$$f(x) = \sum_{i=1}^m c_i \cdot \mathbf{1}_{x_i \in R_i} \quad (3.7)$$

In order to optimize $\sum_{i=1}^n (y_i - f(x_i))^2$, one needs to choose: $\hat{c}_i = \text{ave}(y_j : x_j \in R_i)$.

Consider a splitting variable $j \in 1, \dots, p$ and splitting point $s \in \mathbb{R}$. Define the two half-plane:

$$R_1(j, s) := \{x \in \mathbb{R}^p : x_j \leq s\} \quad R_2(j, s) := \{x \in \mathbb{R}^p : x_j > s\}$$

Choose j, s to minimize:

$$\min_{j,s} \left[\min_{c_1 \in \mathbb{R}} \sum_{x_i \in R_1(j,s)} (y_i - c_1)^2 + \min_{c_2 \in \mathbb{R}} \sum_{x_i \in R_2(j,s)} (y_i - c_2)^2 \right] \quad (3.8)$$

3.1.6. Random Forest Regression

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees[15].

A random forest is a meta estimator that fits a number of classifying decision trees on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. The generic formula is shown:

$$\hat{f}(X) = \frac{1}{B} \sum_{b=1}^B f_b(X) \quad (3.9)$$

where $\hat{f}(X)$ is the output of random forest mapping function, $f_b(X)$ is the output of individual tree function, and B is the number of classifier or estimator.[19]

3.1.7. Extra Trees Regression

The Extra-Tree method (standing for extremely randomized trees) was proposed in [GEW06], with the main objective of further randomizing tree building in the context of numerical input features, where the choice of the optimal cut-point is responsible for a large proportion of the variance of the induced tree.

The Extra-Trees algorithm builds an ensemble of unpruned decision or regression trees according to the classical top-down procedure. Its two main differences with other tree-based ensemble methods are that it splits nodes by choosing cut-points fully at random and that it uses the whole learning sample (rather than a bootstrap replica) to grow the trees. With respect to random forests, the method drops the idea of using bootstrap copies of the learning sample, and instead of trying to find an optimal cut-point for each one of the K randomly chosen features at each node, it selects a cut-point at random. A detailed ex-

planation of ExtraTrees algorithms has been shown in work of Pierre Geurts, Damien Ernst and Louis Wehenkel[20].

3.1.8. Ada Boost Regression

AdaBoost creates T weak learners h_t (in this study trees) and boosts these weak learners to stronger ones. It does so by assigning more weight to data points whose error is high. In this study, we used linear loss function to update the weights. In the next iteration, the algorithm will try to avoid mispredicting these data points as it would greatly increase the penalty/error. Similar to Random Forest, the final prediction H is computed as a weighted average of the weak learners' prediction.[8]

$$H(x) = \text{sign}\left(\sum_{t=1}^T \alpha_t h_t(x)\right) \quad (3.10)$$

where $H(x)$ is the output of Adaboost mapping function, α_t is the weight assigned to each classifier, $h_t(x)$ is the ouput of the weak classifier or estimator.[19]

3.1.9. Gradient Boosting Regression

Similar to AdaBoost, Gradient Boosting (or Gradient Boosting Machine) uses an ensemble of weak prediction classifiers (decision trees, in this study). Each individual tree is trained as a decision tree estimator. The model trains the tree one at a time and calculates mean squared error loss. The next step is gradient descent. But the unique step here is instead of updating parameter value, the gradient descent updates a tree function by modifying its parameters so that it moves to the right gradient[21].

$$F(x) = \sum_{m=1}^M \gamma_m h_m(x) \quad (3.11)$$

where $h_m(x)$ are the basis functions which are usually called weak learners in the context of boosting. Gradient Tree Boosting uses decision trees of fixed size as weak learners. Decision trees have a number of abilities that make them valuable for boosting, namely the ability to handle data of mixed type and the ability to model complex functions.[22]

3.1.10. Extreme Gradient Boosting Regression

XGBoost is one of the most popular and efficient implementations of the Gradient Boosted Trees algorithm, a supervised learning method that is based on function approximation by optimizing specific loss functions as well as applying several regularization techniques. XGBoost is an optimized distributed gradient boosting library designed to be highly efficient, flexible and portable. It implements machine learning algorithms under the Gradient Boosting framework. XGBoost provides a parallel tree boosting (also known as GBDT, GBM) that solve many data science problems in a fast and accurate way. The same code runs on major distributed environment (Hadoop, SGE, MPI) and can solve problems beyond billions of examples.[23] The objective function (loss function and regularization) at iteration t that we need to minimize is the following:

$$Obj = \sum_{i=1}^n \mathcal{L}(y_i, \hat{y}_i) + \sum_{k=1}^K \Omega(f_k) \quad (3.12)$$

$$\Omega(f_t) = \gamma \mathbb{I} + \frac{1}{2} \lambda \sum_{j=1}^T w_j^2$$

$$f_t(x) = w_{q(x)}, w \in \mathbb{R}^T, q \in \mathbb{R}^d \rightarrow 1, 2, \dots, T$$

where $\mathcal{L}(y_i, \hat{y}_i)$ is training loss, $\Omega(f_k)$ is regularization, measures the complexity of trees and w is the leaf weight of the tree.

3.1.11. Light Gradient Boosting Regression

LightGBM is a gradient boosting framework developed by Microsoft that uses tree based learning algorithms. It is designed to be distributed and efficient with the advantages: faster training speed and higher efficiency, lower memory usage, better accuracy, support of parallel and GPU learning, capable of handling large-scale data.[24]

Light GBM grows tree vertically while other algorithm grows trees horizontally meaning that light GBM grows tree leaf-wise while other algorithm grows level-wise. It will choose the leaf with max delta loss to grow. When growing the same leaf, Leaf-wise algorithm

can reduce more loss than a level-wise algorithm.

Both LightGBM and XGBoost are widely used and provide highly optimized, scalable and fast implementations of gradient boosted machines (GBMs).

In terms of LightGBM specifically, a detailed overview of the LightGBM algorithm and its innovations is given in the NIPS paper [24].

3.2. Deep learning

3.2.1. Artificial neural networks

First introduced in the early 1980s, neural networks are a programming paradigm within deep learning that enable computers to learn from observational data. They leverage the physiology and architecture of the human brain [16].

Simple architecture of neural networks is show below:

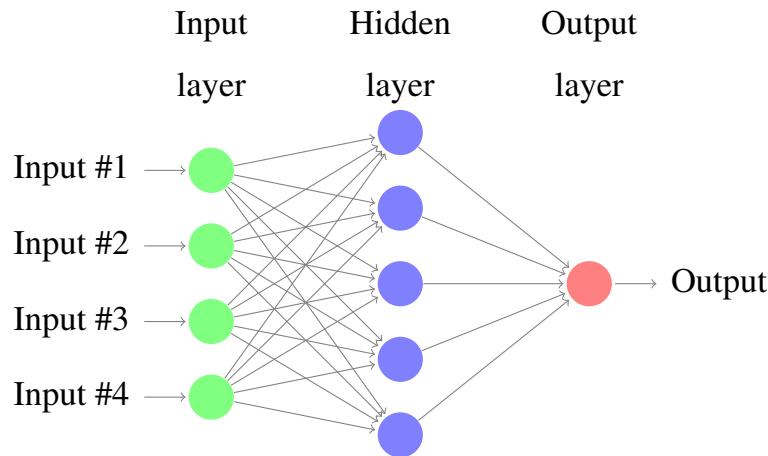


Figure 3.1. Shallow neural networks with one hidden layer

ANN has drawn attention from well testing researchers since the 1990s. An early application in well testing was introduced by Al-Kaabi and Lee (1990) to identify the interpretation models from derivative plots. Juniardi and Ershaghi (1993) studied the complexities of using ANN with a focus on well test analysis of faulted reservoirs, and discussed some of the shortcomings of the ANN method. Around the same time, Ershaghi et al. (1993) proposed an enhanced approach based on the work of Al-Kaabi and Lee (1990), by train-

ing multiple ANNs where each neural net was designed to learn the patterns for a specific reservoir model. In 1995, Athichanagorn and Horne combined ANN with sequential predictive probability method to improve the accuracy of interpretation. ANN was used to generate initial parameter estimates of the candidate reservoir models that it identified. The candidate models and initial estimates were then passed to the sequential predictive probability method for final evaluations.

3.2.2. Deep feed-forward networks

Deep feed-forward networks, also called feed-forward neural networks, or multilayer perceptron (MLPS), are the quintessential learning models. The goal of a feedforward networks is to approximate some function f . For example, for a classifier, $y = f(x)$ maps an input \mathbf{x} to a category y . A feed-forward networks defines a mapping $\mathbf{y} = f(\mathbf{x}, \theta)$ and learns the values of parameters θ that result in the best function approximation.[18]

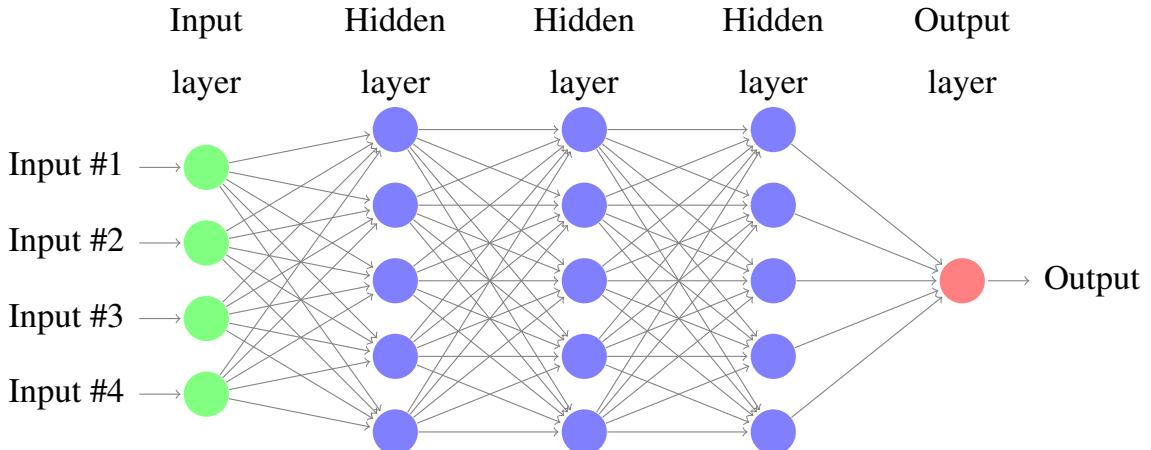


Figure 3.2. Deep feed forward neural networks

Recently, deep feed forward neural networks (or deep neural networks) has been applied widely in oil and gas industry. A new approach to reservoir characterization using deep neural networks has been introduced by M. Korjani, Andrei Popa, Eli Grijalva, and Steve Cassidy [29]; well bore stability prediction [30]; deep recurrent neural network for surface pressure response during the hydraulics fracturing process [31]; deep neural network based prediction of leak-off pressure in offshore Norway [32]; New method for extract-

ing the work status in Shipyard using Deep Neural Networks [33]; diagnostics of rod pumps with use of deep neural networks [35]; automatic salt-body classification using a deep convolutional neural network [36]; seismic problems has been issued by deep neural networks in various papers such as: deep neural network architectures arising in seismic inverse problems[34]; Application of convolutional and deep neural networks for GPU based seismic interpretations [37]. Beside, deep convolutional neural networks has also been built for geology problems: deep learning convolutional neural networks to predict porous media properties [38], visual explanations from convolutional neural networks for fault detection[39].

3.2.3. Simple Recurrent neural networks

Recurrent neural networks (RNNs) exploit characteristics of a feedback network, allowing the model to construct a sequential representation of input data. However, simple RNNs have been notoriously difficult to train, due to their iterative nature and the highly volatile relationship between the parameters and the hidden states [17].

The network is given a sequence of inputs. Then, the network computes a sequence of hidden states and a sequence of predictions for each step in time. The sequences are computed by iterating through a series of equations that updates the model's weight matrices, hidden layers, and the output units. The RNN utilizes predefined vector-valued functions that contain a computable Jacobian and are typically non-linear and applied coordinate-wise[25].

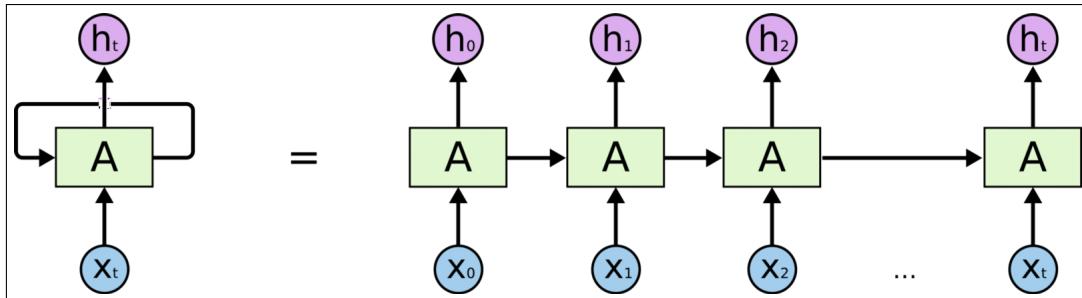


Figure 3.3. Recurrent neural network simple architecture[26]

3.2.4. Long short term memory

Long short-term memory (LSTM) units (or blocks) are a building unit for layers of a recurrent neural network (RNN). A RNN composed of LSTM units is often called an LSTM network. A common LSTM unit is composed of a cell, an input gate, an output gate and a forget gate. The cell is responsible for "remembering" values over arbitrary time intervals; hence the word "memory" in LSTM. Each of the three gates can be thought of as a "conventional" artificial neuron, as in a multi-layer (or feedforward) neural network: that is, they compute an activation (using an activation function) of a weighted sum. Intuitively, they can be thought as regulators of the flow of values that goes through the connections of the LSTM; hence the denotation gate and there are connections between these gates and the cell. A detailed explanation has been introduced by Colah in his blogs about long short term memory[26].

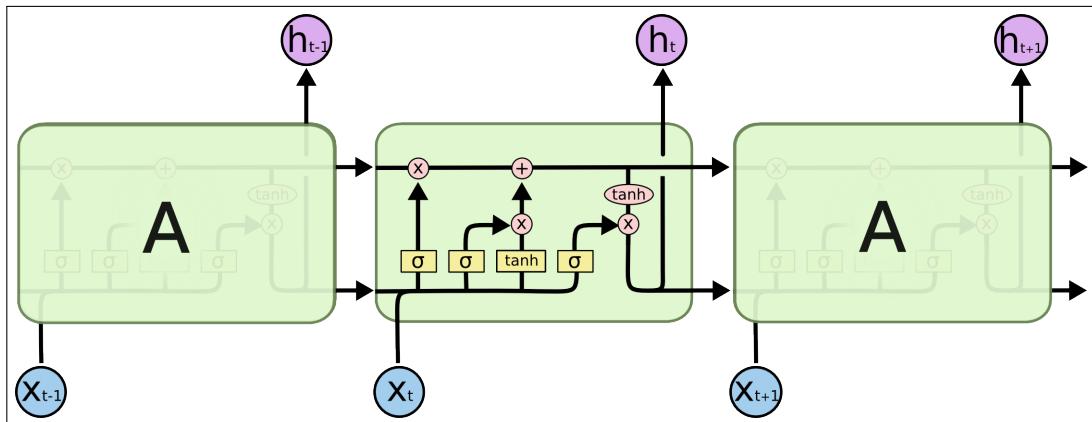


Figure 3.4. Simple stretch of long short term memory[26]

3.2.5. Gated Recurrent Unit

In simple words, the GRU unit does not have to use a memory unit to control the flow of information like the LSTM unit. It can directly makes use of the all hidden states without any control. GRUs have fewer parameters and thus may train a bit faster or need less data to generalize. But, with large data, the LSTMs with higher expressiveness may lead to better results.

They are almost similar to LSTMs except that they have two gates: reset gate and update gate. Reset gate determines how to combine new input to previous memory and update gate determines how much of the previous state to keep. Update gate in GRU is what input gate and forget gate were in LSTM. We don't have the second non linearity in GRU before calculating the output, neither they have the output gate.

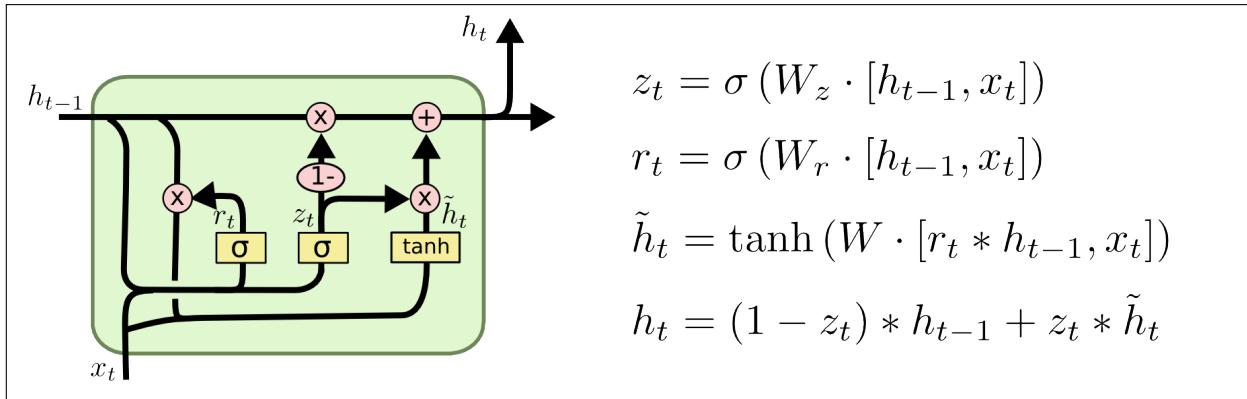


Figure 3.5. Simple stretch of Gated Recurrent Unit[26]

Chapter 4. Methodology

4.1. Input data

The production dataset used in this paper is from Volve field on Norwegian continental shelf with around 8 years history. Daily oil, gas and water production data are all recorded consist with pressure, temperature and choke size as operation constraint. These production data are the main input variables for the machine learning and deep learning model, and it gives multi-phase production predictions in the future as model output.

In this research, bottom hole pressure (BHP), tubing head pressure (THP), bottom hole temperature (BHT), well head temperature (WHT), different pressure in casing (DP), choke size (CS) in percentage is the input for training machine learning and deep learning algorithms.

Table 4.1. Description of data set

	BHP	BHT	DP	CS	WHP	WHT	OIL	GAS	WATER
Count	2365.0	2365.0	2365.0	2365.0	2365.0	2365.0	2365.0	2365.0	2365.0
Mean	249.09	102.09	209.74	78.48	39.36	82.10	1217.04	180380.41	2968.53
Std	14.47	3.35	22.44	24.82	12.61	2.23	1052.86	149356.18	934.56
Min	135.63	54.64	103.14	20.30	27.19	72.14	100.33	5928.54	100.16
25%	241.72	99.73	197.39	55.85	31.28	84.48	322.41	49848.46	2694.0
50%	247.56	101.14	208.75	97.92	32.84	87.20	950.38	151088.07	3203.3
75%	261.78	105.05	230.24	100.0	42.23	88.63	1876.97	285954.06	3541.83
Max	281.31	106.77	239.84	100.0	115.06	93.51	5644.37	786328.36	5691.77

4.2. Work-flow

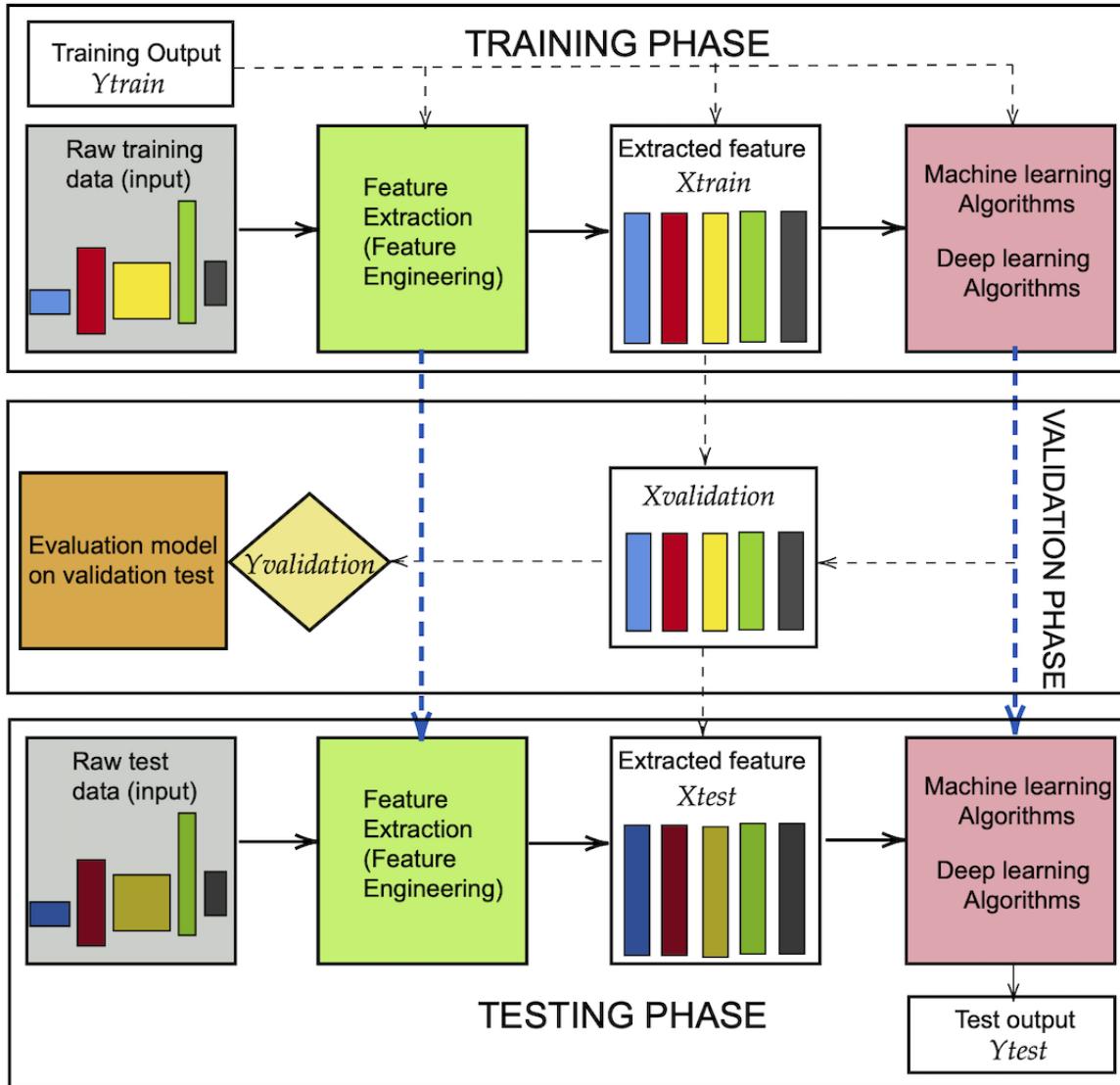


Figure 4.1. Work flow for using machine and deep learning algorithms

Almost every machine learning and deep learning algorithm has followed this work-flow above. First, all raw data need to be collected altogether. Feature engineering means the process of using domain knowledge of the data to create features that make machine learning algorithms work. Feature engineering is an informal topic, but it is considered essential in applied machine learning[28]. From Andrew Ng, a famous Professor in AI : "Applied machine learning is basically feature engineering". Feature engineering in this

research consists of three step from preprocessing to cross-validation (CV) strategies and feature scaling. Machine learning models are implemented with Scikit-learn library, and deep learning models are also implemented with Keras library, two of these very useful and popular libraries for working with ML/DL algorithms.

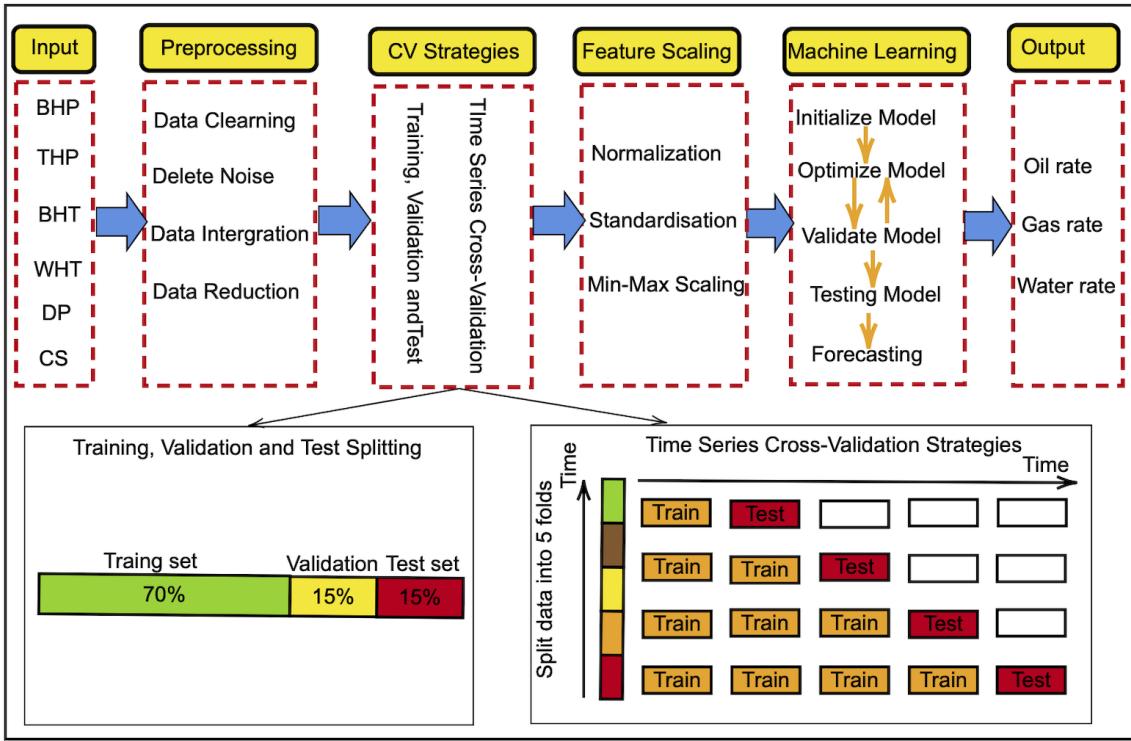


Figure 4.2. Detailed work flow for using machine learning and deep learning algorithms

4.2.1. *Gather data*

The process of gathering data depends on the type of problems that need to solve. In this thesis, raw production data are gathered in excel file include 24 columns data and each column represents specific type of data. Due to the mess of data include missing and noisy data, before applying any machine learning and deep learning methods, data need to be applied feature engineering in order to get clean and useful data.

4.2.2. *Data preprocessing*

Data preprocessing is a processing of cleaning the raw data into clean data. Some type of missing data, noisy data and inconsistent data need to be through a filter which is usually

called "Feature Extraction" or feature engineering. In this thesis, even though there are 24 columns of raw data, almost of them is text data that give information about the type, ID, well-bore code and file name data and other unusefull data for applying ML/DL algorithms. Another important aspect is outlier of the data. There are some error data that might be present in the dataset that deviated significantly other from observed data. So, this kind of data also need to be eliminated.

4.2.3. *Cross validation strategies*

There are two scenarios of cross-validation (CV) strategies have been built in this research. In first scenario, data is divided in three parts: training data (70%), validation data (15%) and test data (15%) with suffering all (figure 7). In the second scenario, data is separated in to 5 different folds (figure 8) and algorithms will train and test alternatively in training and testing data in very loop.

4.2.4. *Feature scaling*

Feature scaling is a method used to normalize the range of independent variables or features of data. In data processing, it is also known as data normalization and is generally performed during the data preprocessing step. In this thesis, three ways feature scaling have been used:

Normalization

$$x' = \frac{x - \text{average}(x)}{\max(x) - \min(x)} \quad (4.1)$$

Standardization

The result of standardization is that the features will be rescaled so that they will have the properties of a standard normal distribution with $\mu = 0 \& \sigma = 1$.

$$x' = \frac{x - \bar{x}}{\sigma} \quad (4.2)$$

where μ is the mean (average) and σ is the standard deviation.

Min-max scaling

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (4.3)$$

Min-max scaling have been chosen due to the reasons of unchangeable distribution of the data and better results compare with other methods.

4.2.5. Machine learning and deep learning algorithms

As shown in the Figure 4.2, a typical loop of using machine learning and deep learning algorithms is from initializing model to optimize, validate, testing and forecasting. Each of these steps have its specific definition and methods.

1. Initialize model means setting all parameters of model to specific values. There are two popular ways is initialize all parameters to zero or to random values from standard normal distribution or uniform distribution and multiply it by a scalar such as 10.
2. Optimize model means finding the best combination of hyperparameters of models for the purpose of getting the best result. During the training process, manual tuning and grid search are used for finding the best combination of hyperparameters of algorithms. There is a loop between optimizing and validate model means that repeating the procedure until reaching the desired accuracy score.
3. Validate and testing model are referred to as the processing where a trained model is evaluated with a validation and testing dataset. They are aimed to find the optimal model with the best performance. There are a number of statical metrics that can be used for validating the results. Mean square error, maximum absolute error, coefficient of determination and person correlation have been used.
4. After completing all previous steps above, the model can have ability to forecast. In this study, production rate include oil rate, gas rate, and water rate are the outputs from the training model.

Chapter 5. Result and Discussion

5.1. Machine learning

The the table bellow we show the result of every algorithm we have used. Coefficient of determination (R^2) and mean squared error (MSE) have been used to evaluation algorithm.

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$$

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

Pearson correlation is also used for testing the model.

$$\rho_{X,Y} = \frac{cov(X, Y)}{\sigma_X \sigma_Y}$$

Table 5.1. Score and error of machine learning algorithms on oil

No	Model	Validation Score	Validation Error	Test Score	Test Error
1	Linear Regression	0.82132	0.00598	0.80275	0.00609
2	Ridge	0.8257	0.00583	0.78125	0.00676
3	KNeighbors Regression	0.9218	0.00262	0.95795	0.00130
4	Support Vector Regression	0.8556	0.00483	0.84564	0.00477
5	Decision Tree Regression	0.94392	0.001876	0.95973	0.00124
6	Random Forest Regression	0.9687	0.00138	0.9585	0.00128
7	Extra Tree Regression	0.9569	0.00144	0.96031	0.00123
8	AdaBoost Regression	0.90031	0.00334	0.90247	0.00301
9	Gradient Boosting Regression	0.9554	0.00149	0.9597	0.00124
10	XgBoost Regression	0.92087	0.00265	0.93438	0.00203
11	LightGBM Regression	0.89392	0.00355	0.93392	0.00204

After comparing the result on validation and test set, we just picked up five algorithms that yield the best score on test set. Score and error of gas and water prediction can be found

in appendix (table 10 and 11). In order to evaluate how possible the algorithms can be applied to forecast a new set of data, we use the Pearson correlation for evaluation. The table below shows the Pearson correlation for the best five algorithms.

Table 5.2. Pearson correlation of best five algorithms

No	Model	Pearson correlation (oil)	Pearson correlation (gas)	Pearson correlation (water)
1	Extreme Gradient Boosting	0.96667	0.96895	0.8988
2	Decision Tree Regression	0.9798	0.9596	0.8970
3	Random Forest Regression	0.97927	0.9757	0.9161
4	Gradient Boosting Machine	0.96689	0.9745	0.92
5	Extra Tree Regression	0.98065	0.98205	0.9351

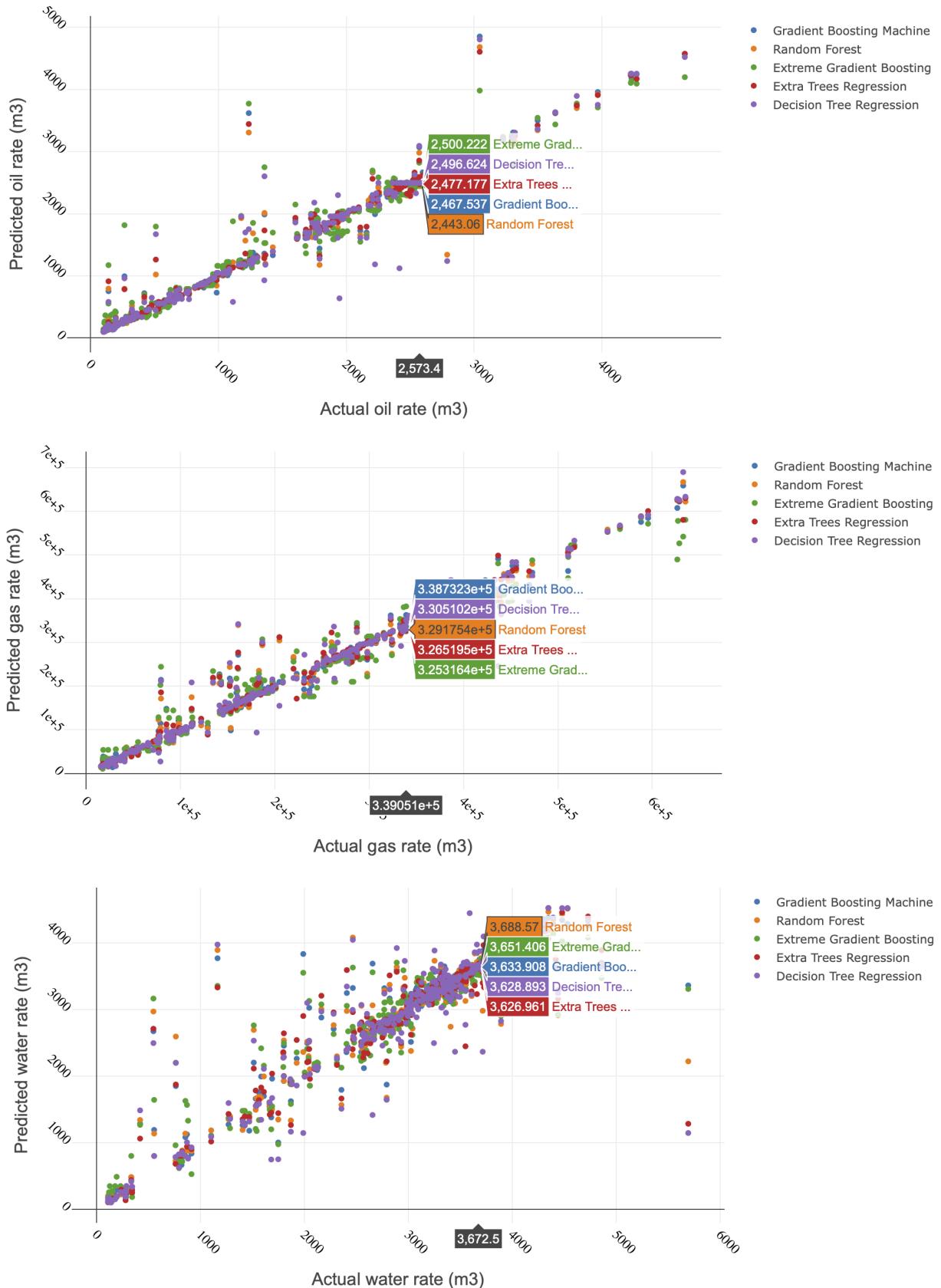


Figure 5.1. Production rate prediction versus actual values

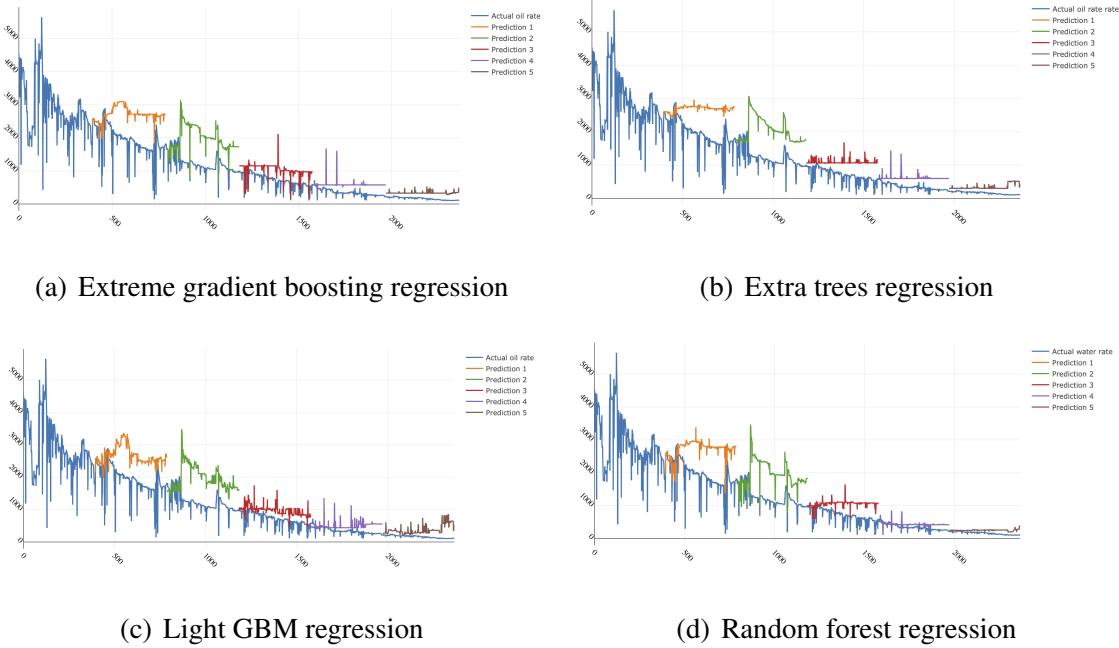


Figure 5.2. Oil prediction using time series cross-validation method.

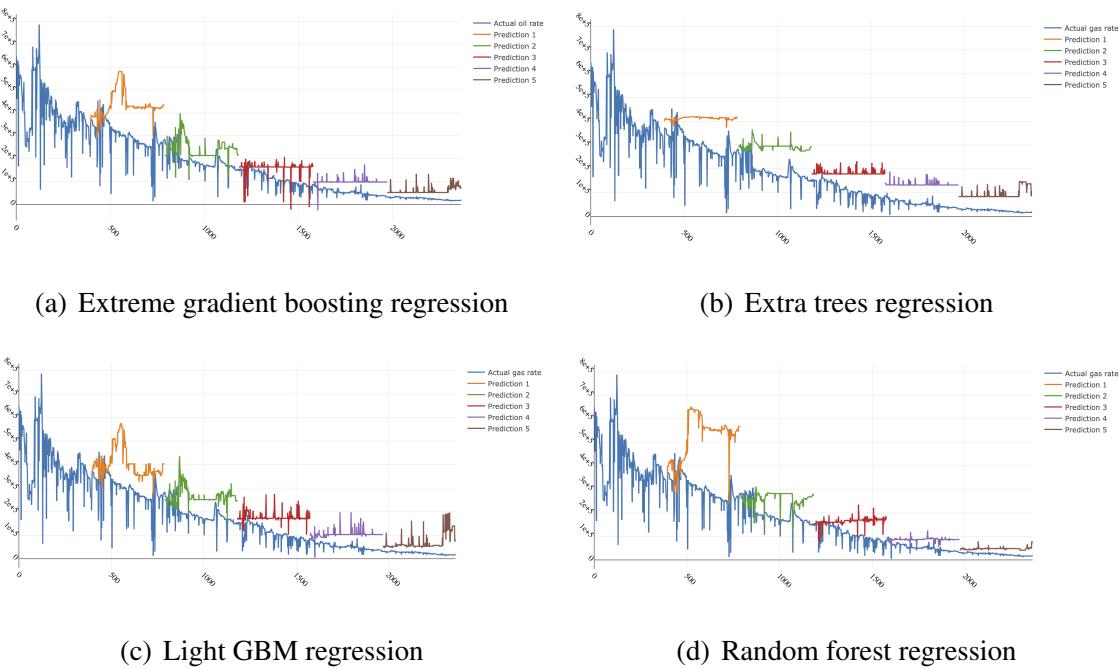


Figure 5.3. Gas prediction using time series cross-validation method.

With all machine learning algorithms we have used, data need to be shuffled before training, the dependence of production rate followed time have vanished. Therefore, we continue to use time series cross-validation strategy on the same data set. Instead of using gradient boosting machine regressor, authors used LGBM and XGB algorithms because these algorithms have better result compare with GBM. Figure 5.2 and 5.3 have shown result of using time series cross-validation method on oil and water. There is a gap between actual values and prediction demonstrate the algorithms are not sufficient for time series prediction. Decision tree and random forest regression give better result compare with other algorithms.

5.2. Deep learning

Neural networks have been used widely in oil and gas industry since the time it is created. Compare with other machine learning algorithms, NNs have given better or worse result depend on the process of training. While shallow neural networks with one hidden layer (25 nodes) give lower score and error is bigger, deep neural networks with three hidden layers have increased the score and reduce error.

In this research, hyperparameters of neural networks are tuned (table 8) to find the best combination of them for giving a good result on the validation test. Moreover, we have used some very novel technique in AI, dropout and ReLU activation function, to train our neural networks.

5.2.1. Neural networks

Table 5.3. Score and error on oil prediction

No	Model	Validation Score	Validation Error	Test Score	Test Error
1	Shallow neural networks	0.8563	0.0048	0.8426	0.00486
2	Deep neural networks	0.9492	0.001568	0.94	0.00199

Table 5.4. Score and error on gas prediction

No	Model	Validation Score	Validation Error	Test Score	Test Error
1	Shallow Neural Networks	0.6826	0.0101	0.6068	0.00948
2	Deep Neural Networks	0.7824	0.00551	0.7767	0.00574

Table 5.5. Score and error on water prediction

No	Model	Validation Score	Validation Error	Test Score	Test Error
1	Shallow Neural Networks	0.6826	0.0101	0.6068	0.00948
2	Deep Neural Networks	0.7824	0.00551	0.7767	0.00574

Table 5.6. Pearson correlation of neural networks

No	Model	Pearson correlation (oil)	Pearson correlation (gas)	Pearson correlation (water)
1	Shallow Neural Networks	0.9225	0.9372	0.8314
2	Deep Neural Networks	0.9798	0.9711	0.9014

By plotting the actual and predicted values in one figure, a clear improvement of using deep neural networks has shown below:

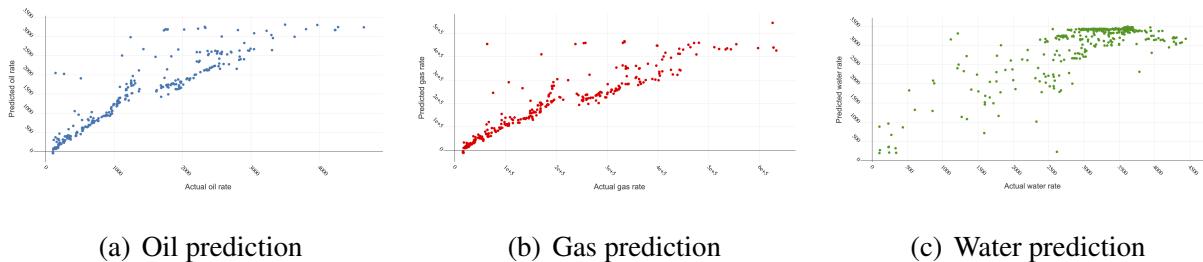


Figure 5.4. Production rate prediction using shallow neural networks.

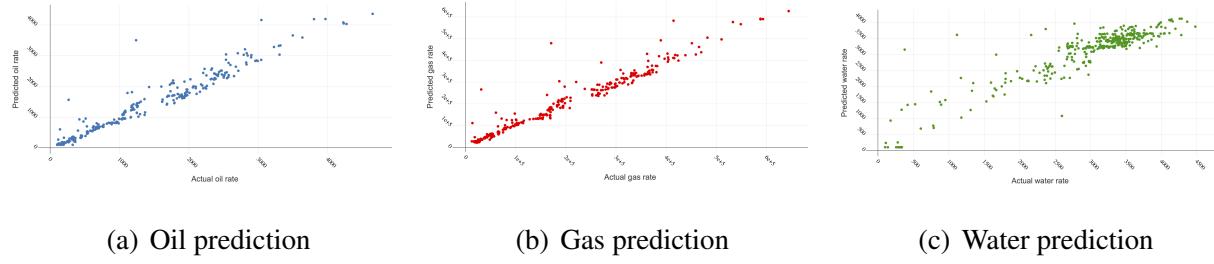


Figure 5.5. Production rate prediction using deep neural networks.

It is obvious that neural networks have given better results when increasing the number of hidden layers from 1 to 3. However, like other machine learning algorithms, data also need to be shuffled before training. That lead to the loss of temporal characteristic. It is the main problem of using machine learning algorithms include neural networks if data is shuffled. This problem can be solved by using the recurrent neural networks which has been proven to be one of the most successful algorithms of time-series problems.

5.2.2. Recurrent Neural Networks

Recurrent neural networks algorithms are very powerful with time-series data. Data set is split in three-part train, validation and final 2 years (15%) is the test set. Compare with machine learning models, RNNs have improved remarkably when predicted in the last two year. From figure 5.6, 5.7 and 5.8 above when using Recurrent neural networks (LSTM, GRU, SimpleRNN), a very close matching between prediction and actual values. Since GRU and LSTM give the best results on oil prediction, SimpleRNN is better with water forecasting.

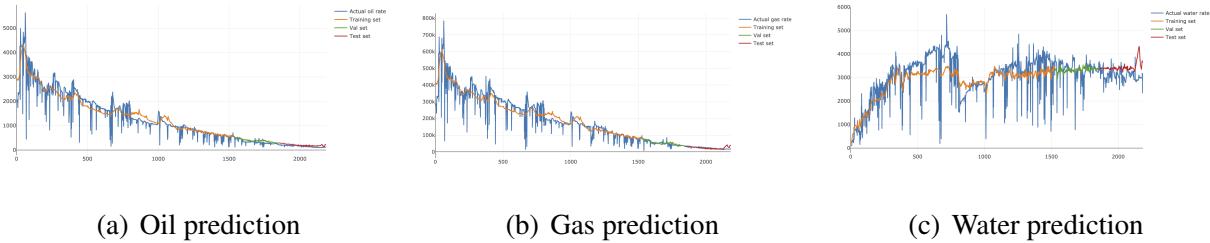


Figure 5.6. Production rate prediction using long short term memory.

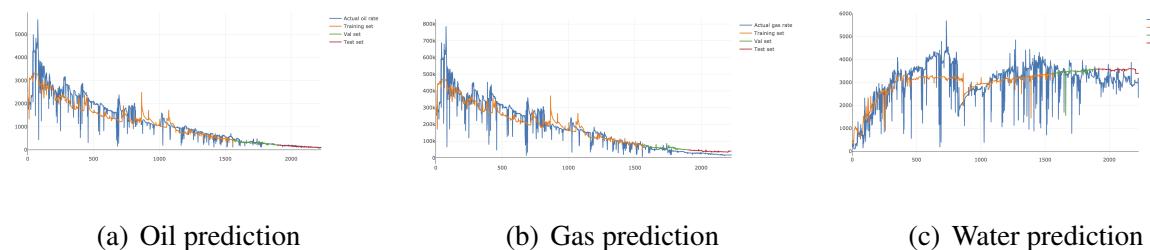


Figure 5.7. Production rate prediction using gated recurrent unit.

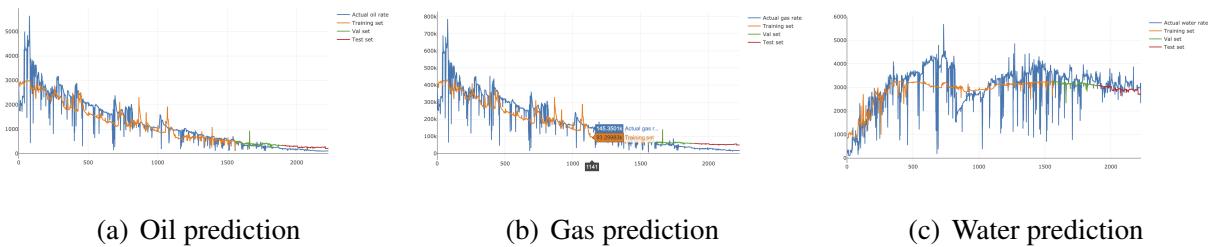


Figure 5.8. Production rate prediction using simple recurrent neural networks.

5.3. Applied long short term memory as the best algorithms

After a comparison of the result with different algorithms, recurrent neural networks seem to be the most appropriate for time series production prediction. Hence, Long short term memory has been chosen to be continued to train to compare with convention and reservoir modeling methods.

5.3.1. Comparison with DCA

A production profile from 01/01/2015 to 07/12/2016 has been used for comparison between LSTM and DCA.

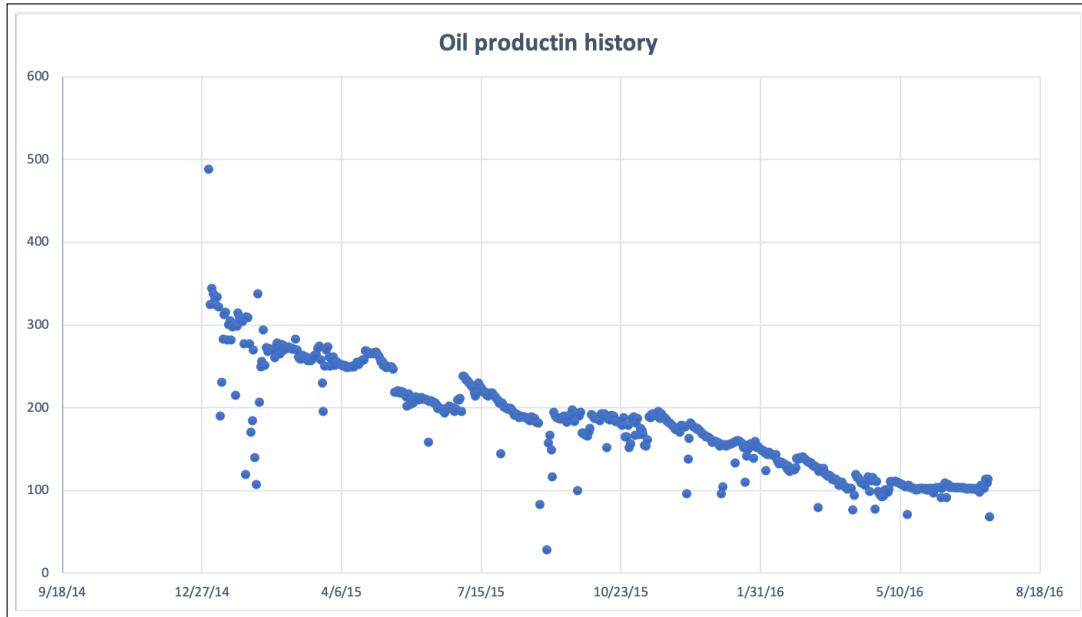


Figure 5.9. History production of well F14

Due to the limit of decline curve analysis methods, DCA is only used when the well is in a boundary-dominated flow condition. This means DCA is possible with F14 and F12 wells prediction and the F14 well is chosen for using decline curve analysis. There are four periods in use for creating decline curves. A comparison between conventional decline curve analysis and LSTM algorithms have shown in figure below:

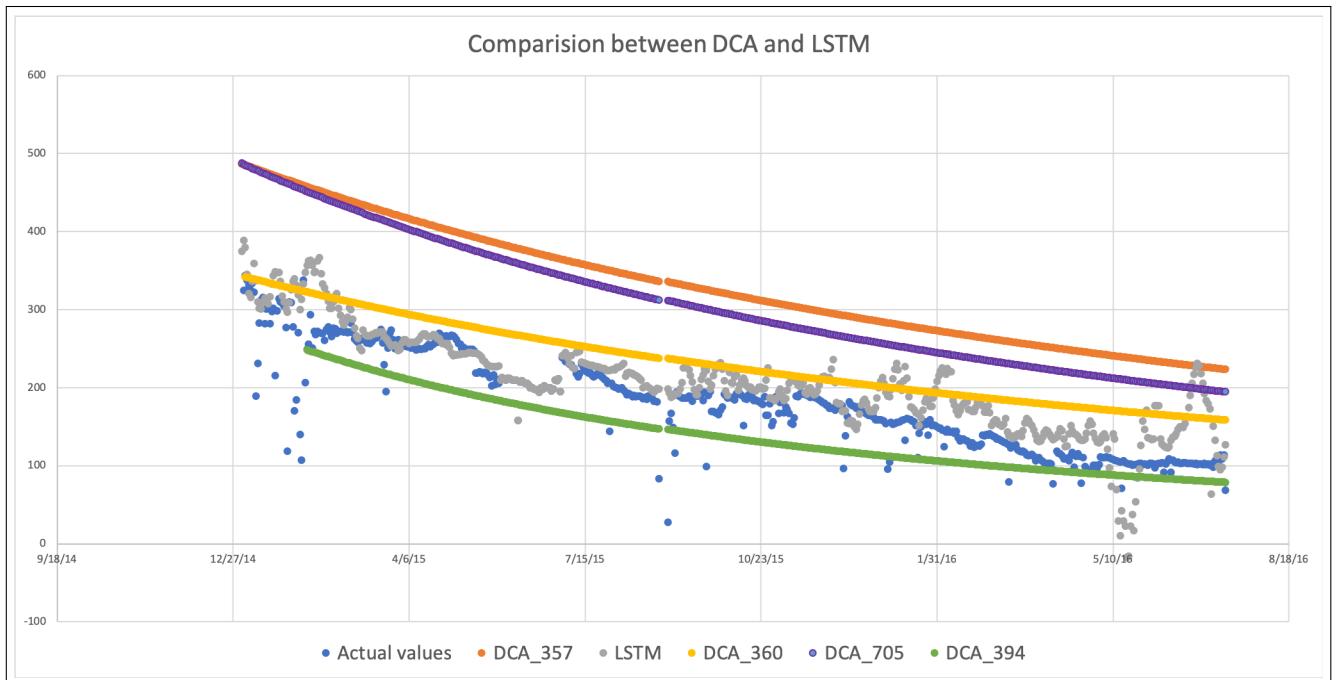


Figure 5.10. Comparison of actual values, LSTM and DCA

Table 5.7. Decline (D) values and time periods

	DCA_357	DCA_360	DCA_394	DCA_705
Time (days)	357	360	394	705
D values	0.00172	0.0017	0.0031	0.0021

Percentage error is used for comparison. The relative error is defined by:

$$\delta = \frac{\Delta x}{x} = \frac{|x_0 - x|}{x} \quad (5.1)$$

where Δx is absolute error. Percentage error is 100% times to relative error.

Table 5.8. Cumulative oil production comparison

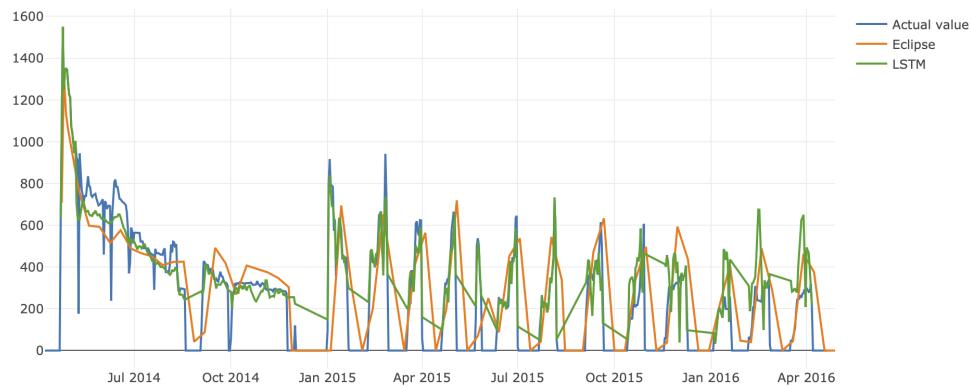
	Actual values	DCA_357	DCA_360	DCA_394	DCA_705	LSTM
Cumulative oil	101424.84	183504.591	130018.689	83019.914	171241.554	107553.954
Error	0%	80.92%	28.2%	18.15%	68.83%	6.04%

Ideally, the optimized error should be as small as possible and could not be excessed 10%.

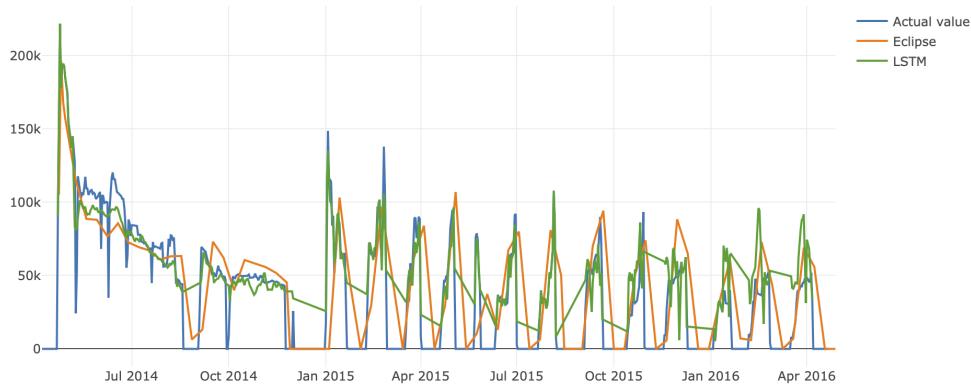
From the table below, LSTM algorithms (6.04% of error) have outperformed decline curve analysis methods in predict cumulative oil production, moreover Figure 5.9 shows that while LSTM fluctuates near the end of production, it has been the closest curve with the actual curve compare with the other decline analysis curve.

5.3.2. Comparison with Eclipse

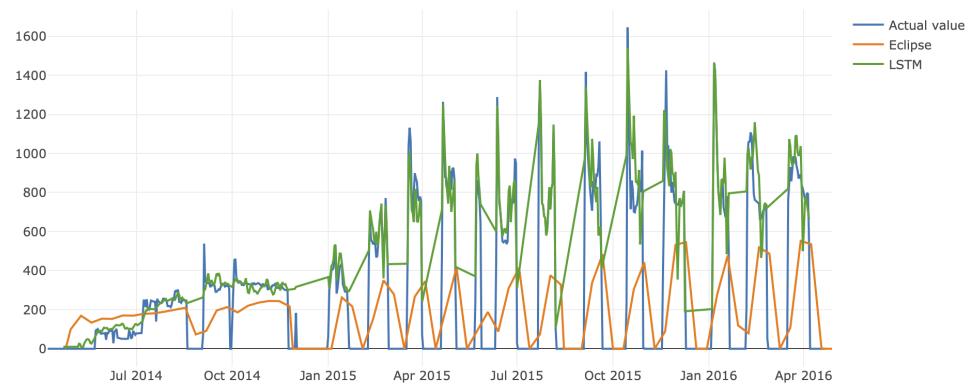
5.3.2.1. Well F1



(a) Oil prediction



(b) Gas prediction



(c) Water prediction

Figure 5.11. Production prediction on well F1

5.3.2.2. Well F11

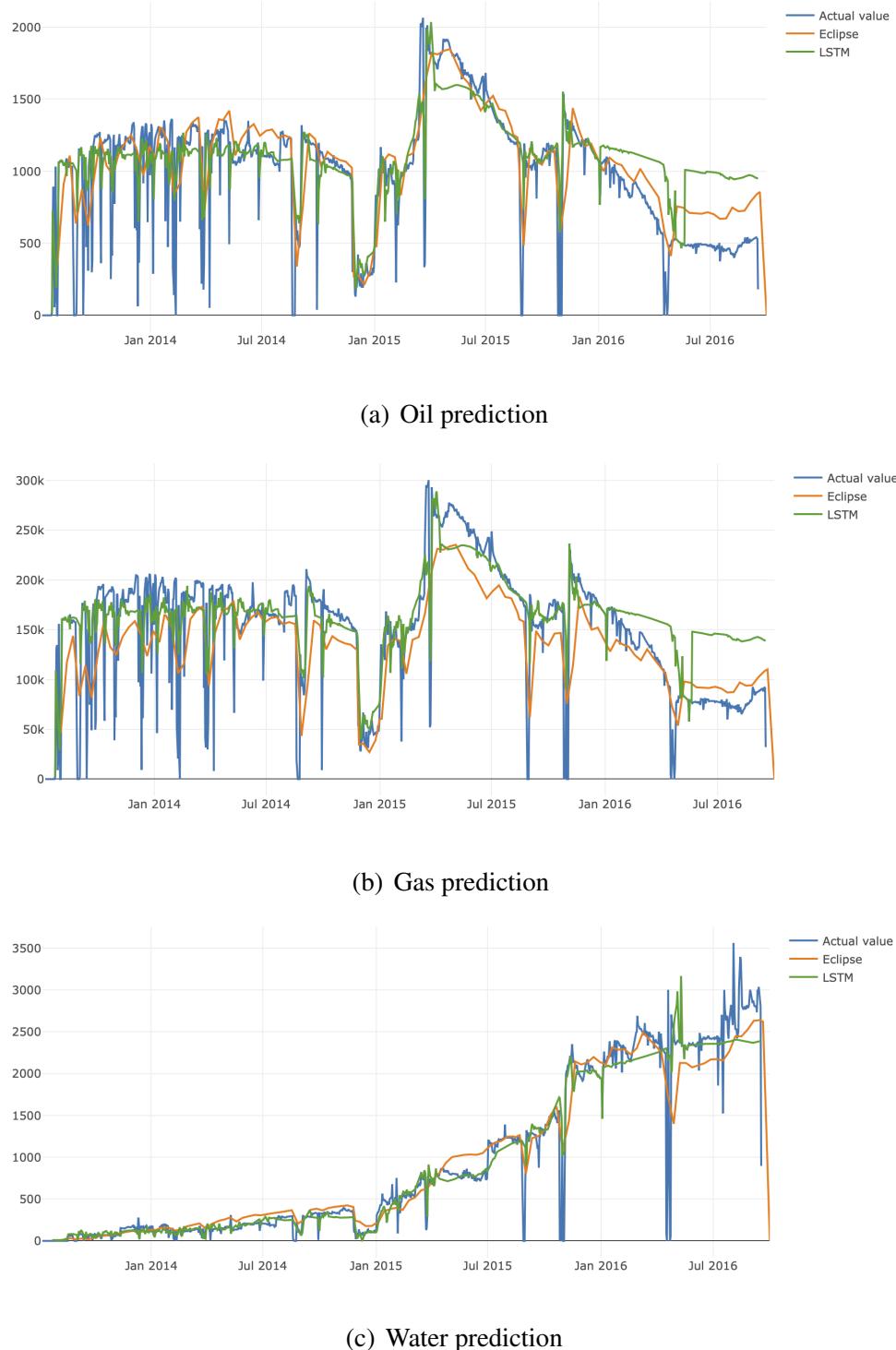


Figure 5.12. Production prediction on well F11

5.3.2.3. Well F12

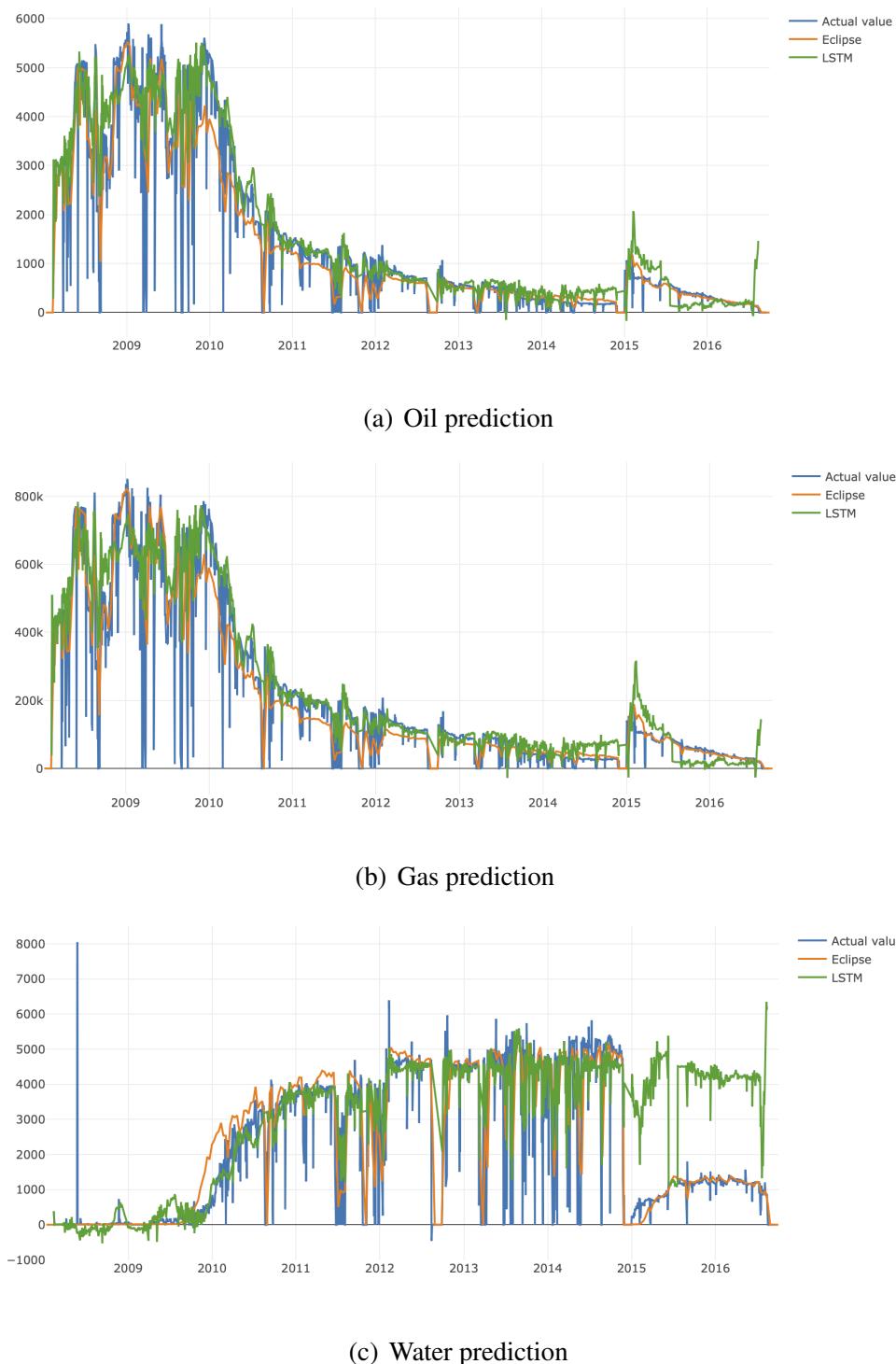


Figure 5.13. Production prediction on well F12

5.3.2.4. Well F14

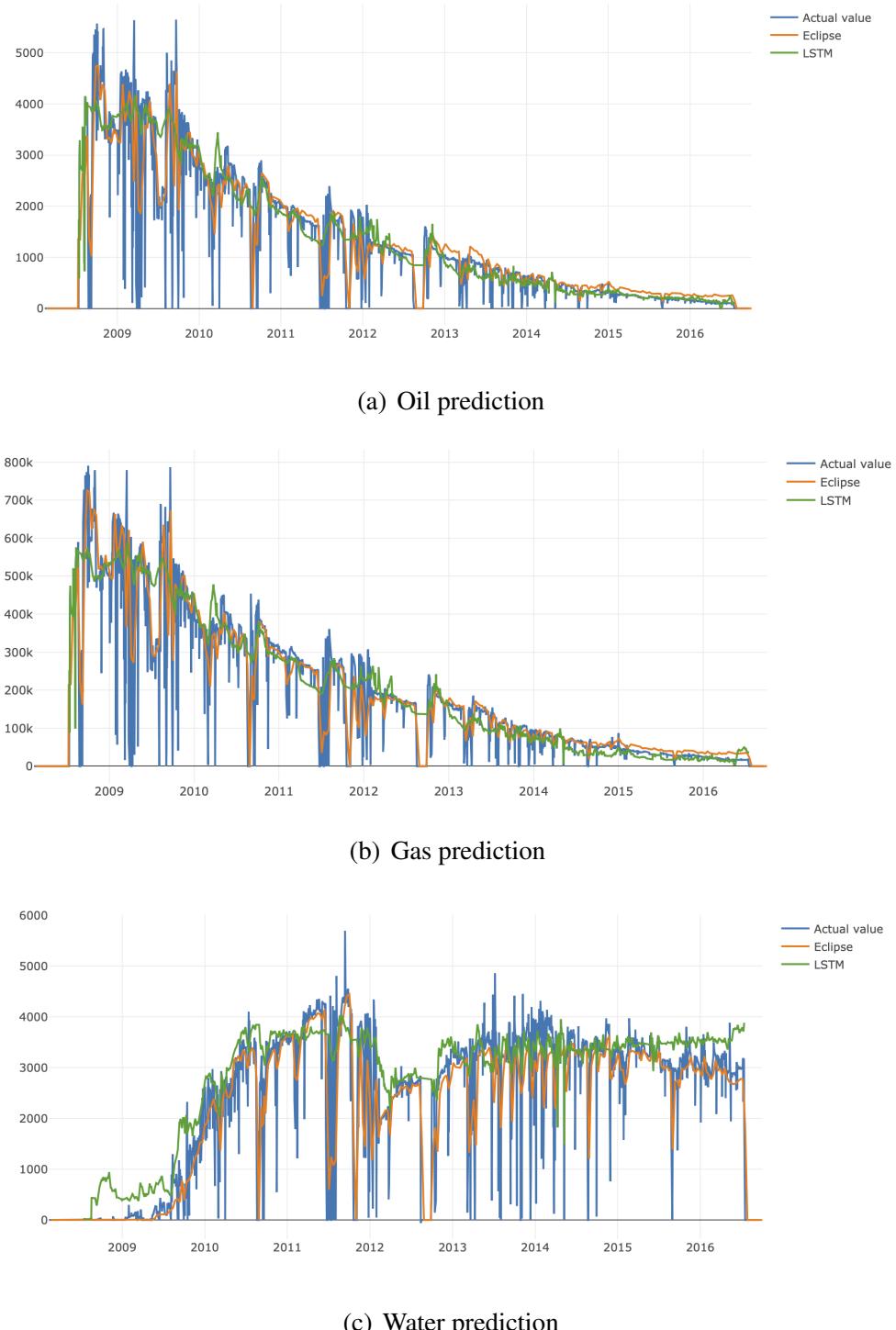


Figure 5.14. Production prediction on well F14

5.3.2.5. Well F15

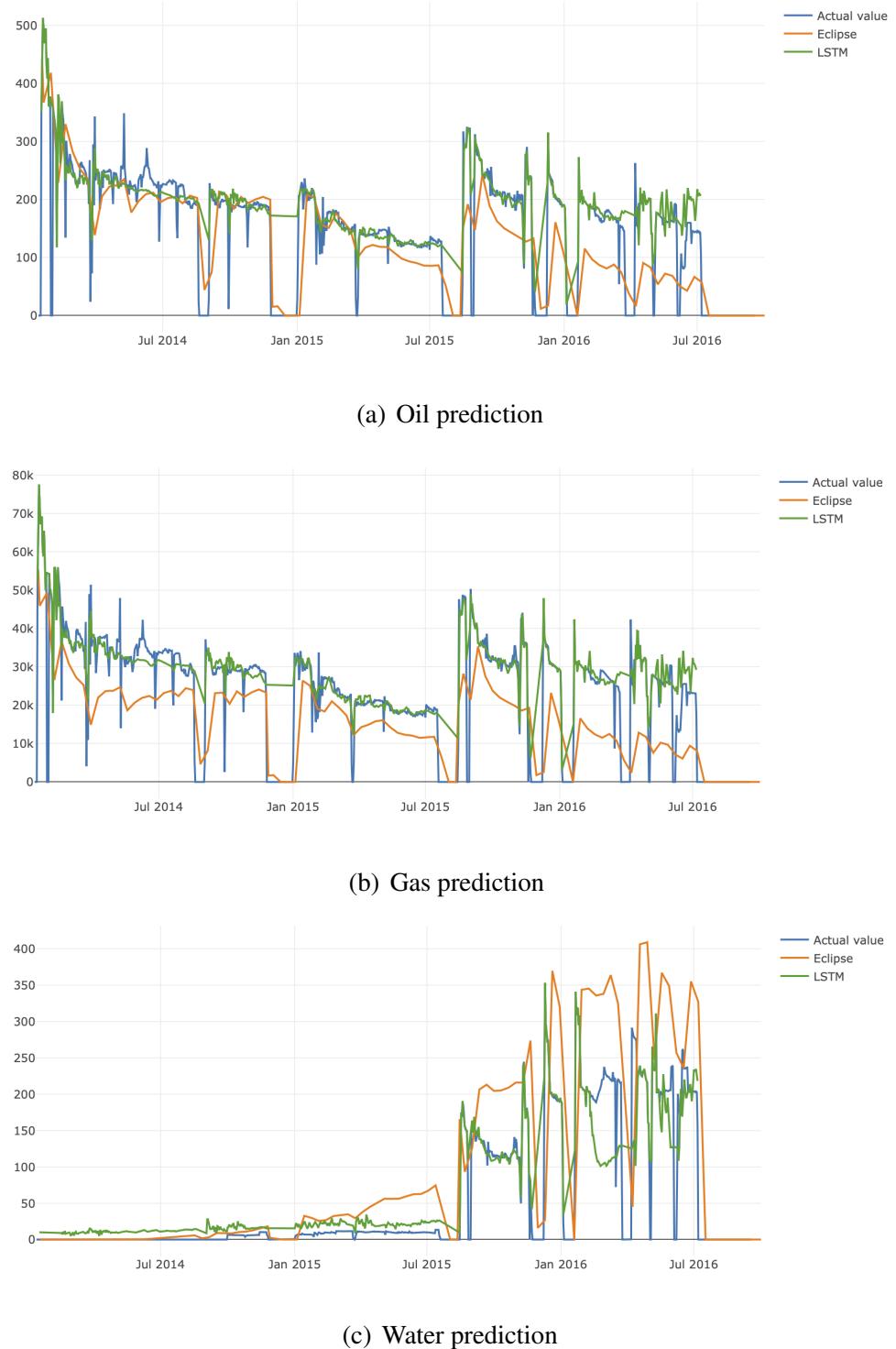


Figure 5.15. Production prediction on well F15

Different from Eclipse simulator where an average of 10 days of production has been matched with actual values, using LSTM memory can predict each day during production period.

According to the five tables 5.11, 5.12, 5.13, 5.14 and 5.15 above, LSTM algorithm has given better results compared with Eclipse model in well F1, F14, and F15. As shown in the figures above, LSTM not only give a better matching with the actual values in these well, but also in prediction they are more likely to catch the trend of production in the future. However, in the well F11 and F12, Eclipse model has better matching. This might be the limit of LSTM algorithms because of the data noise.

Next, cumulative production will be compared between Eclipse output and LSTM. Oil rate will be chosen for this comparison in this study. Gas and water comparison can be found in the appendix for reference.

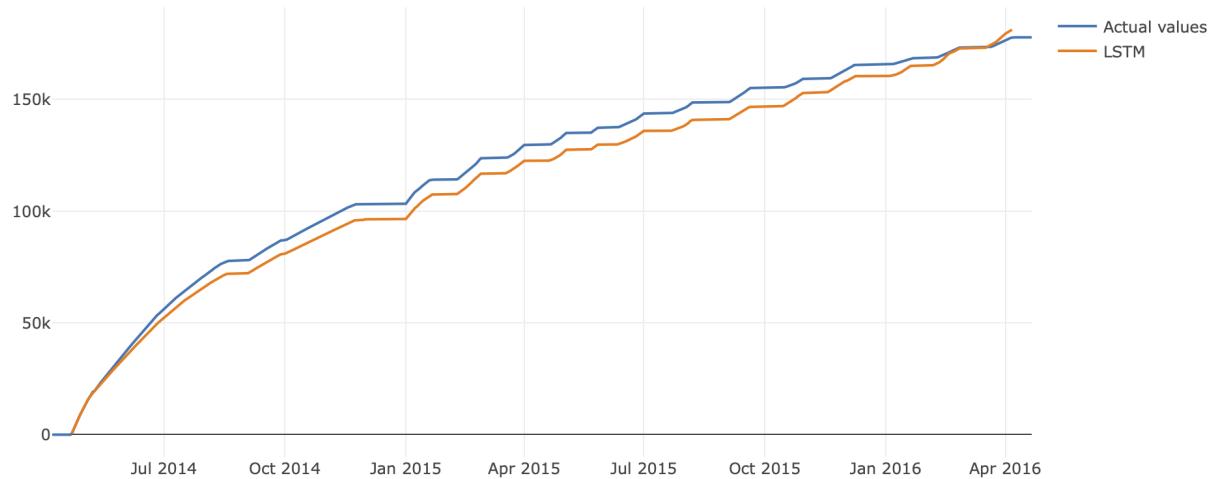


Figure 5.16. Cumulative oil production on F1

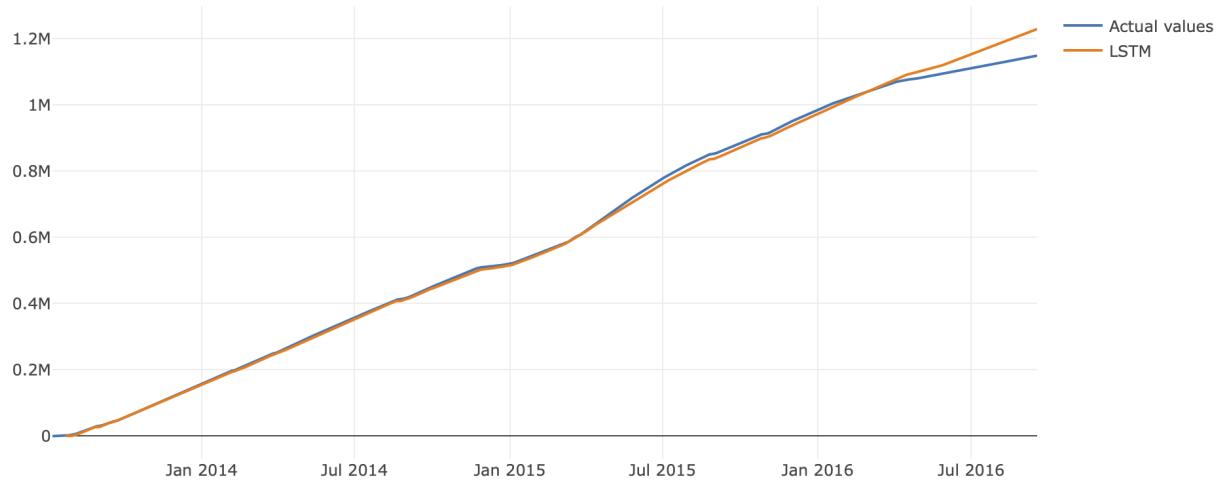


Figure 5.17. Cumulative oil production on F11

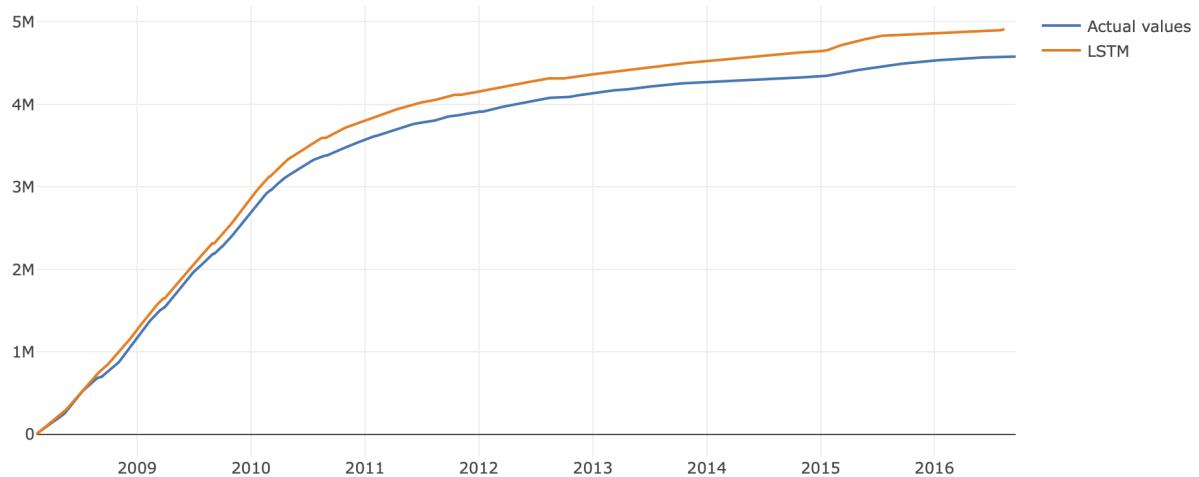


Figure 5.18. Cumulative oil production on F12

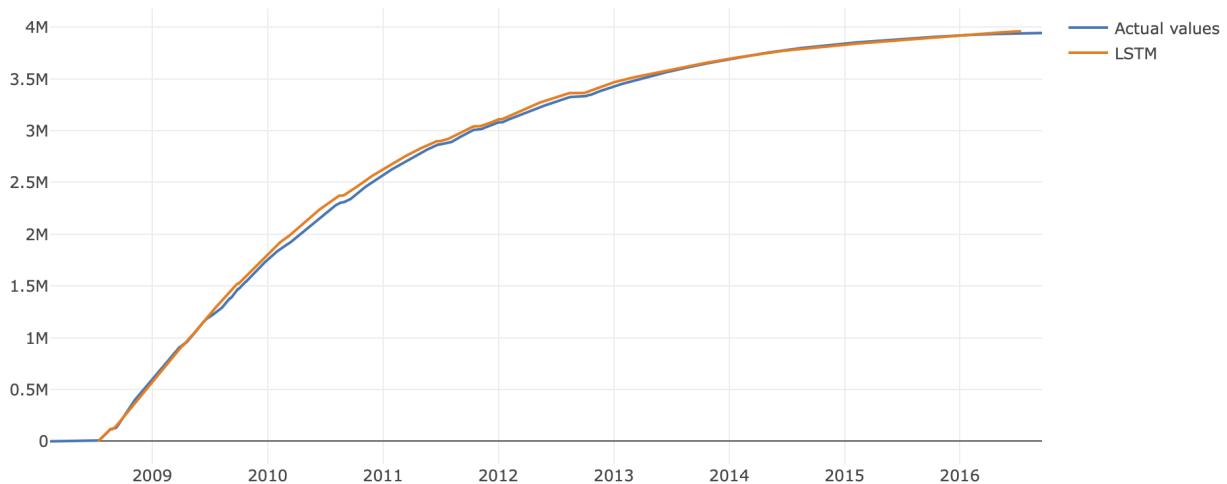


Figure 5.19. Cumulative oil production on F14

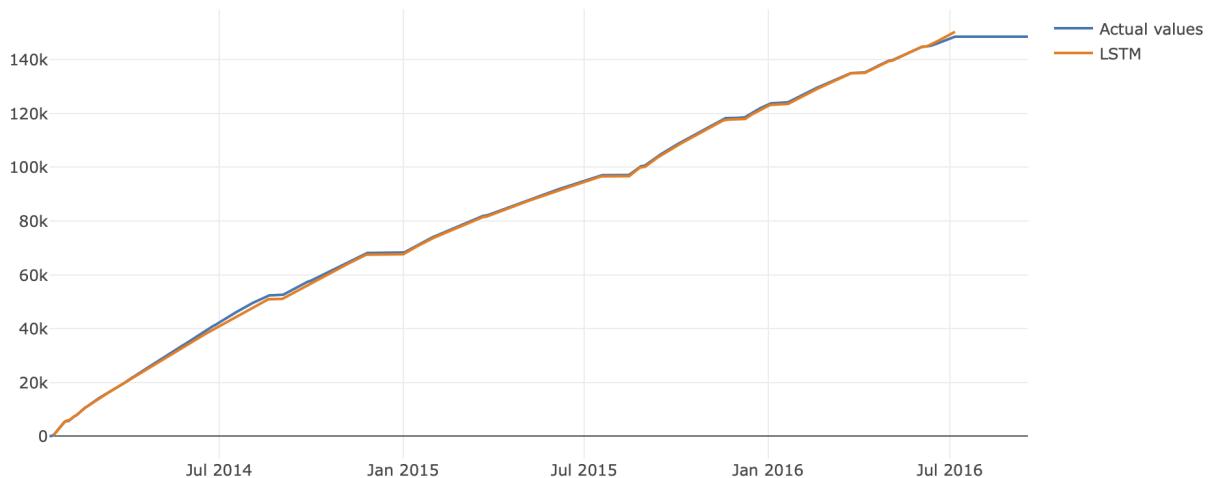


Figure 5.20. Cumulative oil production on F15

Table 5.9. Cumulative oil production by well

	Well F1	Well F11	Well F12	Well F14	Well F15
Actual cumulative production (m3)	177709.3	1227203	4579610	3942233	148518.6
Cumulative oil of LSTM (m3)	181144.8	1224349	4912712	3960222	150325.6
Error (%)	1.93	0.23	7.27%	0.45	1.21

On well F11 and F14, LSTM gives the prediction of cumulative oil production with 0.23%

and 0.45% error respectively. On well F1 and F15, errors are 1.93% and 1.21 and F12 well is 7.27%. These errors are supposed to be acceptable for further using of LSTM algorithm.

5.3.2.6. Total cumulative oil production in comparison

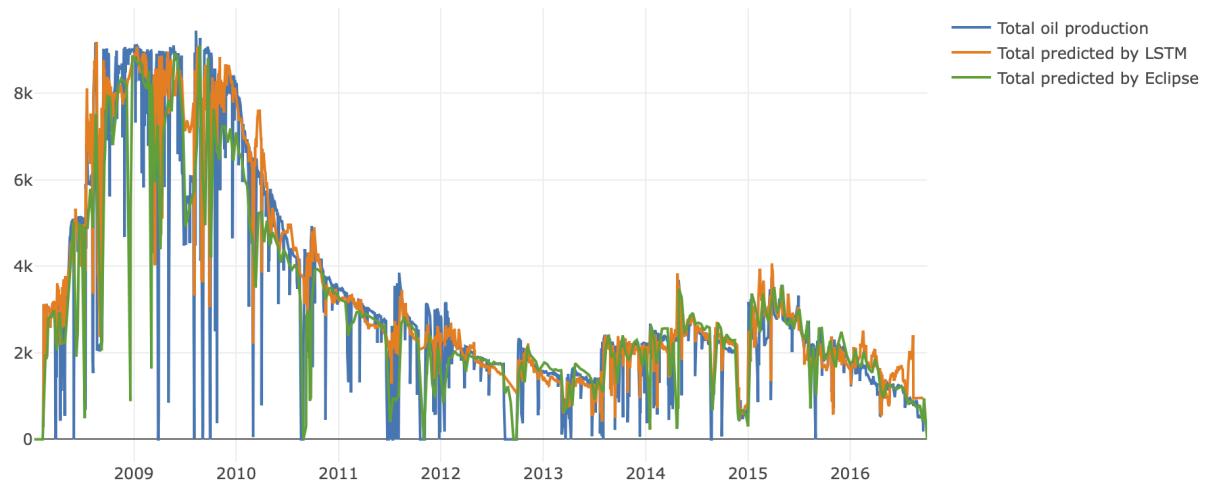


Figure 5.21. Total production by day from 2008 to 2016

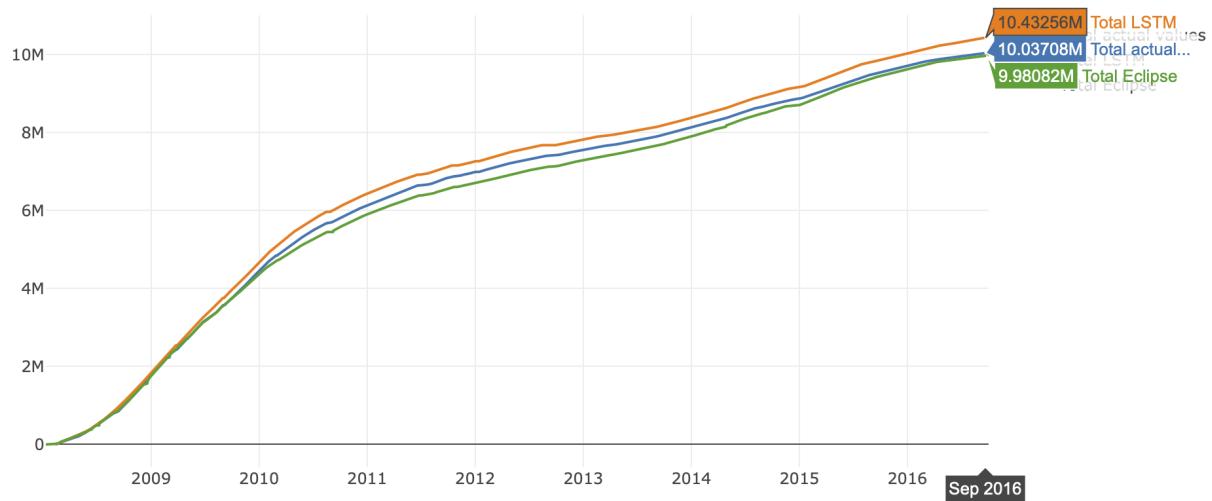


Figure 5.22. Total cumulative oil production comparison

Table 5.10. Total cumulative oil production and errors

	Actual cumulative production	Cumulative oil of LSTM	Cumulative oil of Eclipse
Values	10037081	1043256	9980819
Error(%)	0	3.94	0.56

The Volve reservoir model produced 9980819 m^3 of oil for 3,197 days. The cumulative oil from historical production was 10037081 m^3 . Long short term memory gives the output of 1043256 m^3 , slightly greater than actual values due to the mismatch of well F12 (7.97 % from table 5.9). Although the error given by LSTM is slightly bigger than reservoir model, it still a very good match of production rate.

In addition, compare with building a complex reservoir model, LSTM algorithms just cost less time and effort to construct. Since the objective is to study the potential of artificial intelligence in production prediction, it is practically to use AI in brown field which have a great amount of historical data. But in case of lack of data, we can use it as a tool for production optimization in a short period of time.

Chapter 6. Conclusion

In this thesis, the author has applied some AI technique include machine and deep learning algorithms to predict the production rate of oil well in different scenarios. Different algorithms have been used and comparison between them have shown that while machine learning gain good result in the production phases with the shuffle, the recurrent neural networks have improved the prediction with .

The following conclusions from this study are:

- The machine learning and deep learning algorithms used in the thesis significantly rely on the amount of data to develop the capacity of production prediction.
- The quality of prediction highly depends on the data input. If the input is messy and inconsistent, then the production prediction can be unreliable.
- Since machine learning have gained good result when the data is shuffled, hence in order to predict time series production rate, recurrent neural networks are performed. Machine learning algorithms can also be used for production optimization. The constraint conditions of producing well can be modified and regulated based on a complicated relationship that was acquired from using machine learning algorithms.
- Compare with conventional decline curve analysis (Arps), LSTM has outperformed due to the ability in time-series prediction. In Volve field studies, LSTM is better than reservoir simulation (Eclipse) at well F14 and F15 and close to the accuracy of actual values in all five wells. It demonstrates the ability of using Artificial Intelligence in production forecasting.
- While reservoir modeling needs a great amount of time and efforts to construct a reservoir model with extensive domain knowledge of geology, geophysics, petrophysics, reservoir engineering, and reservoir simulation, LSTM only based the historical data to complete the tasks. So, in this circumstance LSTM is working on the

cutting-edge methods, and need to be studied carefully in the future in order to give better results.

Also, a work flow of using machine learning and deep learning algorithms has been built with detailed explanations.

This study has demonstrated that without domain knowledge in oil and gas industry and complicated physics-based modeling, Artificial Intelligent can help engineer in production forecasting in effective ways. AI has been proven their potential and could be useful tools in oil and gas industry.

Furthermore, in the next study, gas-oil ratio (GOR), water-cut, water gas ratio (WGR) and bottom hole pressure can be predicted.

Bibliography

- [1] Hedong Sun. Advanced Production Decline Analysis and Application. Gulf Professional Publishing (2015).
- [2] Tarek Ahmed. Reservoir Engineering Handbook - Fourth Edition.
- [3] Havard Johnsen Reitan History Matching: Effekten av tilgjengelig informasjon.
- [4] James R. Gilman and Chet Ozgen (Society of Petroleum Engineers) Reservoir Simulation: History Matching and Forecasting.
- [5] Williams, M.A., Keating, J.F., Barghouty, M.F. The stratigraphic method: a structured approach to history-matching complex simulation models. SPE Reservoir Evaluation Engineering 1(2),169–176 (1998).
- [6] Dean S.Oliver and Yan Chen. Recent progress on reservoir history matching: a review.
- [7] Cao, Q., Banerjee, R., Gupta, S., Li, J., Zhou, W., & Jeyachandra, B. 2016. Data Driven Production Forecasting Using Machine Learning Presented at SPE Argentina Exploration and Production of Unconventional Resources Symposium, Buenos Aires, Argentina, 1-3 June.
- [8] Tita Ristanto. Machine Learning applied To Multiphase Production Problems. MS Thesis, Standford University, 2018.
- [9] Boomer, R.J. (1995). Predicting Production Using a Neural Network (Artificial Intelligence Beats Human Intelligence). Society of Petroleum Engineers (SPE 30202).
- [10] Suhag, A., Ranjith, R., and Aminzadeh, F. (2017). Comparison of Shale Oil Production Forecasting using Empirical Methods and Artificial Neural Networks. University of Southern California. SPE ATCE (SPE-187112-MS).

- [11] J. Sun, X. Ma, and M. Kazi (2018), CSE ICON. Comparison of Decline Curve Analysis DCA with Recursive Neural Networks RNN for Production Forecast of Multiple Wells SPE-190104-MS doi: SPE-190104-MS
- [12] Al-Kaabi, A. U. and Lee, W.J. 1990. Using Artificial Neural Networks to Identify the Well Test Interpretation Model. Presented at the Petroleum Computer Conference, Denver, Colorado, 25-28 June. SPE-20332.
- [13] Christopher M. Bishop Pattern Recognition and Machine Learning. Linear Model page 138-139
- [14] Dominique Guillot Introduction to Data Mining and Analysis Decision trees. April 6, 2016.
- [15] Ho, Tin Kam (1995). Random Decision Forests (PDF). Proceedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, QC, 14–16 August 1995. pp. 278–282.
- [16] D. O. Hebb. The Organization of Behavior: A Neuropsychological Approach. John Wiley & Sons, 1949.
- [17] Y.Bengio, P.Simard, and P.Frasconi. Learning long-term dependencies with gradient descent is difficult. Neural Networks, IEEE Transactions on 5, no. 2 (1994): 157-166.
- [18] Ian Goodfellow and Yoshua Bengio and Aaron Courville. Deep learning text book. An MIT press book.
- [19] Hastie, T. et al., (2001). The Elements of Statistical Learning. Springer New York Inc., New York, NY, USA.
- [20] Pierre Geurts and Damien Ernst and Louis Wehenkel. Extremely randomized trees
- [21] Friedman, J.H. (2011). Greedy Function Approximation: A Gradient Boosting Machine. Annals of Statistics. JSTOR.

- [22] Sklearn Library. Machine learning in Python Journal of machine learning research, volume 12, 2825–2830, 2011.
- [23] Chen, Tianqi and Guestrin, Carlos. XGBoost: A Scalable Tree Boosting System. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 2016.
- [24] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, Tie-Yan Liu. "LightGBM: A Highly Efficient Gradient Boosting Decision Tree". Advances in Neural Information Processing Systems 30 (NIPS 2017), pp. 3149-3157.
- [25] Martens, James, and IlyaSutskever. Learning recurrent neuralnetworks with hessian-free optimization. In Proceedings of the 28th International Conference on Machine Learning (ICML-11), pp. 1033-1040. 2011.
- [26] Understanding LSTM Networks
- [27] Volve dataset by Equinor company
- [28] Wikipedia feature engineering
- [29] M. Korjani, Andrei Popa, Eli Grijalva, Steve Cassidy and I. Ershaghi. A New Approach to Reservoir Characterization Using Deep Learning Neural Networks.
- [30] E. E. Okpo, A. Dosunmu, and B. S. Odagme, University of Port Harcourt Artificial Neural Network Model for Predicting Wellbore Instability.
- [31] Srinath Madasu and Keshava P. Rangarajan, Halliburton Deep Recurrent Neural Network DRNN Model for Real-Time Multistage Pumping Data.
- [32] Jung Chan Choi, Elin Skurtveit, and Lars Grande, NGI Deep Neural Network Based Prediction of Leak-Off Pressure in Offshore Norway.
- [33] Tanaka, Takashi, Shinoda, Takeshi Kyushu University A Method for Extracting the Work Status in Shipyard Using Deep Neural Networks.

- [34] Maarten V. de Hoop, Rice University Deep neural network architectures arising in seismic inverse problems.
- [35] A.G. Mihajlov, S.S. Shubin, A.V. Alferov, R.N. Imashev, V.U. Yamaliev Improvement of efficiency of diagnostics of rod pumps with use of deep neural networks.
- [36] Yunzhi Shi, Xinming Wu and Sergey Fome. Automatic salt-body classification using a deep convolutional neural network.
- [37] Sarblund Haroon, Sergey Alyamkin, and Ramachandra Shenoy, AlphaX Decision Sciences LLC Application of Convolutional and Deep Neural Networks for GPU Based Seismic Interpretations.
- [38] Naif Alqahtani, Ryan T. Armstrong, and Peyman Mostaghimi. Deep Learning Convolutional Neural Networks to Predict Porous Media Properties.
- [39] Zhining Liu, Chenyung Song, Bin She, Kunhong Li, Xingmiao Yao, Guangmin Hu. Visual explanations from convolutional neural networks for fault detection.

Appendix A. Preprocessing and data used in thesis

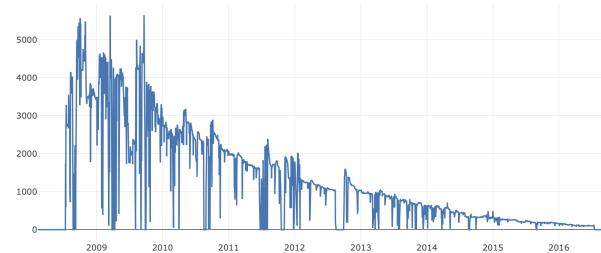


Figure A.1. Data before feature engineering



Figure A.2. Data after feature engineering

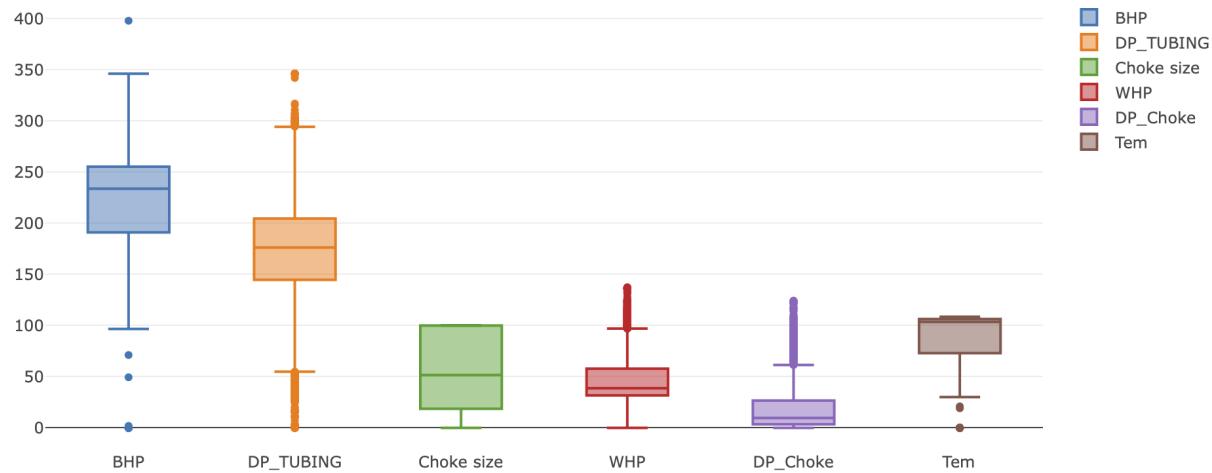


Figure A.3. Box plot of input

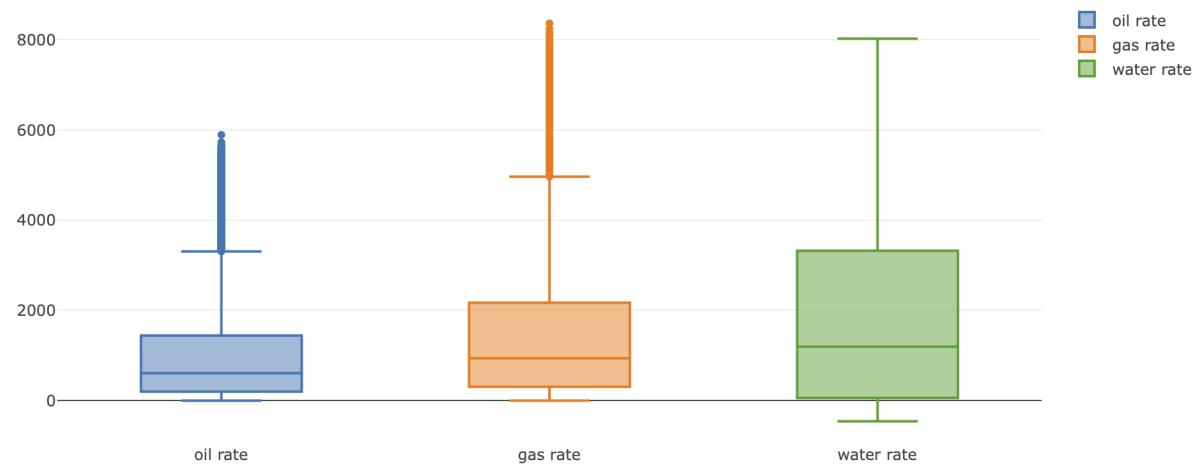


Figure A.4. Box plot of production rate

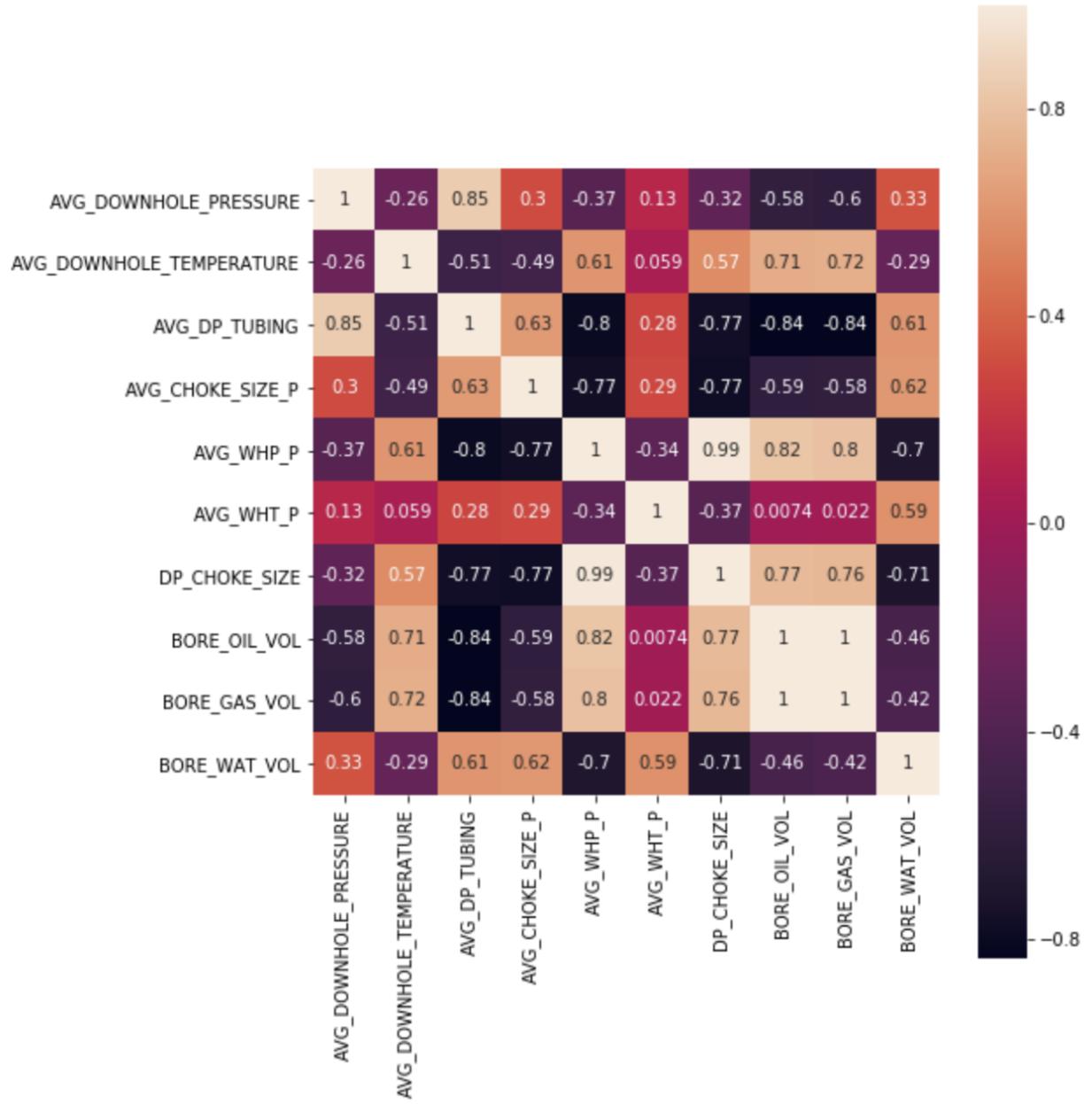


Figure A.5. Correlation maxtrix of variables

Table A.1. Sample data for applying machine learning and deep learning

DATE	BHP	BHT	DP	CS(%)	WHP	WHT	OIL	GAS	WAT
1/30/09	257.44	105.34	163.29	35.30	94.15	61.05	4535.43	649388.07	298.19
2/11/09	261.48	105.36	164.35	34.70	97.13	65.80	4379.88	629307.34	143.54
2/20/09	264.39	105.41	166.21	34.78	98.17	64.99	4509.07	638750.17	108.74
2/22/09	266.71	105.40	166.27	34.05	100.44	67.33	4319.02	612912.62	106.60
2/23/09	266.67	105.41	166.51	34.40	100.15	66.99	4417.66	625514.01	117.37
2/25/09	269.71	105.36	166.93	31.05	102.78	69.95	3226.61	460948.01	118.99
2/26/09	268.34	105.43	167.28	34.31	101.06	67.85	4411.90	628668.27	134.19
2/27/09	268.75	105.44	167.40	34.31	101.35	68.19	4376.91	625510.25	152.76
...
...
6/28/16	269.88	100.17	238.91	45.13	30.97	5.45	102.26	16444.37	2973.66
6/29/16	269.83	100.19	238.94	45.20	30.89	5.37	102.38	16458.56	2980.12
6/30/16	269.83	100.20	238.89	45.16	30.93	5.40	101.47	16397.18	2985.45
7/1/16	269.89	100.20	238.91	45.16	30.98	5.32	102.89	16369.86	2979.13
7/2/16	269.79	100.22	238.87	45.18	30.93	5.39	101.58	16365.91	2992.00
7/3/16	269.79	100.23	238.83	45.11	30.96	5.44	102.43	16269.74	2976.87
7/4/16	269.78	100.24	238.82	45.08	30.96	5.44	100.67	16263.02	2990.10
7/5/16	269.77	100.25	238.80	45.08	30.97	5.43	101.88	16284.48	2974.30
7/7/16	266.20	100.31	238.44	93.42	27.76	2.19	106.19	17427.78	3172.96
7/8/16	266.04	100.33	238.47	100.00	27.56	2.00	106.30	17541.20	3187.95
7/9/16	268.81	100.30	239.08	82.19	29.73	4.11	102.09	16681.29	2326.24
7/10/16	265.92	100.34	238.40	100.00	27.52	1.96	113.38	18753.12	3185.47
7/11/16	267.77	100.32	238.64	91.16	29.13	3.41	108.84	17979.28	3056.29
7/12/16	266.00	100.35	238.27	100.00	27.73	1.94	113.84	18543.76	3148.91

Appendix B. Hyperparametes optimization

Table B.1. Hyperparametes to optimize machine learning algorithms

No	Model	Hyperparametes
1	Linear Regression	Nomalization
2	Ridge	alpha (0.05), solver('auto')
3	KNeighbors Regression	n_neighbors (3),
4	Support Vector Regression	kernel ('rbf), C (1e1) , gamma = 10
5	Decision Tree Regression	max_depth (10)
6	Random Forest Regression	n_estimators = 200, max_depth (8),,max_features('sqrt')
7	Extra Tree Regression	n_estimators=200, min_samples_split(25), min_samples_leaf(35)
8	AdaBoost Regression	n_estimators = 50
9	Gradient Boosting Regression	n_estimators(300), learning_rate (0.005),,max_depth(4)
10	XgBoost Regression	n_estimators(220), learning_rate(0.05), max_depth(3)
11	LightGBM Regression	n_estimators(720), learning_rate (0.05)

Table B.2. Hyperparameters to optimize deep learning algorithms

No	Hyperparameters	Type
1	Activation function	tanh, ReLU, sigmoid, linear
2	Dropout (for DNN)	0.1 - 0.9, 0.4 , 0.5
3	Hidden layers (in NNs)	0 - 4
4	Hidden layer nodes	10 - 500, 8 ,20, 50
5	Loss function	MSE, MAE, RMSE
6	Metrics	MAE, MSE
7	Optimizer	SGD, LBFGS, Adagrad, Adam, RMSProp
8	Momentum	None, regular, Nesterov
9	Learning rate	0.0001, 0.001, 0.005, 0.01, 0.03, 0.3
10	Learning rate decay	0 - 1e-3, 1e-6
11	Regularization	None, L1, L2, EarlyStopping
12	Hyperparameters search	GridSearch, GridsearchCV

Appendix C. Machine learning result

Table C.1. Score and error of machine learning algorithms on gas production

No	Model	Validation Score	Validation Error	Test Score	Test Error
1	Linear Regression	0.8539	0.00577	0.8011	0.00599
2	Ridge	0.8187	0.00619	0.8477	0.0057
3	KNeighbors Regression	0.9323	0.00231	0.9741	0.000967
4	Support Vector Regression	0.8610	0.00474	0.8817	0.00442
5	Decision Tree Regression	0.9452	0.00187	0.9625	0.0014
6	Random Forest Regression	0.9652	0.00155	0.9545	0.0013
7	Extra Tree Regression	0.9506	0.00168	0.9821	0.00068
8	AdaBoost Regression	0.9056	0.00322	0.9195	0.00301
9	Gradient Boosting Regression	0.9685	0.00112	0.9474	0.00188
10	XgBoost Regression	0.9256	0.00253	0.955	0.00168
11	LightGBM Regression	0.8980	0.00355	0.9496	0.00204

Table C.2. Score and error of machine learning algorithms on water production

No	Model	Validation Score	Validation Error	Test Score	Test Error
1	Linear Regression	0.6133	0.0101	0.5228	0.013
2	Ridge	0.599	0.0105	0.523	0.013
3	KNeighbors Regression	0.9032	0.00254	0.8889	0.00303
4	Support Vector Regression	0.8501	0.00394	0.8387	0.00477
5	Decision Tree Regression	0.881	0.00313	0.7822	0.00595
6	Random Forest Regression	0.9197	0.00211	0.8904	0.00299
7	Extra Tree Regression	0.9057	0.00248	0.9015	0.00268
8	AdaBoost Regression	0.7418	0.00679	0.7473	0.0069
9	Gradient Boosting Regression	0.9071	0.00244	0.9124	0.00239
10	XgBoost Regression	0.8819	0.0031	0.8856	0.00312
11	LightGBM Regression	0.8481	0.00399	0.8591	0.00385

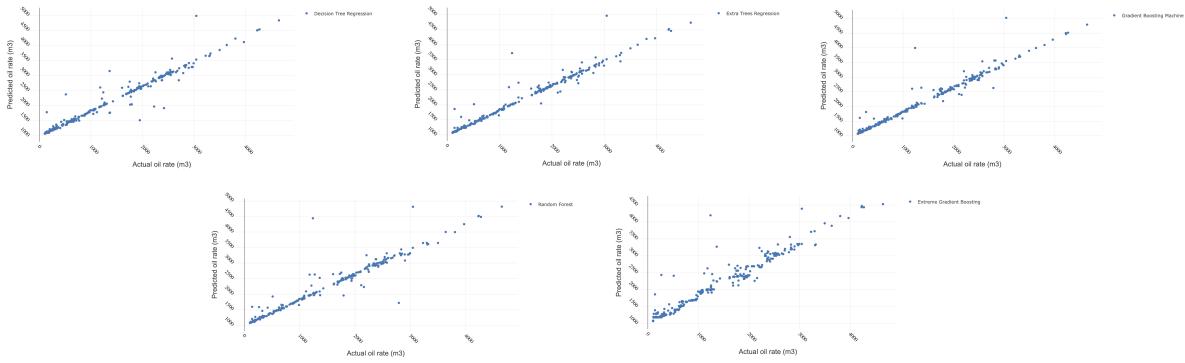


Figure C.1. Best five model on oil prediction

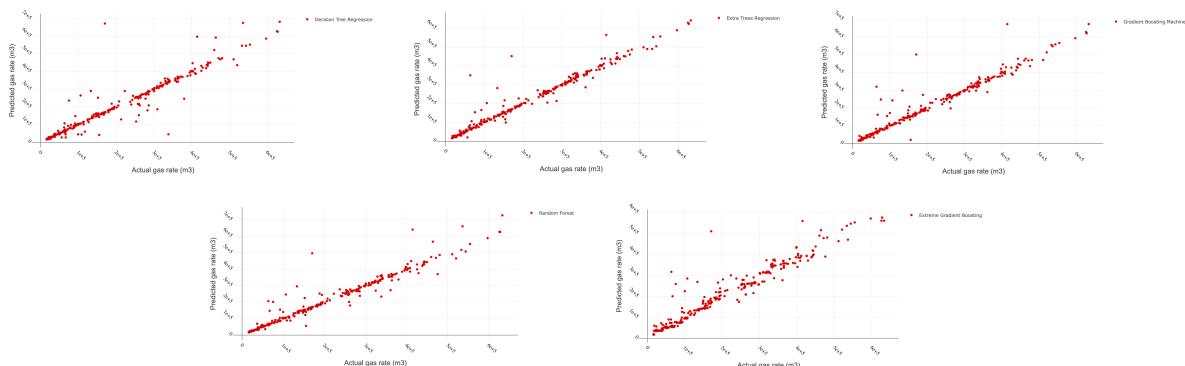


Figure C.2. Best five model on gas prediction

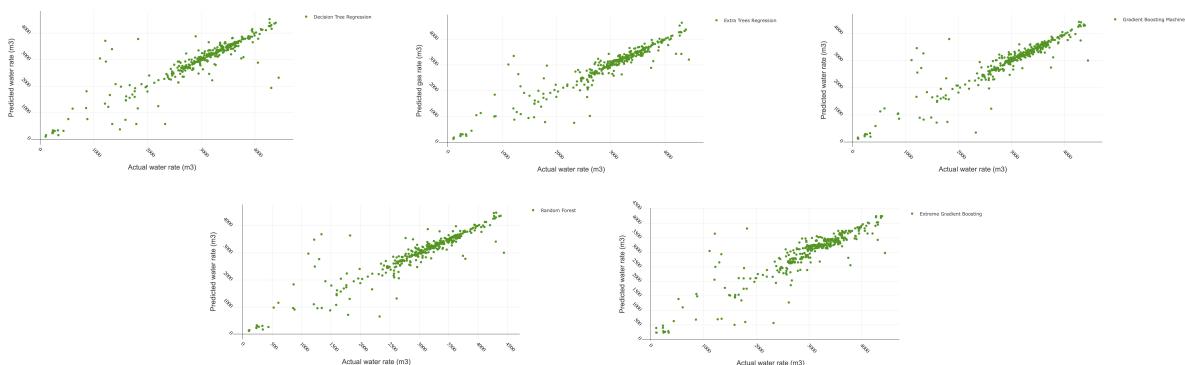
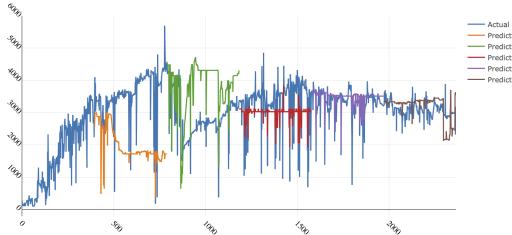
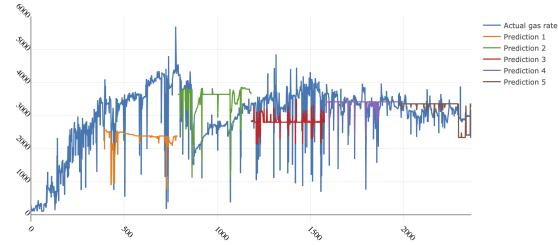


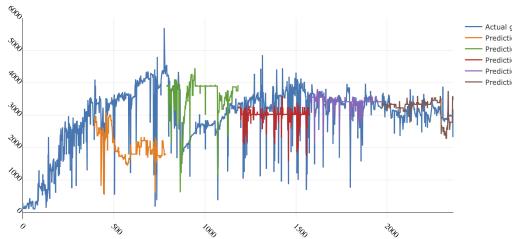
Figure C.3. Best five model on water predict



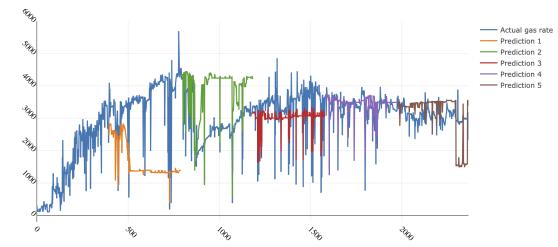
(a) Extreme gradient boosting regression



(b) Extra trees regression

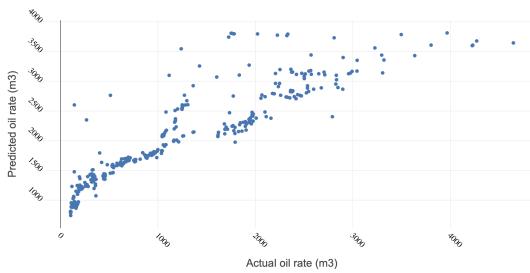


(c) Light GBM regression

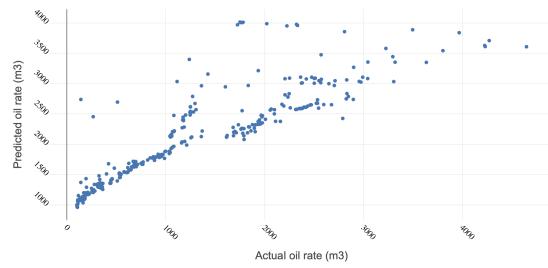


(d) Random forest regression

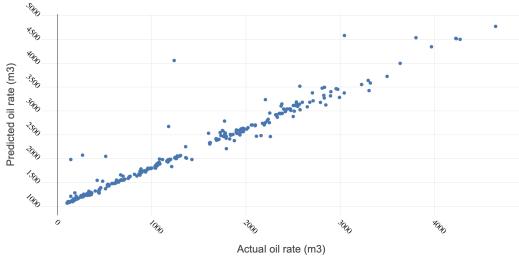
Figure C.4. Water prediction using time series cross-validation method.



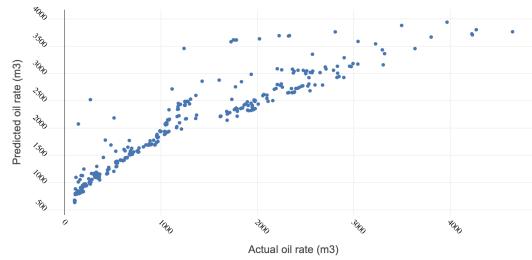
(a) Oil prediction using linear regression



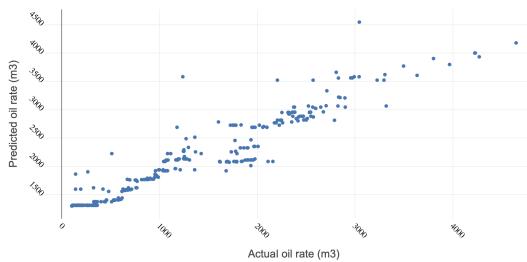
(b) Oil prediction using ridge



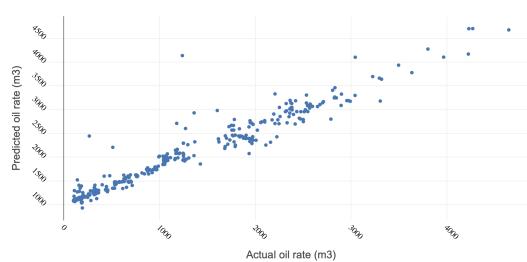
(c) Oil prediction using Kneighbors regression



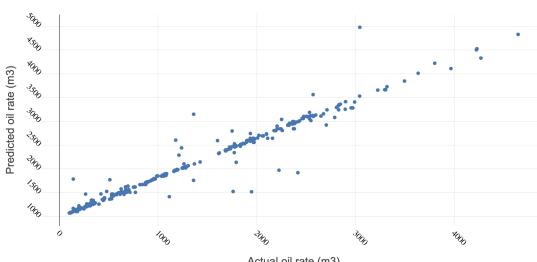
(d) Oil prediction using support vector regression



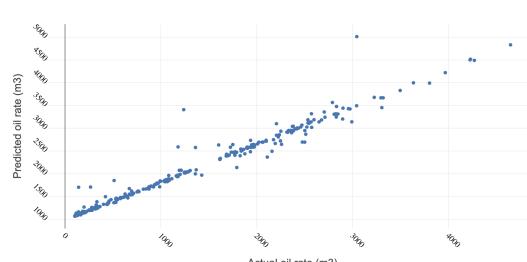
(e) Oil prediction using Ada boost regression



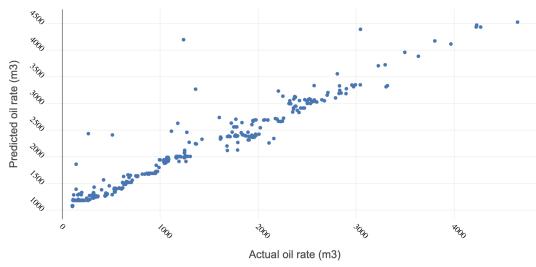
(f) Oil prediction using light gradient boosting machine



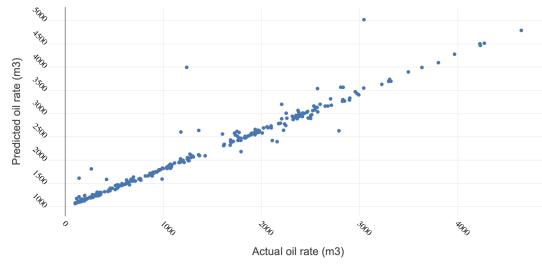
(g) Oil prediction using decision tree regression



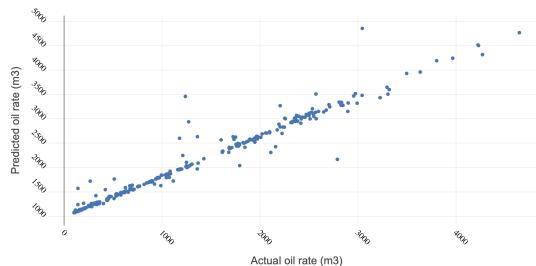
(h) Oil prediction using extra trees regression



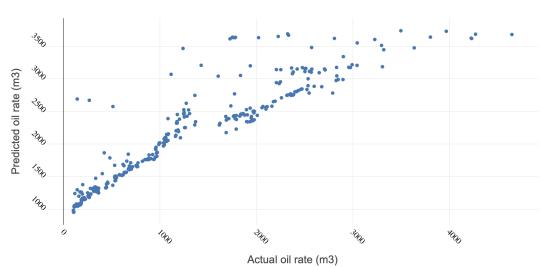
(i) Oil prediction using extreme gradient boosting



(j) Oil prediction using gradient boosting machine



(k) Oil prediction using random forest regression



(l) Oil prediction using multi-layers perceptron

Appendix D. Results of LSTM on five well

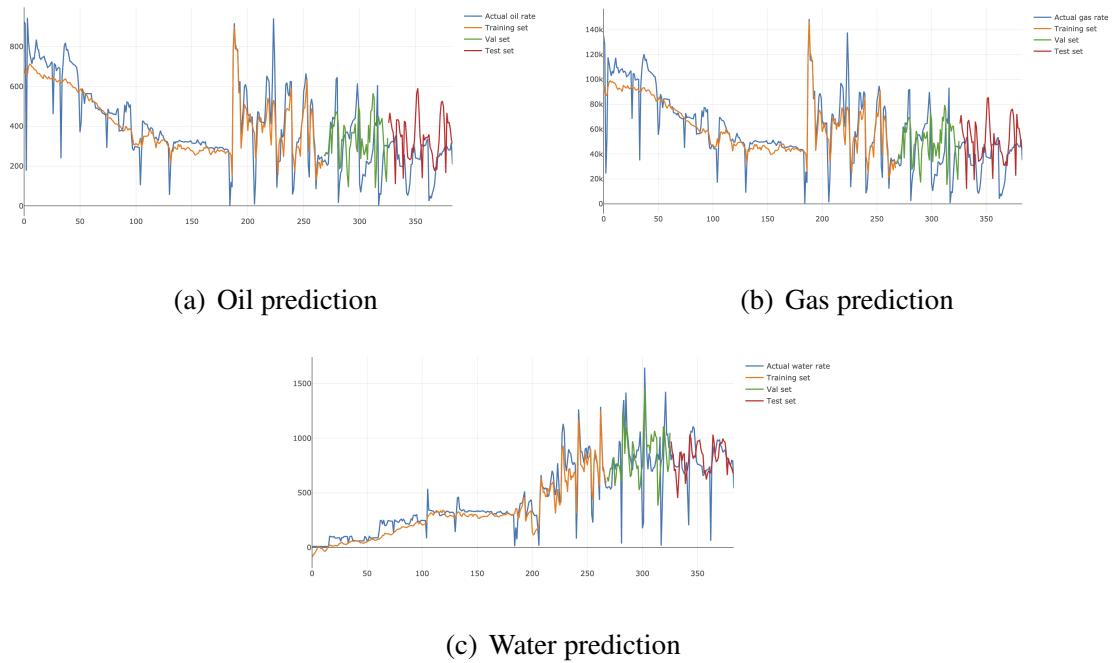
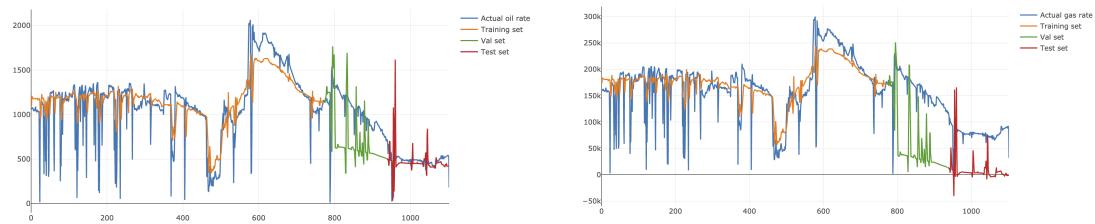
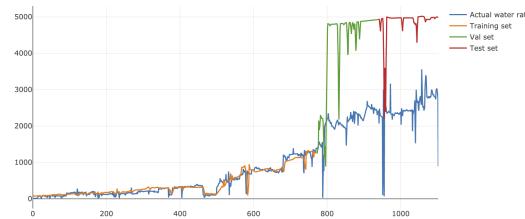


Figure D.1. Production prediction on well F1



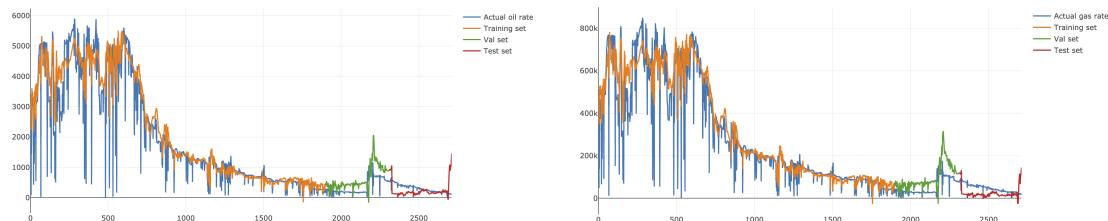
(a) Oil prediction

(b) Gas prediction



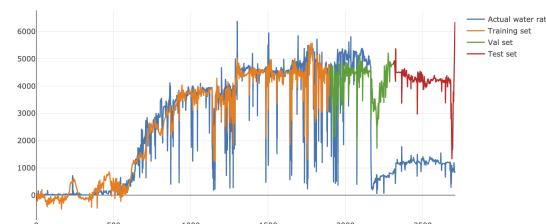
(c) Water prediction

Figure D.2. Production prediction on well F11



(a) Oil prediction

(b) Gas prediction



(c) Water prediction

Figure D.3. Production prediction on well F12

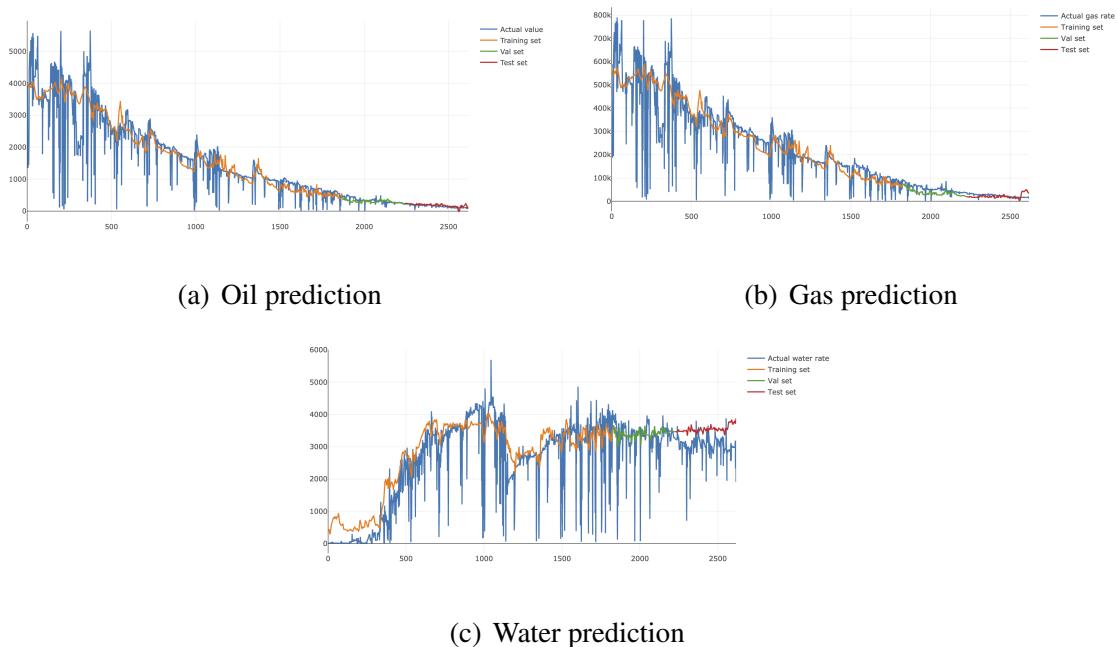


Figure D.4. Production prediction on well F14

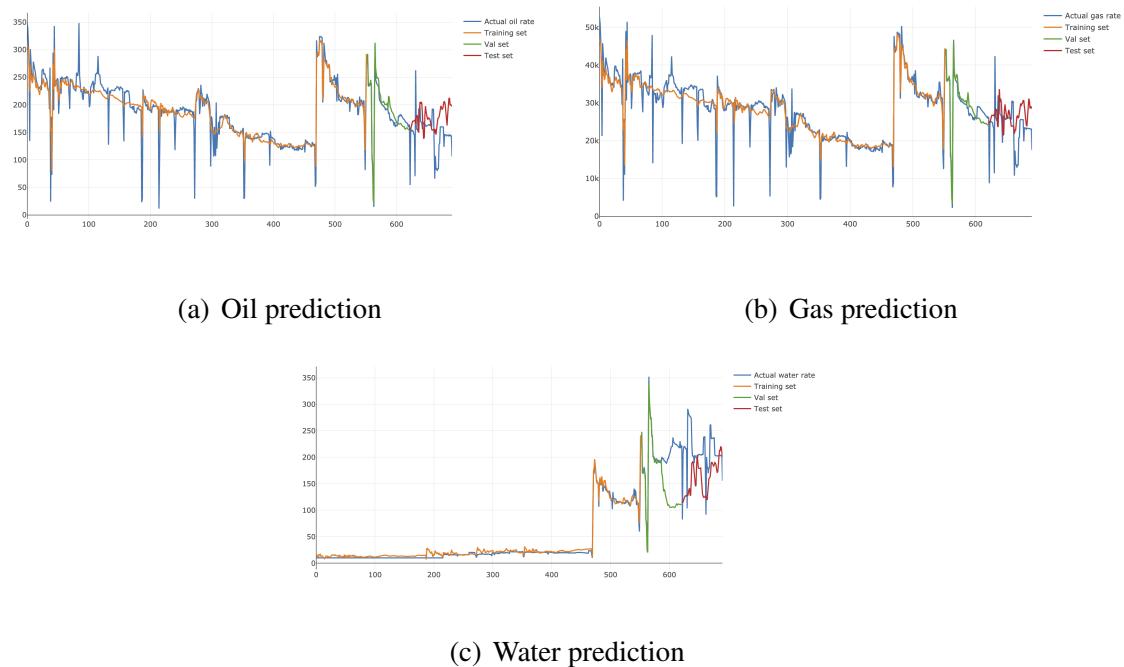


Figure D.5. Production prediction on well F15

Appendix E. Results of LSTM cumulative gas and water

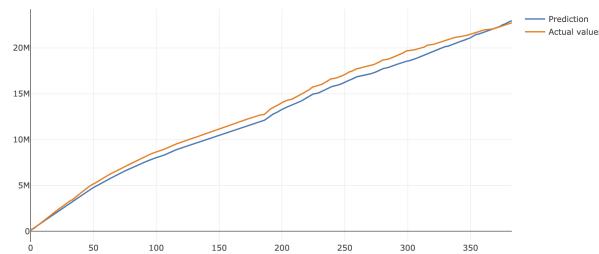


Figure E.1. Cumulative gas prediction F1

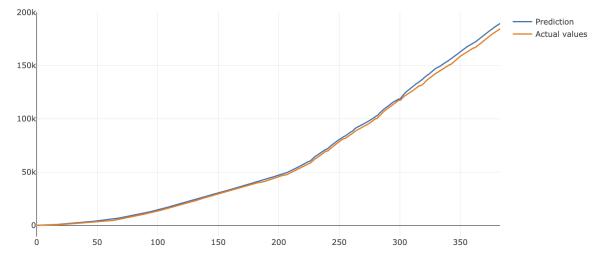


Figure E.2. Cumulative water prediction F1

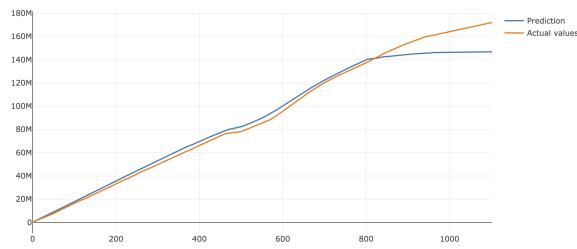


Figure E.3. Cumulative gas prediction F11

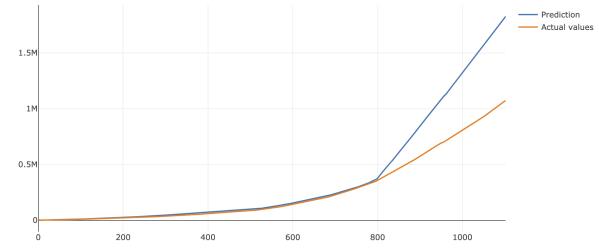


Figure E.4. Cumulative water prediction F11

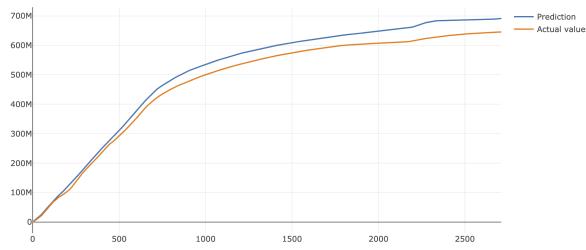


Figure E.5. Cumulative gas prediction F12

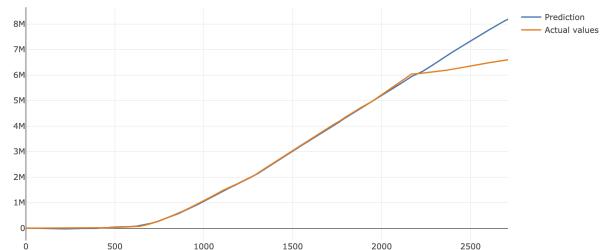


Figure E.6. Cumulative water prediction F12

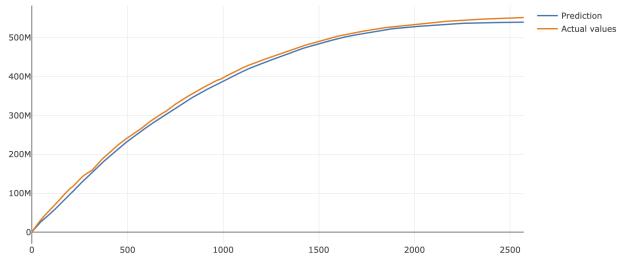


Figure E.7. Cumulative gas prediction F14

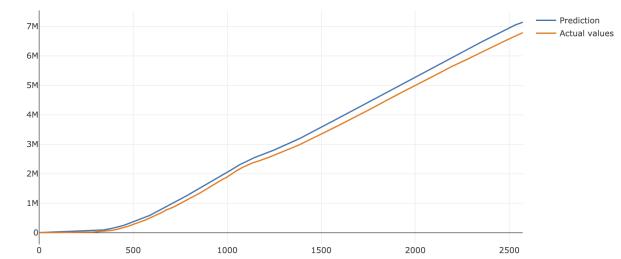


Figure E.8. Cumulative water prediction F14

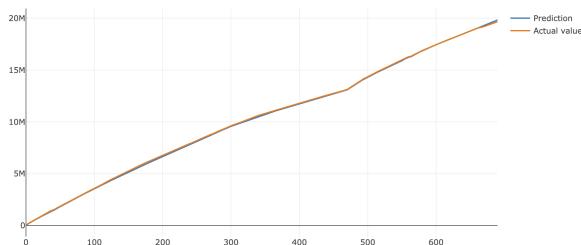


Figure E.9. Cumulative gas prediction F15

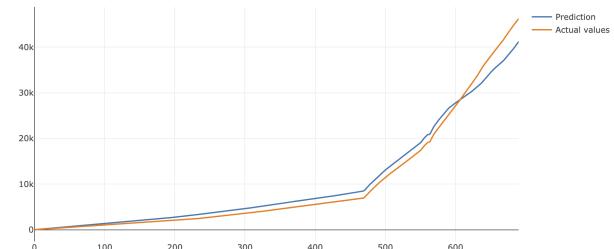


Figure E.10. Cumulative water prediction F15

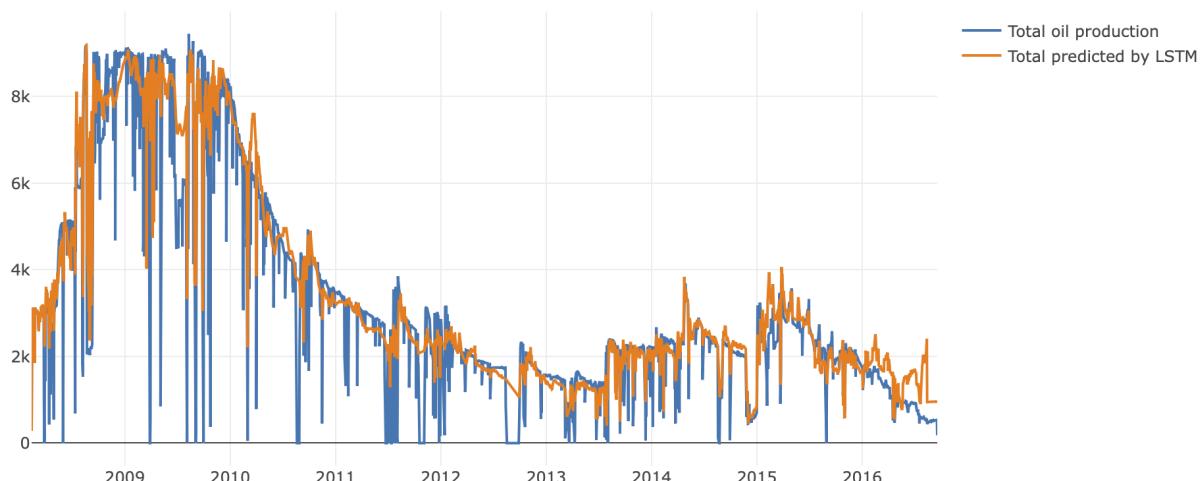


Figure E.11. Total oil production by day

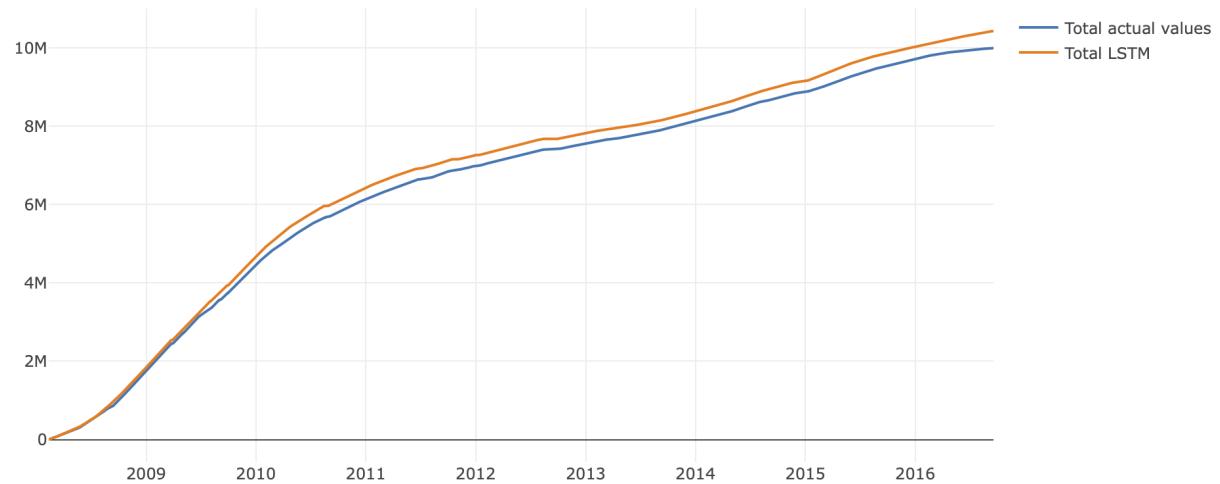


Figure E.12. Total cumulative oil production

BIOGRAPHY

Full name: Luong Khanh Loc

Date of birth: 25/10/1994

Place of birth: Quang Ngai

Contact Address: Phuoc Thinh Hamlet, Duc Thanh Ward, Mo Duc District, Quang Ngai Province.

CONFIRMATION OF THE QUALIFIED THESIS FOR STORAGE IN LIBRARY

RECTOR

Dr. Phan Minh Quoc Binh

**HEAD OF OFFICE OF
ACADEMIC AFFAIRS**

Dr. Le Quoc Phong

SUPERVISOR

Dr. Nguyen Van Hung