Sudan University of Science and Technology

College of Petroleum and Mining Engineering

Department of Petroleum Engineering

# Predicting liquid loading in gas wells using machine learning

## Case study (FN 21, FN4-7)

## التنبؤ بتراكم السوائل باستخدام تقنية الذكاء الاصطناعي

## دراسة الحالة (الفولة شمال 21 والفولة شمال 7-4)

A Thesis Submitted to the College of Petroleum and Mining Engineering - Sudan University of Science and Technology, in Partial Fulfilment of bachelor's degree (BSc) in Petroleum Engineering

**Prepared by:**

Abdullah Abdalmonem Abdullah.
Ahmed Mohamed Khair Eisa.
Sami Faisal Jaafar Rezig.
Osman Kamaludeen Osman.
Mujtaba Haider Abdulrahman.

**Supervised by:**
Dr. Muhanned Mahjoub Khairy

**February** 2022

# الإستهلال

قال تعالى:

بسم الله الرحمن الرحيم

{ قَالَ يَٰقَوْمِ أَرَءَيْتُمْ إِن كُنتُ عَلَىٰ بَيِّنَةٍۢ مِّن رَّبِّى وَرَزَقَنِى مِنْهُ رِزْقًا حَسَنًا ۚ وَمَآ أُرِيدُ أَنْ أُخَالِفَكُمْ إِلَىٰ مَآ أَنْهَىٰكُمْ عَنْهُ ۚ
إِنْ أُرِيدُ إِلَّا ٱلْإِصْلَٰحَ مَا ٱسْتَطَعْتُ ۚ وَمَا تَوْفِيقِىٓ إِلَّا بِٱللَّهِ ۚ عَلَيْهِ تَوَكَّلْتُ وَإِلَيْهِ أُنِيبُ }

**سورة هود الآية (88)**

# Dedication

We would like to donate this unpretentious effort to

**Our Parents;** who have endless presence and for the never ending love and encouragement

**Our brothers and sisters;** who sustained us in our life and still

**Our teachers;** who lighted candle in our ways and provided us with light of knowledge

**For her;** for being there when no one were

**Finally; our best friends……..**

# Acknowledgement

Everything that has a beginning must equally have an end. Thanks, of Allah for the gift of life in good health and abundant grace throughout our stay in this great citadel. It’s indeed a privilege and honor to pass through this college.
We acknowledge the effort of every lecturer that has impacted knowledge into us, without your contributions we would not be who we are today.

# Abstract

The time consumed between observing the production decline and identifying the liquid loading problem is a big challenge, facing the production engineer. The objective of this project is to predict the liquid loading condition using a machine learning approach. This work included visualizing, preprocessing and modeling the data by k-nearest neighbors regression algorithm. For this study gaseous wells were selected. Production and completion history for each well were collected. First study was performed on (synthetic) data where 70 percent of the information were used for the training purpose, 15 percent for calibration and 15 percent for validation of the model. The model successfully anticipated the liquid loading status with an accuracy of 93% and 93% of the data trained. A local data obtained from the well (FN 21) had experienced an attempt to be modeled by the same way, but due to lack in completion and production data, the attempt has failed. Another local data obtained from the well (FN4-7) with complete completion and production data was modeled. The model successfully predicted liquid loading status with an accuracy of 92 % and 100% of the data trained. This new smart model developed for local data shows a great promise that this approach can be applied in other areas where a limited history of production and liquid are available

# التجريد

الزمن المستهلك بين ملاحظة الانخفاض في كمية الإنتاج وتحديد ان السبب في ذلك هو مشكلة (liquid loading) يعتبر من أكبر التحديات التي تواجه مهندس الإنتاج. الهدف من هذا المشروع هو توقع الظروف التي تحدث فيها مشكلة (liquid loading) عن طريق استخدام نموذج تم بناؤه عن طريق لغة البايثون. في هذا العمل تم عرض ومعالجة البيانات ثم تم تصميم النموذج باستخدام k-nearest neighbors) regression algorithm). لهذه الدراسة تم اختيار ابار انتاج غازية وتم جمع بيانات الاكمال والإنتاج لكل بئر. تم اجراء دراسة أولى على بيانات تخليقية (synthetic) حيث تم استخدام 70 % من البيانات لغرض تدريب النموذج و15% للمعايرة و15% للمطابقة. نجح النموذج في توقع حالة ال (liquid loading) بدقة قدرها 93% وتم تدريب 93% من البيانات. تم استخدام بيانات محلية من البئر (FN 21) وكانت هنالك محاولة لتنفيذ هذا النموذج عليها ولكن بسبب النقص في بيانات الإنتاج والاكمال باءت المحاولة بالفشل. تم استخدام بيانات محلية أخرى من البئر (FN4-7) وكانت بيانات الاكمال والإنتاج مكتملة هذه المرة. نجح النموذج هذه المرة في توقع حالة ال (liquid loading) بدقة مقدارها 92% وتم تدريب 100% من البيانات.

# List of Contents

# Introduction

## 1.1 INTROIDUCTION:

One of the big challenges associated with horizontal and highly inclined wellbores is related to liquid loading in wellbores, which is defined as the phenomenon when a drop-in gas rate hinders a wells capability to lift liquid up to the surface. Onset of liquid loading is the condition at which critical gas velocity is reached and liquid starts to accumulate in the wellbore.

Some of the problems encountered due to liquid loading involve pipeline internal corrosion, disturbance and/or damage to the downstream facilities, steep production declines due to an increase of back pressure pipeline fatigue production instability.

Accurate prediction of the onset of liquid loading requires a good prediction of pressure gradient and liquid holdup for segregated flow. Segregated flow is a commonly encountered flow pattern during oil and gas operations.

A generalized solution to predict friction factors is proposed in this work by combining machine learning with the physics-based two-fluid model to accurately determine pressure gradient and liquid holdup for a wide range of flow conditions and fluid properties Once the pressure gradient and liquid holdup are determined the model predicts critical gas velocity.

### 1.1.1 Characteristics of gases:

1. They are easy to compress.
2. They expand to fill their containers.
3. They occupy far more space than other liquid.

### 1.1.2 Natural gas is categorized by composition to:

#### 1.1.2.1 Dry Gas:

Natural gas that occurs in the absence of condensate or liquid hydrocarbons, or gas that had condensable hydrocarbons removed, is called dry gas. It is primarily methane with some intermediates. The hydrocarbon mixture is solely gas in the reservoir and there is no liquid (condensate surface liquid) formed either in the reservoir or at surface. The pressure path line does not enter into the phase envelope in the phase diagram, thus there is only dry gas in the reservoir. Note the surface separator conditions also fall outside the phase envelope (in contrast to wet gas); hence, no liquid is formed at the surface separator.

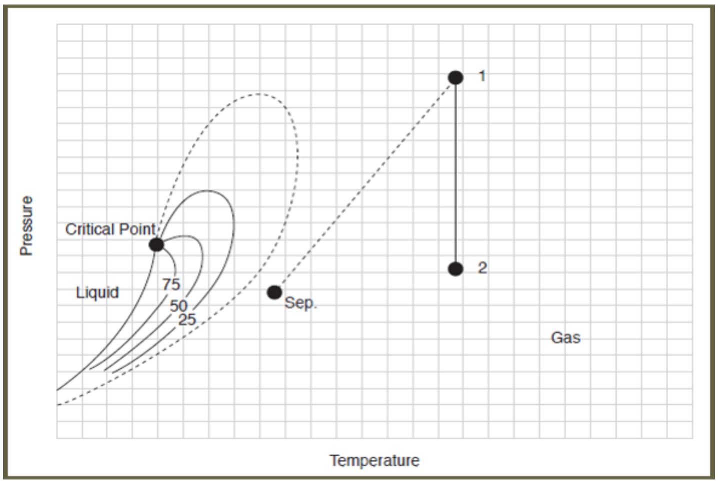


FIGURE 1.1 DRY GAS

#### 1.1.2.2 Wet Gas:

Natural gas that contains significant heavy hydrocarbons such as propane, butane and other liquid hydrocarbons is known as wet gas or rich gas. The general rule of thumb is if the gas contains less methane (typically less than 85% methane) and more ethane, and other more complex hydrocarbons, it is labeled as wet gas.

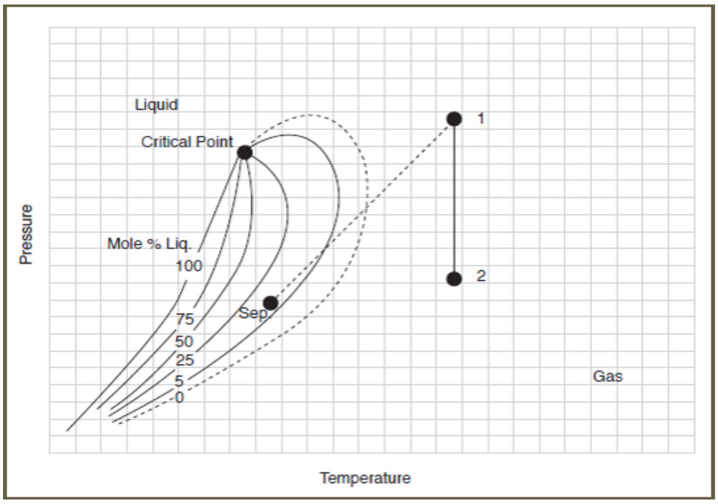


FIGURE 1.2 WET GAS

## 1.1.2.3 Condensate gas:

Condensate gas is very similar to volatile oils in terms of the color (green, orange, brown, even clear) and gravity (40° to 60° API) of the produced oil. However, the reservoir temperature of a condensate gas reservoir is greater than the critical temperature of the fluid, and so where a volatile oil is a liquid at original reservoir pressure and temperature, a condensate gas is a gas.

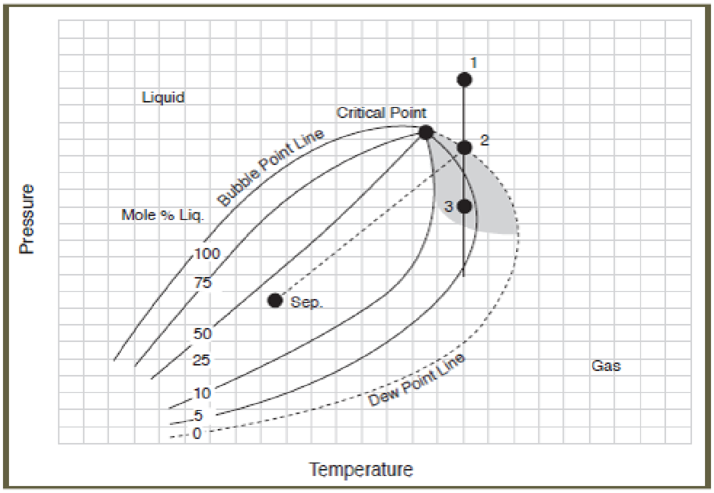


FIGURE 1.3 CONDENSATE GAS

## 1.2 Problem Statement:

Inability of the well to lift the fluid associated with produced gas to the surface, as observed when the flow pattern progresses from mist flow to bubble flow.

Here in this project we will use the machine learning to predict the critical gas velocity, under which liquid accumulation will occur.

## 1.3 Objectives:

### 1.3.1 Primary objective:

The research aims to analyze the well production data to predict the status of the well either loading or unloading.

### 1.3.2 Sub-objective:

1 Preprocessing acquired well data.

2 Modeling the data using python programing language.

3 Predicting liquid loading status.

## 1.4 Methodology:

A supervised learning algorithm is used to create a model that aims to predicting liquid loading status in gaseous wells.

The model consists of three stages:

1. Visualizing the data.
2. Preprocessing the data.
3. Modeling the data.

## 1.5 Project Layout:

This project report has been divided into five chapters: -

### 1.5.1 Chapter one:

Represents a brief introduction related to our project.

### 1.5.2 Chapter two:

Explains the literature review with latest publications related to the most problem caused reduction in gas well performance.

### 1.5.3 Chapter three:

Customized our methods and program used to mention the problems which called by methodology.

### 1.5.4 Chapter four:

We analyze the collected data and make prediction calculations of optimum production of gas well by using OLGA-software, then solve the problem.

### 1.5.5 The last chapter:

We put conclusion, our future Recommendation and References helped us to understand these problems.

# Literature Review

This chapter is cornering to learn the liquid loading problem, the different method used to predict it.

## 2.1 Theoretical Background:

### 2.1.1 Liquid Loading in Gas Wells:

#### 2.1.1.1Flow Patterns in a Gas Well:

The flow pattern in a vertical production conduit of a gas well is usually illustrated by four basic flow patterns or flow regimes as shown in Fig. 2.1. The flow regimes are largely classified with bubble flow, slug flow, slug-annular transition flow and annular mist flow, which are determined by the velocity of the gas and liquid phases and the relative amounts of gas and liquid at any given point in the flow stream.

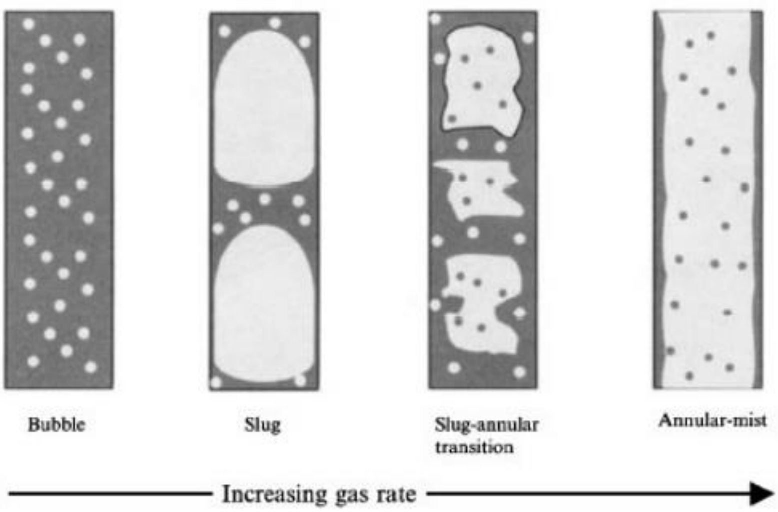


**FIGURE 2. 1** FLOW REGIME IN VERTICAL MULTIPHASE FLOW

If the flow pattern is an annular-mist type, the well still may have a relatively low gravity pressure drop. However, as the gas velocity begins to drop, the well flow can become a slug type and then bubble flow. In these cases, a much larger fraction of the tubing volume is filled with liquid. A gas well may go through any or all of these flow regimes during its lifetime. The general progression of a typical gas well from initial production to its end of life is shown in Fig. 2.2.

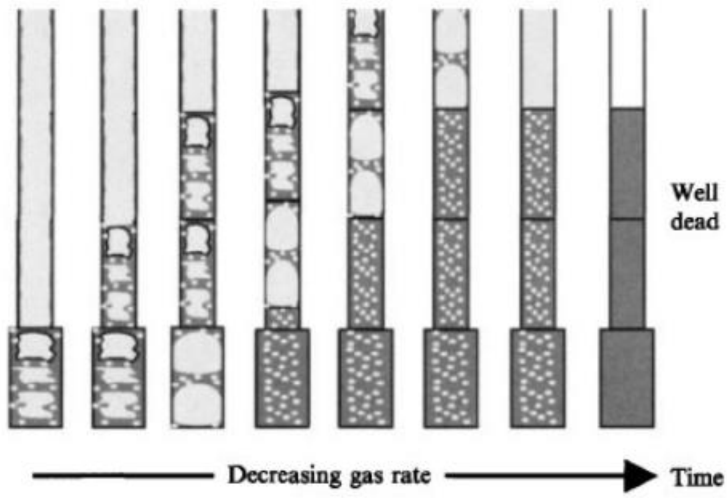


FIGURE 2. 2 PROGRESSION OF A TYPICAL GAS WELL

Initially, the well may show the mist flow regime that brings a high gas rate and then transit into slug-annular transition, slug, and bubble flow with time. Liquid production may also increase as the gas production declines. Flow at the surface will remain in mist flow until the conditions change sufficiently at the surface so that the flow exhibits transition flow. Flow downhole may show bubble or slug flow even though the flow regime at the surface looks like a mist flow.

### 2.1.2 Occurrence of Liquid Loading:

Gas and liquid are both produced to surface if the gas velocity is high enough to lift or carry liquid. The problem happens because the velocity of the gas in the tubing drops with time, and the velocity of the liquids decline even faster as the production goes on. As a result, the liquid begins to accumulate in the bottom of the well and liquid slugs are formed in the conduit, which increase the percentage of liquids in the conduits while the well is flowing. The bottomhole pressure increases and gas production decreases until gas flow stops. In other words, the liquid loading process occurs when the gas velocity within the well drops below a certain critical gas velocity. The gas is then unable to lift the water coproduced with the gas (either condensed or formation water) to surface. The water will fall back and accumulate downhole. A hydrostatic column is formed that imposes a back pressure on the reservoir and hence reduces gas production. The process eventually results in intermittent gas production and well die-out. Several sources may be suspected as the source of liquid causing the problem. It is reasonably said that the liquid sources may be from water coning, aquifer water, water produced from another zone, free formation water, and hydrocarbon condensate.

### 2.1.3 Recognizing of Symptoms of Liquid Loading:

The occurrence of liquid loading in a gas well can be recognized by several symptoms. If it is found out early and then the appropriate action is taken at a proper time, the losses in gas production can be minimized. The symptoms indicating liquid loading summarized by James F. Lea (2004) are like following:

1. Sharp reduction of flow rate

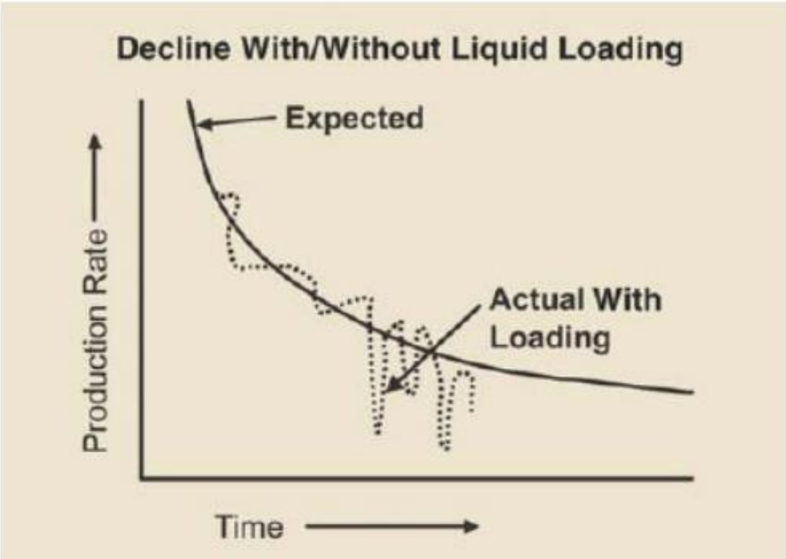


FIGURE 2. 3 DECLINE CURVE SHOWING ONSET OF LIQUID LOADING

2. Onset of liquid slugs at the surface of the well
3. Increasing difference between the tubing and casing flowing pressure (i.e. pcf-ptf) with time, measurable without packers present.
4. Sharp changes in gradient on a flowing pressure survey.

### 2.1.4 Remedial Lifting Options to Reduce Liquid Loading Problem:

Many types of technique of remedial lifting have been developed so far. Most of the techniques focus on increasing gas velocity and artificially water lifting to reduce liquid loading problems. The following table (Veeken, 2003) shows the remedial measures depending on the purpose of use. These methods may be used singly or in combination of two or more.

**TABLE 2. 1 REMEDIAL MEASURES TO REDUCE LIQUID LOADING**

| Classification | Techniques |
|---|---|
| Increase gas velocity | - Intermittent production - Gas lift<br>- Stimulation - Venting<br>- Compression - Velocity string |
| Reduce critical velocity | - Compression<br>- Velocity string<br>- Mechanical liner solutions (stinger)<br>- Batch soap sticks / surfactant<br>- Continuous surfactant injection (capillary strings)<br>- Bubble breakers (restriction) |
| Artificially lift water | - Plunger<br>- Chamber (plunger plus lift gas)<br>- Downhole pump (rod, PCP, ESP)<br>- Swabbing |
| Remove water | - Downhole separation & Injection (intermittent production)<br>- Heated tubing |

## 2.2 Liquid Loading:

Liquid loading in gas wells is the accumulation of liquids (water, condensate, or both) in the wellbore due to pressure decline. It occurs in vertical or deviated wells during production from natural gas reservoirs as a result of condensation and coalescence of liquids from gas streams. This is common in both offshore and onshore production systems and results in the simultaneous flow of gas, liquid hydrocarbons and water.

Discussed below are the basics of Turner et al. model, Coleman et al. model and LI's Model, which have been applied in this project.

**Turner, Hubbard, and Dukler (1969),** after studying the earlier observations, proposed two physical models for the removal of gas well liquids. The models are based on: This model assumes that annular liquid film should have to be continuously moved upward along the wells to achieve liquid unloading. The model calculates the minimum flow rate requirement to move the film upward. Turner concluded that the predictions of the film model do not provide a clear definition between the adequate and inadequate rates. Liquid droplets entrained in the high velocity gas core. The minimum gas flow rate that will lift the drops out of the well to the surface. According to the study, a free-falling particle reaches a terminal velocity which is the maximum velocity it can attain against gravity. Therefore, that terminal velocity, or in other terms the critical gas velocity which is determined by the flow conditions necessary to remove the liquids on a continual basis, is based on drag & gravitational forces on the droplet. Applying this concept of liquid droplets in a flowing core of natural gas column, the critical velocity, Vc of the drop is, which assumes a fixed droplet size, shape and drag coefficient and includes the 20% adjustment suggested by Turner, based on field results matching.

$$V_c = \frac{1.912\sigma^{\frac{1}{4}}(\rho l-\rho g)^{\frac{1}{4}}}{\rho^{\frac{1}{2}}} \text{ ..........................................} (2{:}1)$$

Where:

$V_c$= critical velocity, ft/sec

σ = surface tension, dynes/cm

ρl = liquid density, lbm/ft3

ρg = gas density, lbm/ft3

Inserting typical values of:

Surface Tension = 20 and 60 dyne/cm for condensate and water, respectively.

Density = 45 and 67 lbm/ft3 for condensate and water, respectively. Gas Z factor = 0.9

$$\rho_g = \frac{PM_aY_g}{ZRT} \text{..........................................} (2{:}2)$$

By substituting the above typical values, a simplified pressure equation was developed:

$$\rho_{g=0.0031*P} \text{..........................................} (2{:}3)$$

The critical velocity can be converted to the critical rate at standard conditions for a given pressure, P, and tubular dimensions using the following equation:

$$Q_c = \frac{V_c A}{B_g} \qquad (2:4)$$

Where $B_g$ is the gas formation volume factor defined as follows:

$$B_g = \frac{ZTP_{sc}}{PT_{sc}} \qquad (2:5)$$

Substituting for standard conditions, pressure $P_{sc}$ = 14.65 psi and temperature $T_{sc}$ = 520 or, Eq (2:4) can be written as

$$Q_c = \frac{3.06 P V_c A}{(T+460)Z} \qquad (2:6)$$

Where:

$$A = \frac{\pi dt^2}{4*(12)^2} \qquad (2:7)$$

T = surface temperature.°F

P = pressure at the evaluation point, psi

A = tubing cross-sectional area, $ft^2$

dt = tubing ID, inches

**Coleman et al. (1991),** using the Turner model but validating with field data of lower reservoir and wellhead flowing pressures all below approximately 500 Pisa, Coleman et al. discovered that a better prediction could be achieved without a 20% upward adjustment to fit field data with the following expressions:

$$\vartheta_{crit} = 1.593\left(\frac{\sigma(\rho_L - \rho_g)}{{\rho_g}^2}\right)^{\frac{1}{4}} \qquad (2:8)$$

$$q_{crit} = \frac{3060 P V_{crit} A}{Tz} \qquad (2:9)$$

**Nosier et al (1997),** focused their studies on the impact of flow regimes in addition, changes in flow conditions on gas well loading.

They followed the path of turner droplet model but they made a difference from turner model by considering the impact of flow regimes on the drag coefficient. On comparing nosier observed that Turner model values were not matching with the real data for highly turbulent flow regime. Dealing with this deviation nosier found out the reason to be the change in value of Cd for this regime from .44 to 0.2. As a result, they proposed two new equations regarding the critical velocity:

For transition regime

$$\vartheta_{crit} = \frac{14.6\sigma^{0.35}(\rho_p-\rho)^{0.21}}{\mu^{0.134}\rho^{0.426}} \ldots\ldots (2:10)$$

For highly turbulent regime

$$\vartheta_{crit} = \frac{21.3\sigma^{0.25}(\rho_p-\rho)^{0.25}}{\rho^{0.5}} \ldots\ldots (2:11)$$

**Li et al. (2001),** Li, Li, Sun in there research posited that turner and Coleman's models did not consider deformation of the free falling liquid droplet in a gas medium.

They contended that as a liquid droplet is entrained in a high- velocity gas stream, a pressure difference exists between the fore and aft portions of the droplet

(Figure2:1) shows the droplet's shape changes from spherical to flat in a high velocity. Compared with spherical droplets, flat ones need low gas velocity and flow rate due to having more efficient area. For the Reynolds number range $10^4$<Re<2× $10^5$, drag coefficient ($C_D$) for turner's model is 0.44, but for flat shaped one is 1.0, which means smaller critical velocity than spherical droplet.

$$\vartheta_{crit}=0.7241\sqrt[4]{\frac{(\rho_l-\rho_g)^{\sigma}}{{\rho_g}^{0.5}}} \ldots\ldots (2:12)$$

$$Q_{crit}=3060\frac{Apv_{crit}}{ZT} \ldots\ldots (2:13)$$

## 2.3 Literature Review:

### 2.3.1 Previous experimental and modeling studies on onset of liquid loading:

This section provides a comprehensive background of relevant experimental and modeling studies focused on the estimation of critical gas velocity. The scope is restricted to upward inclined pipes. The modeling work has been further divided into liquid droplet models and film reversal-based models, which are the two primary mechanisms through which liquid loading is determined.

#### 2.3.1.1 Previous experimental studies on onset of liquid loading:

Multiple experimental studies can be found in the literature which have made a successful attempt to determine the onset of liquid loading. However, most of these experiments have been carried out in pipes with small size diameters, such as Wang et al. 2016, Fan et al. 2018, Brito 2015, etc. A summary of experimental studies for two-phase pipe flow along with the defining parameters is available in Table 2.2.

TABLE 2. 2 SUMMARY OF EXPERIMENTAL STUDIES FOR TWO-PHASE PIPE FLOW ALONG WITH THE DEFINING PARAMETERS

| Authors | $d$ (m) | $\theta$ (°) | $\rho_L$ (kg/m$^3$) | $\rho_G$ (kg/m$^3$) | $\mu_L$ (Pa s) | $v_{SL}$ (m/s) | Fluids |
|---|---|---|---|---|---|---|---|
| Alsaadi et al. (2015) | 0.0762 | 2 ~ 30 | 1000 | 1.18 | 0.001 | 0.01-0.1 | Air/Water |
| Brito (2015) | 0.0508 | 1 | 1000 | 1.18 | 0.001 | 0.0015-0.023 | Air/Water |
| Nair (2017) | 0.1524 | 1 | 1000 | 1.18 | 0.001 | 0.003-0.03 | Air/Water |
| Fan et al. (2018) | 0.0762 | 2 ~ 20 | 1000 | 1.18 | 0.001 | 0.001-0.01 | Air/Water |
| Guner et al. (2015) | 0.0762 | 45 ~ 90 | 1000 | 1.18 | 0.001 | 0.01-0.1 | Air/Water |
| Langsholt and Holm (2007) | 0.1 | 0.5 ~ 5 | 812, 821, 998 | 22.6 46.9 | 0.0018 0.001 | 0.001 | $SF_6$ / Exxsol D80, $SF_6$/Water |
| Rodrigues (2018) | 0.154 | 2 | 760 | 1.165 | 0.0013 | 0.01-0.05 | Nitrogen/Isopar L |
| Espedal (1998) | 0.1 | 1, 2 | 829 | 50 | 0.0018 | 0.0005 | $SF_6$ / Exxsol D80, |
| Current Study | 0.1524 | 0.5 ~ 90 | 997 | 0.97 | 0.001 | 0.0007-0.02 | Air/Water |

According to Alsaadi et al. (2015) and Fan et al. (2018), an increase in critical gas velocity was observed when inclination angles were increased from 2° to 30°. They concluded that liquid flow rate effects on critical gas velocity are more significant for higher inclination angles.

## 2.3.1.2 M. F. Riza, A. R. Hasan and C. S. Kabir, August 18 2016 "A Pragmatic Approach To Understanding Liquid Loading in Gas Wells":

This study found that the flow condition at, or very near, the well bottom controls the onset of liquid loading. By use of the data sets of Turner et al. (1969), Coleman et al. (1991a, 1991b), and Veeken et al. (2010), we showed that prediction quality appears to improve with the entire wellbore-modeling approach.

Forward modeling suggested that the tubing inside diameter and the well productivity index are the most important variables in determining the critical liquid-loading rate and the onset of liquid loading.

### 2.3.1.2 Ankit Malik, Ravi Prakash, Mukesh Kumar and Miten Barot, October 17 2017, " Predicting Start-Up Liquid Loading in a Mature Oil & Gas Field: A Case Study":

The key outcomes of this case study are presented below:

1. The field observation of loading of high GLR oil wells could not be explained with steady state analysis. This approach was found to be optimistic in predicting onset of liquid loading.
2. Dynamic model or production data analysis data derived steady-state well productivity and phase ratios are more relevant for normal flowing periods. For the duration of the unloading process, these parameters are highly uncertain and difficult to capture in modeling. An element of risking has been captured to predict the most likely time of start-up loading using a risking criterion, which was based on individual well's flow parameters.
3. The workflow discussed in this paper helped to justify requirement of lift gas pipeline on a platform in the subject field. This lift gas pipeline has helped to revive wells after shutdowns and without it all the current production from this platform would have been risked.
4. It is recommended to evaluate liquid loading from predicted parameters at field development planning stage using the proposed workflow to secure future well production and save on additional costs of retrofitting gas lift infrastructure.

### 2.3.1.3 Hewei Tang and Zhi Chai, 2018, "What Happens after the Onset of Liquid Loading? an Insight from Coupled Well-Reservoir Simulation":

In this study, we introduced a fully implicitly coupled wellbore reservoir model to examine the production performance and pressure dynamics of horizontal wells after the onset of liquid loading. The model incorporated a modified drift-flux model that is able to predict the onset of liquid loading and the subsequent unstable well behaviors. We applied the model to analyze the gas and water production scenarios of an open-hole horizontal well, a horizontal well with uniform stimulation, and a horizontal well with multi- stage hydraulic fractures. The following conclusions can be obtained:

1. There exists a gas-water coproduction period and a zero liquid production period after the onset of liquid loading for most production scenarios being investigated. The lengths of both production periods increase as reservoir permeability decreases from 5md to 0.3md.
2. For reservoir permeability equals to 0.1md, the horizontal gas well experiences natural cyclical production after the onset of liquid loading. It is because of the periodic buildup and draw down of reservoir pressure. The production phenomenon is consistent with reported field observations.
3. The natural cyclical production is introduced by the high initial pressure difference between the wellbore and the reservoir. Both uniform stimulation and hydraulic fracturing mitigate or eliminate this production phenomenon.

### 2.3.1.4 Ayush Rastogi and Yilin Fan, 2019, Experimental Investigation and Modeling of Onset of Liquid Accumulation in Large- Diameter Deviated Gas Wells:

An experimental facility with a 6-in. pipe diameter having a test section of 32-ft. is constructed to conduct experiments for two-phase flow with an inclination angle range from 0° to 90°. The current experimental data and the comparison with other previous data show that the critical gas velocity increases with increasing pipe diameter and gas density. It increases first with increasing pipe inclination angle until 40° – 50° approximately, and then decreases due to the decrease of the maximum liquid film thickness. A new model was developed to predict the maximum liquid film thickness at the pipe bottom as a function of the inclination angle. In general, the new model gives the best prediction for the critical gas velocity when compared with others.

# Methodology

Machine learning is a process through which computer will learn from data to find a possible pattern in the data set. This process encompasses three main components; Learning algorithm, Data, and Pattern in the data. If these three components are present, a successful learning process can be achieved based on the capability of the learning algorithm. There are two major types of Machine Learning: supervise learning and unsupervised learning. In supervised learning, both input and output are available, and the learning algorithm tries to find the relationship between them. One of the supervised learning algorithm that will be used in this article is "Artificial Neural Network" (ANN).

In unsupervised learning, there is no information about the output. The learning algorithm tries to find the pattern inside the input data alone. One of unsupervised learning algorithm that will be used in this article is "K-mean Clustering".

## 3.1 Why python?

Python has become the lingua franca for many data science applications. It combines the power of general-purpose programming languages with the ease of use of domain-specific scripting languages like MATLAB or R. Python has libraries for data loading, visualization, statistics, natural language processing, image processing, and more.

One of the main advantages of using Python is the ability to interact directly with the code, using a terminal or other tools like the Jupyter Notebook, which we'll look at shortly. Machine learning and data analysis are fundamentally iterative processes, in which the data drives the analysis. It is essential for these processes to have tools that allow quick iteration and easy interaction.

## 3.2 scikit-learn:

scikit-learn is an open source project, meaning that it is free to use and distribute, and anyone can easily obtain the source code to see what is going on behind the scenes.

The scikit-learn project is constantly being developed and improved, and it has a very active user community. It contains a number of state-of-the-art machine learning algorithms, as well as comprehensive documentation about each algorithm. scikit-learn is a very popular tool, and the most prominent Python library for machine learning. It is widely used in industry and academia, and a wealth of tutorials and code snippets are available online. scikit-learn works well with a number of other scientific Python tools.

## 3.3 production data preprocessing:

Data preprocessing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model.

A real-world data generally contains noises, missing values, and maybe in an unusable format which cannot be directly used for machine learning models. Data preprocessing is required tasks for cleaning the data and making it suitable for a machine learning model which also increases the accuracy and efficiency of a machine learning model.

```
In [2]: data = pd.read_csv('WellLoadingData.csv')
```

```
In [3]: data.head(10)
```

Out[3]:

| | AvgFlowRate | FlowRate | AvgStaticPressure | StaticPressure | AvgCasingPressure | CasingPressure | AvgTubingPressure | TubingPressure | Loading |
|---|---|---|---|---|---|---|---|---|---|
| **0** | 711 | 706.5961 | 522 | 525.9678 | 776 | 733.9601 | 609 | 607.4733 | False |
| **1** | 711 | 706.5961 | 522 | 523.8666 | 776 | 734.1985 | 609 | 607.5895 | False |
| **2** | 711 | 706.5961 | 522 | 527.5427 | 776 | 734.3147 | 609 | 609.6815 | False |
| **3** | 711 | 690.1027 | 522 | 527.6544 | 776 | 734.4816 | 609 | 607.4539 | False |
| **4** | 711 | 690.1027 | 522 | 525.5143 | 776 | 734.0510 | 609 | 607.3660 | False |
| **5** | 711 | 690.1027 | 522 | 523.1964 | 776 | 734.7111 | 609 | 604.5931 | False |
| **6** | 711 | 701.5277 | 522 | 525.8849 | 776 | 734.2759 | 609 | 608.7160 | False |
| **7** | 711 | 690.1027 | 522 | 526.5861 | 776 | 733.3358 | 609 | 601.0349 | False |
| **8** | 711 | 708.5102 | 522 | 525.3587 | 776 | 733.8141 | 609 | 602.5637 | False |
| **9** | 711 | 690.1027 | 522 | 526.0814 | 776 | 737.7075 | 609 | 605.1563 | False |

**FIGURE 3. 1 DATA VISUALIZATION**

## 3.4 Data visualization:

Data visualization is an integral part of any data science project. Understanding insights using excel spreadsheets or files becomes more difficult when the size of the dataset increases. It’s certainly not fun to scroll up/down to do an analysis.

It involves the creation and study of the visual representation of data. To communicate information clearly and efficiently, data visualization uses statistical graphics, plots, information graphics and other tools. Numerical data may be encoded using dots, lines, or bars, to visually communicate a quantitative message.

In data visualization, we use different graphs and plots to visualize complex data to ease the discovery of data patterns.

### 3.4.1 Importance of data visualization:

Data visualization also helps identify areas that need attention, e.g outliers, which can later impact our machine learning model. It also helps us understand the factors that have more impacts on your results: for example, in house price predictions, the house price will be impacted more by the size of the house than the house style.

## 3.5 model selection:

### 3.5.1 k-neighbors regression:

There is also a regression variant of the k-nearest neighbors algorithm. Again, let’s start by using the single nearest neighbor, this time using the wave dataset. We’ve added three test data points as green stars on the x-axis. The prediction using a single neighbor is just the target value of the nearest neighbor. These are shown as blue stars in Figure 3:2 :

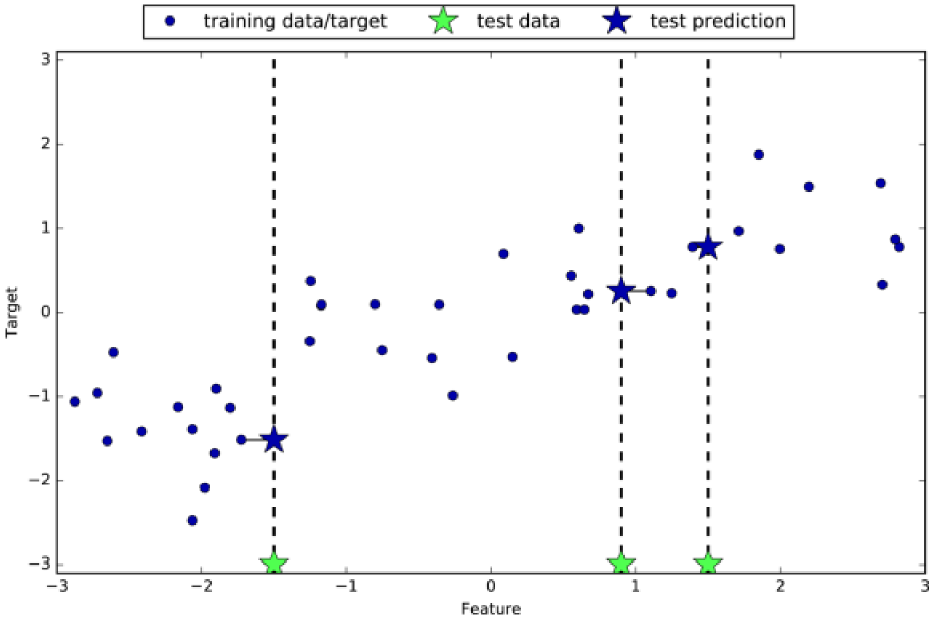


**FIGURE 3. 2 PREDICTIONS MADE BY ONE-NEAREST-NEIGHBOR REGRESSION ON THE WAVE DATASET**

we can use more than the single closest neighbor for regression. When using multiple nearest neighbors, the prediction is the average, or mean, of the relevant neighbors (Figure 3:3)

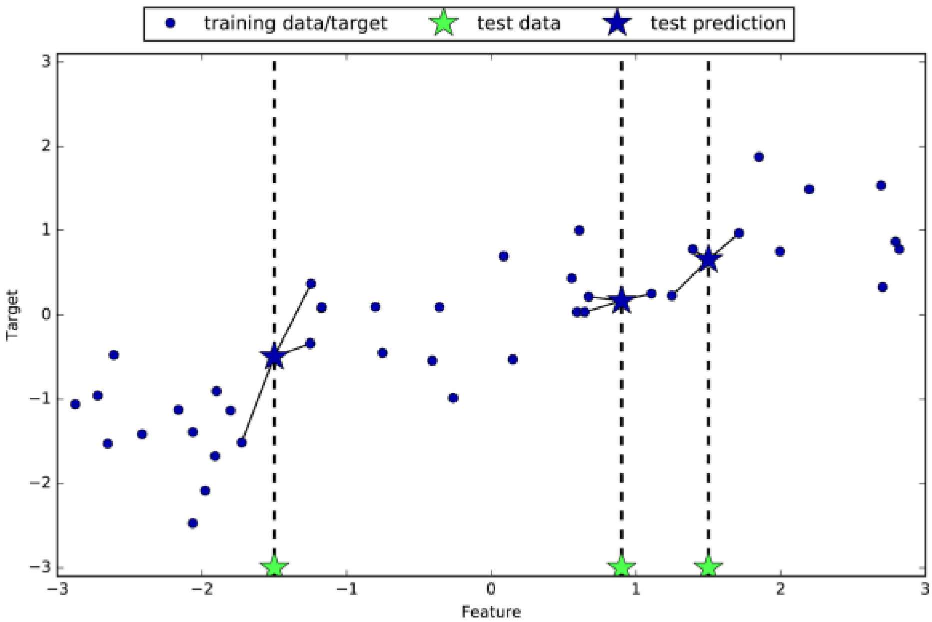


**FIGURE 3. 3 PREDICTIONS MADE BY THREE-NEAREST-NEIGHBORS REGRESSION ON THE WAVE DATASET**

The k-nearest neighbors algorithm for regression is implemented in the KNeighbors Regressor class in scikit-learn. It's used similarly to KNeighborsClassifier.

```
from sklearn.neighbors import KNeighborsRegressor

X, y = mglearn.datasets.make_wave(n_samples=40)

# split the wave dataset into a training and a test set
X_train, X_test, y_train, y_test = train_test_split(X, y, random_state=0)

# instantiate the model and set the number of neighbors to consider to 3
reg = KNeighborsRegressor(n_neighbors=3)
# fit the model using the training data and training targets
reg.fit(X_train, y_train)
```

we can make predictions on the test set:

```
print("Test set predictions:\n{}".format(reg.predict(X_test)))
```

```
Test set predictions:
[-0.054  0.357  1.137 -1.894 -1.139 -1.631  0.357  0.912 -0.447 -1.139]
```

can also evaluate the model using the score method, which for regressors returns the R 2 score. The R 2 score, also known as the coefficient of determination, is a measure of goodness of a prediction for a regression model, and yields a score between 0 and 1. A value of 1 corresponds to a perfect prediction, and a value of 0 corresponds to a constant model that just predicts the mean of the training set responses, y train:

```
print("Test set R^2: {:.2f}".format(reg.score(X_test, y_test)))
```

```
Test set R^2: 0.83
```

Here, the score is 0.83, which indicates a relatively good model fit.

### 3.5.2 Analyzing KNeighborsRegressor:

For our one-dimensional dataset, we can see what the predictions look like for all possible feature values (Figure 2-10). To do this, we create a test dataset consisting of many points on the line:

```
fig, axes = plt.subplots(1, 3, figsize=(15, 4))
# create 1,000 data points, evenly spaced between -3 and 3
line = np.linspace(-3, 3, 1000).reshape(-1, 1)
for n_neighbors, ax in zip([1, 3, 9], axes):
    # make predictions using 1, 3, or 9 neighbors
    reg = KNeighborsRegressor(n_neighbors=n_neighbors)
    reg.fit(X_train, y_train)
    ax.plot(line, reg.predict(line))
    ax.plot(X_train, y_train, '^', c=mglearn.cm2(0), markersize=8)
    ax.plot(X_test, y_test, 'v', c=mglearn.cm2(1), markersize=8)

    ax.set_title(
        "{} neighbor(s)\n train score: {:.2f} test score: {:.2f}".format(
            n_neighbors, reg.score(X_train, y_train),
            reg.score(X_test, y_test)))
    ax.set_xlabel("Feature")
    ax.set_ylabel("Target")
axes[0].legend(["Model predictions", "Training data/target",
                "Test data/target"], loc="best")
```

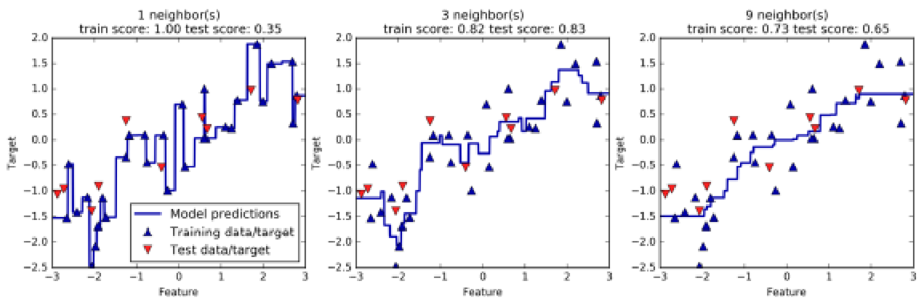


**FIGURE 3. 4 COMPARING PREDICTIONS MADE BY NEAREST NEIGHBORS REGRESSION FOR DIFFERENT**

As we can see from the plot, using only a single neighbor, each point in the training set has an obvious influence on the predictions, and the predicted values go through all of the data points. This leads to a very unsteady prediction. Considering more neighbors leads to smoother predictions, but these do not fit the training data as well.

### 3.5.3 Strengths, weaknesses, and parameters:

In principle, there are two important parameters to the KNeighbors classifier: the number of neighbors and how you measure distance between data points. In practice, using a small number of neighbors like three or five often works well, but you should certainly adjust this parameter. Choosing the right distance measure is somewhat beyond the scope of this book. By default, Euclidean distance is used, which works well in many settings.

One of the strengths of k-NN is that the model is very easy to understand, and often gives reasonable performance without a lot of adjustments. Using this algorithm is a good baseline method to try before considering more advanced techniques. Building the nearest neighbors model is usually very fast, but when your training set is very large (either in number of features or in number of samples) prediction can be slow. When using the k-NN algorithm, it’s important to preprocess your data. This approach often does not perform well on datasets with many features (hundreds or more), and it does particularly badly with datasets where most features are 0 most of the time (so-called sparse datasets).

# Results and Discussion

In this chapter, we will extract the results from modeling data acquired from tow wells using python programming language, to predict the liquid loading status in the wells.

## 4.1 case study for synthetic data:

### 4.1.1 Data visualization:

The figures 4:1 and 4:2 shows the data of the Latin well, which well be preprocessed and then modeled.

The figure below refers to unloading status of the well (0)

Out[5]:

| | FlowRate | StaticPressure | CasingPressure | TubingPressure | Loading |
|---|---|---|---|---|---|
| **0** | 706.5961 | 525.9678 | 733.9601 | 607.4733 | 0 |
| **1** | 706.5961 | 523.8666 | 734.1985 | 607.5895 | 0 |
| **2** | 706.5961 | 527.5427 | 734.3147 | 609.6815 | 0 |
| **3** | 690.1027 | 527.6544 | 734.4816 | 607.4539 | 0 |
| **4** | 690.1027 | 525.5143 | 734.0510 | 607.3660 | 0 |
| **...** | ... | ... | ... | ... | ... |
| **417** | 674.6470 | 528.1722 | 745.8297 | 618.9897 | 0 |
| **418** | 674.6470 | 526.3243 | 746.2081 | 621.3455 | 0 |
| **419** | 674.6470 | 528.3762 | 746.1083 | 609.6547 | 0 |
| **420** | 678.9901 | 526.2998 | 749.8631 | 618.6455 | 0 |
| **421** | 674.6470 | 524.7834 | 752.0654 | 614.6701 | 0 |

325 rows × 5 columns

FIGURE 4. 1 UNLOADING DATA VISUALIZATION

The figure below refers to loading status of the well (1)

Out[6]:

| | FlowRate | StaticPressure | CasingPressure | TubingPressure | Loading |
|---|---|---|---|---|---|
| **132** | 552.5272 | 522.9531 | 873.0584 | 501.9169 | 1 |
| **133** | 552.5272 | 523.7603 | 885.5612 | 500.0335 | 1 |
| **134** | 552.5272 | 523.8813 | 901.1707 | 500.1304 | 1 |
| **135** | 552.5272 | 522.7551 | 911.7425 | 497.1399 | 1 |
| **136** | 552.5272 | 523.0791 | 919.0675 | 496.9730 | 1 |
| ... | ... | ... | ... | ... | ... |
| **315** | 667.4582 | 519.2561 | 785.7444 | 559.2784 | 1 |
| **316** | 667.4582 | 516.6292 | 805.6303 | 515.0887 | 1 |
| **317** | 483.0143 | 515.7208 | 818.7723 | 579.3625 | 1 |
| **318** | 699.7637 | 519.5049 | 812.7854 | 643.1445 | 1 |
| **319** | 780.0994 | 517.2853 | 777.4763 | 674.6049 | 1 |

97 rows × 5 columns

FIGURE 4. 2 LOADING DATA VISUALIZATION

In the figure 4:3 the casing pressure is plotted vs the gas flowrate.

```
In [11]: # sns.scatterplot(data = df , x = 'FlowRate', y = 'TubingPressure', hue = 'Loading')
         sns.scatterplot(data = df , x = 'FlowRate', y = 'CasingPressure', hue = 'Loading')

Out[11]: <AxesSubplot:xlabel='FlowRate', ylabel='CasingPressure'>
```

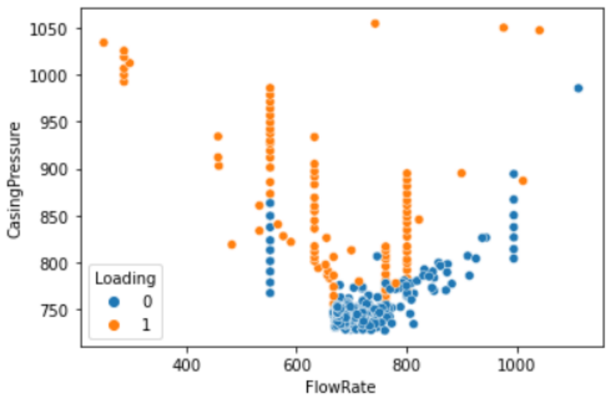


FIGURE 4. 3 CASING PRESSURE VS FLOWRATE

In the figure 4:4 the tubing pressure, casing pressure and static pressure are plotted vs the gas flow rate

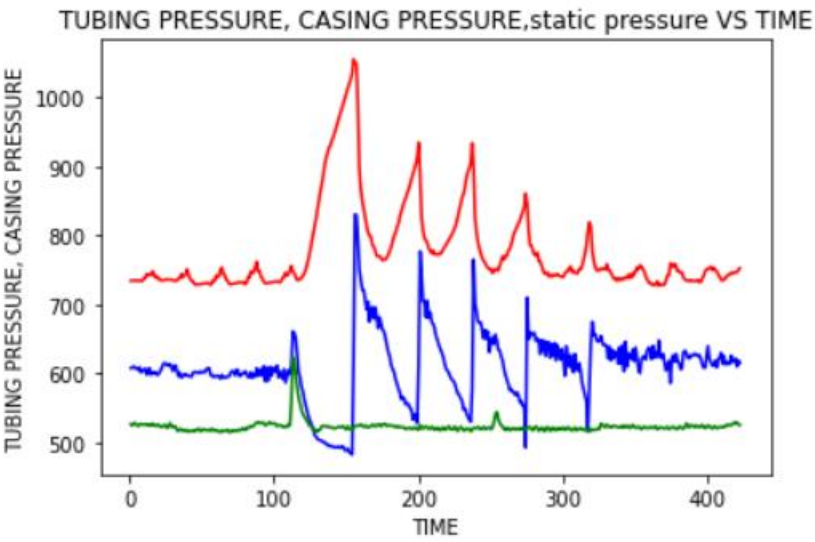


**FIGURE 4. 4** TUBING, CASING, STATIC PRESSURES VS TIME

## 4.1.2 Data preprocessing:

In the figure 4:5 using missingno function the data is visualized to find out whether the data is complete or not.

```
In [43]: import missingno as msno
         msno.matrix(f_14)

Out[43]: <AxesSubplot:>
```

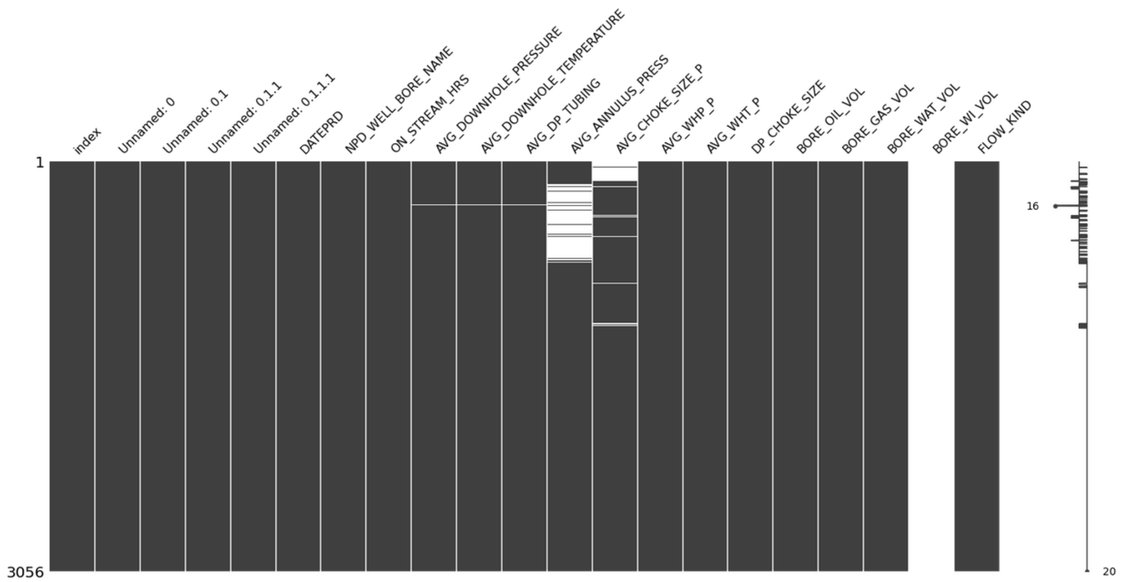


**FIGURE 4. 5** DATA CHECK USING MISSINGNO METHOD

In the figure 4:6 the missing data is interpolated.

```
In [44]: f_14 = f_14.interpolate(method='linear', axis=0, )

In [45]: msno.matrix(f_14)
Out[45]: <AxesSubplot:>
```

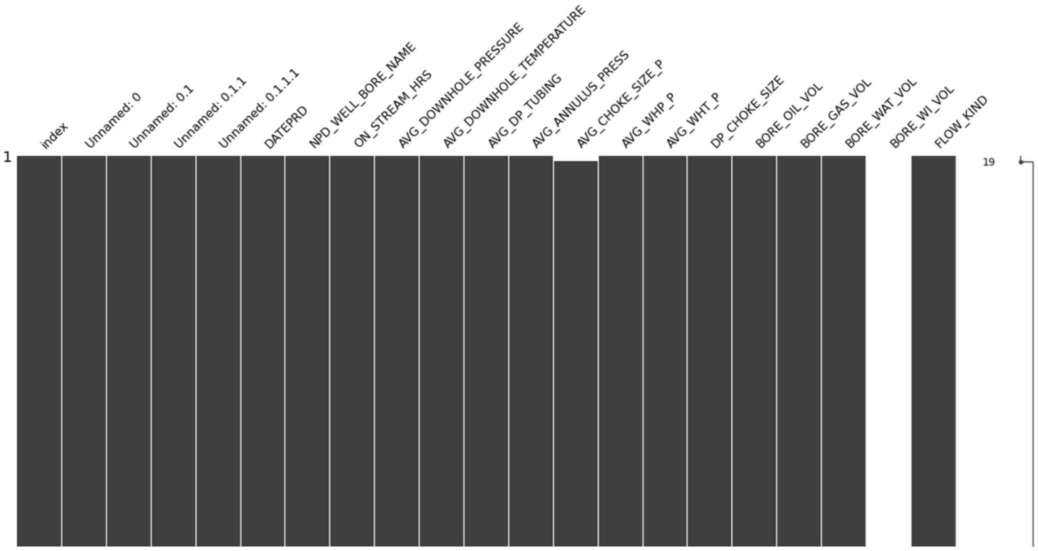


FIGURE 4. 6 DATA INTEGRATION USING INTERPOLATE METHOD

### 4.1.3 Data modeling:

In the figure 4:7 shows the data modeled using K-nearest-neighbors (KNN)

```
In [31]: from sklearn.neighbors import KNeighborsRegressor
         from sklearn.model_selection import train_test_split
         model= KNeighborsRegressor(n_neighbors=3)
         X =  df[["FlowRate", "StaticPressure" ,"CasingPressure" ,"TubingPressure"]]
         y = df["Loading"]
         X_train, X_test, y_train, y_test=train_test_split(X,y)
         model.fit(X_train,y_train)

Out[31]: KNeighborsRegressor(n_neighbors=3)
```

FIGURE 4. 7 DATA MODELING ALGORITHM

### 4.1.4 The results:

In the figure 4:8 the model is run, the accuracy of this model is 0.93 and all the data is tested.

```
In [32]: print("Training set score: {:.2f}".format(model.score(X_train, y_train)))
         print("Test set score: {:.2f}".format(model.score(X_test, y_test)))
         results1 = model.predict(X_test)

         Training set score: 0.93
         Test set score: 0.93

In [33]: z = ([[1800,900,1200,500]])
         model.predict(z)

Out[33]: array([0.66666667])

In [34]: results1

Out[34]: array([0.        , 0.        , 0.        , 0.        , 0.        ,
                0.        , 0.        , 1.        , 0.        , 0.        ,
                0.        , 0.        , 1.        , 1.        , 0.        ,
                0.        , 0.        , 1.        , 0.        , 0.        ,
                0.        , 0.        , 0.        , 0.        , 0.        ,
                0.        , 0.        , 0.        , 0.        , 1.        ,
                0.        , 0.        , 1.        , 0.        , 0.        ,
                0.        , 0.        , 0.        , 1.        , 0.        ,
                1.        , 1.        , 0.        , 1.        , 1.        ,
                1.        , 1.        , 0.66666667, 0.        , 1.        ,
                0.        , 0.        , 0.        , 0.        , 0.        ,
                0.        , 0.        , 0.        , 0.        , 0.33333333,
                1.        , 0.        , 0.        , 0.        , 0.        ,
                0.        , 0.        , 0.        , 0.        , 0.        ,
                0.        , 0.        , 0.        , 0.        , 0.        ,
                0.        , 1.        , 0.        , 0.        , 0.        ,
                0.        , 0.        , 1.        , 0.        , 0.66666667,
                0.        , 0.        , 1.        , 1.        , 0.        ,
                0.        , 0.        , 0.        , 0.        , 0.        ,
                1.        , 1.        , 0.        , 1.        , 0.        ,
```

FIGURE 4. 8 MODELING FINAL RESULT

## 4.2 Case study for the well FN-21:

### 4.2.1 Data visualization:

As it seen in Figure 4:9, in the data below there are missing features, which is flowline pressure and average casing pressure, therefore the data is not appropriate to be modeled.

Out[11]:

| | AvgCsg | AvgTBG | FL_Temb | Chock_size | FL_pressure | Gas | Loading |
|---|---|---|---|---|---|---|---|
| **23** | NaN | 1957.500000 | 0.000000 | 3.200000 | NaN | 2318.857582 | 1 |
| **24** | NaN | 2047.400000 | 0.000000 | 4.640000 | NaN | 1100.778703 | 1 |
| **25** | NaN | 2030.000000 | 20.000000 | 5.175000 | NaN | 1561.086415 | 1 |
| **26** | NaN | 2003.806452 | 25.806452 | 3.929032 | NaN | 1733.521970 | 1 |
| **27** | NaN | 1937.303571 | 35.392857 | 4.400000 | NaN | 1891.611093 | 1 |
| **28** | NaN | 2044.500000 | 30.000000 | 4.500000 | NaN | 1661.655830 | 1 |
| **29** | NaN | 2044.500000 | 30.000000 | 4.500000 | NaN | 1706.867222 | 1 |
| **30** | NaN | 2044.500000 | 30.000000 | 4.683333 | NaN | 1675.895988 | 1 |
| **31** | NaN | 2044.500000 | 30.000000 | 4.000000 | NaN | 1715.108798 | 1 |
| **32** | NaN | 2044.500000 | 30.000000 | 4.000000 | NaN | 1737.394988 | 1 |
| **33** | NaN | 2065.548387 | 30.000000 | 4.000000 | NaN | 2001.343612 | 1 |
| **34** | NaN | 2175.000000 | 27.000000 | 4.000000 | NaN | 1945.806565 | 1 |
| **35** | NaN | 2175.000000 | 27.000000 | 4.183333 | NaN | 1858.044476 | 1 |
| **36** | NaN | 2175.000000 | 26.000000 | 4.000000 | NaN | 1742.608428 | 1 |
| **37** | NaN | 2175.000000 | 26.000000 | 4.000000 | NaN | 1746.787857 | 1 |
| **38** | NaN | 2175.000000 | 31.548387 | 4.000000 | NaN | 1722.491830 | 1 |
| **39** | NaN | 2175.000000 | 30.000000 | 4.000000 | NaN | 1698.936009 | 1 |
| **40** | NaN | 2184.822581 | 34.838710 | 4.000000 | NaN | 1660.308803 | 1 |
| **41** | NaN | 2199.790323 | 34.612903 | 4.000000 | NaN | 1671.817178 | 1 |
| **42** | NaN | 2247.500000 | 35.000000 | 4.000000 | NaN | 1687.273797 | 1 |
| **43** | NaN | 2303.629032 | 35.000000 | 4.000000 | NaN | 1687.794769 | 1 |

FIGURE 4. 9 DATA LOADING VISUALIZATION

Moreover, the figure 4:10 shows that the flowline pressure and average casing pressure are missed.

```
In [13]: import missingno as msno
         msno.matrix(df)

Out[13]: <AxesSubplot:>
```

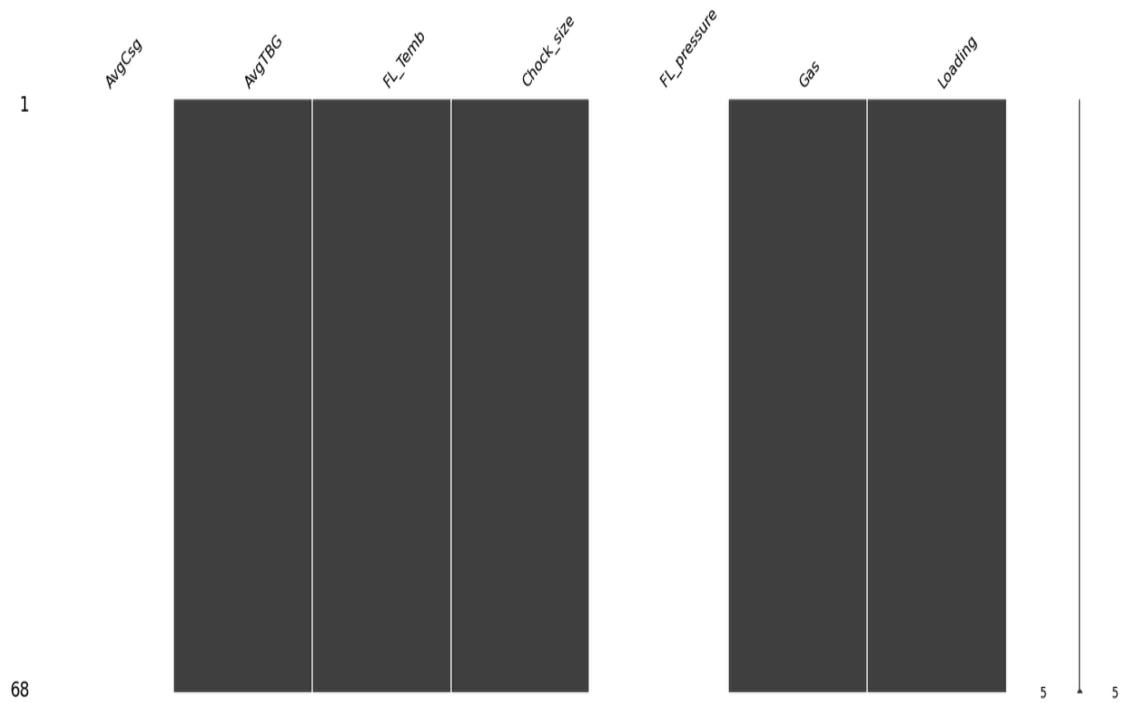


**FIGURE 4. 10 FN21 DATA CHECK USING MISSINGNO METHOD**

From all above it appears that the data acquired from this well is not capable of being used in this project.

## 4.3 Case study for the well FN4-7:

### 4.3.1 Data visualization:

The figures 4:11 and 4:12 shows the data of the well FN4-7, which well be preprocessed and then modeled.

The figure below refers to unloading status of the well (0)

Out[16]:

| | AvgCsg | TBG_PRESSURE | FL_TEMP | GAS | Loading |
|---|---|---|---|---|---|
| 0 | 373.000000 | 959.338710 | 30.000000 | 1100.000000 | 0 |
| 1 | 580.000000 | 1160.000000 | 30.000000 | 1100.000000 | 0 |
| 2 | 580.000000 | 1160.000000 | 30.000000 | 1100.000000 | 0 |
| 3 | 580.000000 | 1160.000000 | 30.000000 | 1100.000000 | 0 |
| 4 | 580.000000 | 1160.000000 | 30.000000 | 1100.000000 | 0 |
| 5 | 580.000000 | 1160.000000 | 30.000000 | 1100.000000 | 0 |
| 6 | 580.000000 | 1160.000000 | 30.000000 | 1100.000000 | 0 |
| 7 | 580.000000 | 1160.000000 | 30.000000 | 1100.000000 | 0 |
| 8 | 580.000000 | 1160.000000 | 30.000000 | 1100.000000 | 0 |
| 9 | 580.000000 | 1160.000000 | 30.000000 | 1100.000000 | 0 |
| 10 | 580.000000 | 1160.000000 | 30.000000 | 1100.000000 | 0 |
| 11 | 580.000000 | 1160.000000 | 30.000000 | 1100.000000 | 0 |
| 12 | 580.000000 | 1160.000000 | 30.000000 | 1100.000000 | 0 |
| 13 | 1087.500000 | 1268.266667 | 30.000000 | 1100.000000 | 0 |
| 14 | 1667.500000 | 1392.000000 | 30.000000 | 1008.333300 | 0 |
| 15 | 1667.500000 | 1392.000000 | 30.000000 | 1008.333300 | 0 |
| 16 | 1667.500000 | 1392.000000 | 30.000000 | 1055.645161 | 0 |
| 17 | 1667.500000 | 1392.000000 | 30.000000 | 1172.580645 | 0 |
| 18 | 1667.500000 | 1392.000000 | 30.000000 | 1323.333333 | 0 |
| 19 | 1667.500000 | 1507.532258 | 30.000000 | 1150.000000 | 0 |
| 20 | 1926.428571 | 1820.785714 | 30.000000 | 1150.000000 | 0 |

FIGURE 4. 11 UNLOADING DATA VISUALIZATION

The figure below refers to loading status of the well (1)

Out[8]:

| | AvgCsg | TBG_PRESSURE | FL_TEMP | GAS | Loading |
|---|---|---|---|---|---|
| **22** | 2030.000000 | 1882.100000 | 1.000000 | 1.000 | 1 |
| **23** | 2030.000000 | 1882.100000 | 1.000000 | 1.000 | 1 |
| **24** | 2030.000000 | 1882.100000 | 20.000000 | 1.000 | 1 |
| **25** | 2073.016667 | 1882.728333 | 19.166667 | 1.000 | 1 |
| **26** | 2175.000000 | 1885.000000 | 1.000000 | 1.000 | 1 |
| **27** | 2175.000000 | 1885.000000 | 15.000000 | 1.000 | 1 |
| **28** | 1885.000000 | 1885.000000 | 38.000000 | 765.352 | 1 |
| **29** | 1885.000000 | 1885.000000 | 38.000000 | 765.352 | 1 |
| **30** | 1885.000000 | 1885.000000 | 38.000000 | 765.352 | 1 |
| **31** | 1867.225806 | 1875.645161 | 38.000000 | 765.352 | 1 |

FIGURE 4. 12 DATA LOADING VISUALIZATION

In the figure 4:13 the average casing pressure is plotted against the gas flow rate.

```
In [15]: sns.scatterplot(data = df , x = 'GAS', y = 'AvgCsg', hue = 'Loading')
Out[15]: <AxesSubplot:xlabel='GAS', ylabel='AvgCsg'>
```

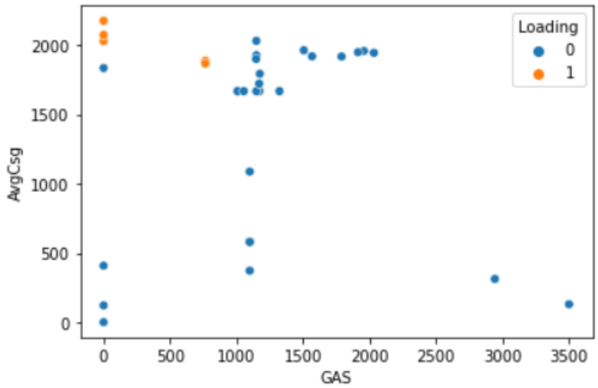


FIGURE 4. 13 GAS FLOWRATE VS AVERAGE CASSING PRESSURE

In figure 4:14 the average tubing pressure is plotted against the gas flow rate.

```
In [33]: sns.scatterplot(data = df , x = 'GAS', y = 'TBG_PRESSURE', hue = 'Loading')

Out[33]: <AxesSubplot:xlabel='GAS', ylabel='TBG_PRESSURE'>
```

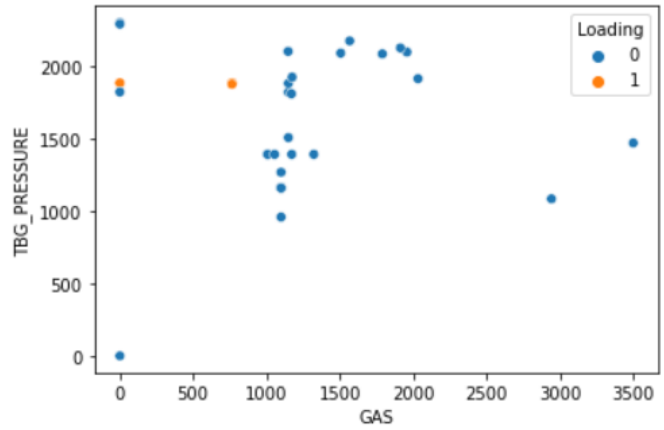


FIGURE 4. 14 GAS FLOWRATE VS TUBING PRESSURE

In the figure below the average casing and tubing pressure is plotted against the gas flowrate

```
In [32]: plt.plot(df['Data'],df['TBG_PRESSURE'],color='blue')
         plt.plot(df['Data'],df['AvgCsg'],color='red')
         plt.title('TUBING PRESSURE, CASING PRESSURE VS TIME')
         plt.xlabel('TIME')
         plt.ylabel('TUBING PRESSURE, CASING PRESSURE')
         plt.show()
```

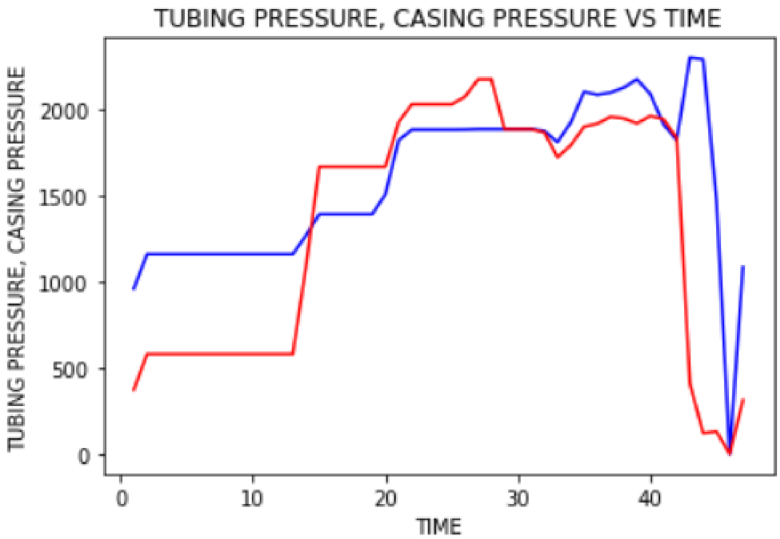


FIGURE 4. 15 TUBING,CASING PRESSURES VS TIME

### 4.3.2 Data preprocessing:

In the figure 4:16 using missingno function the data is visualized to find out whether the data is complete or not.

```
In [34]: import missingno as msno
         msno.matrix(df)

Out[34]: <AxesSubplot:>
```

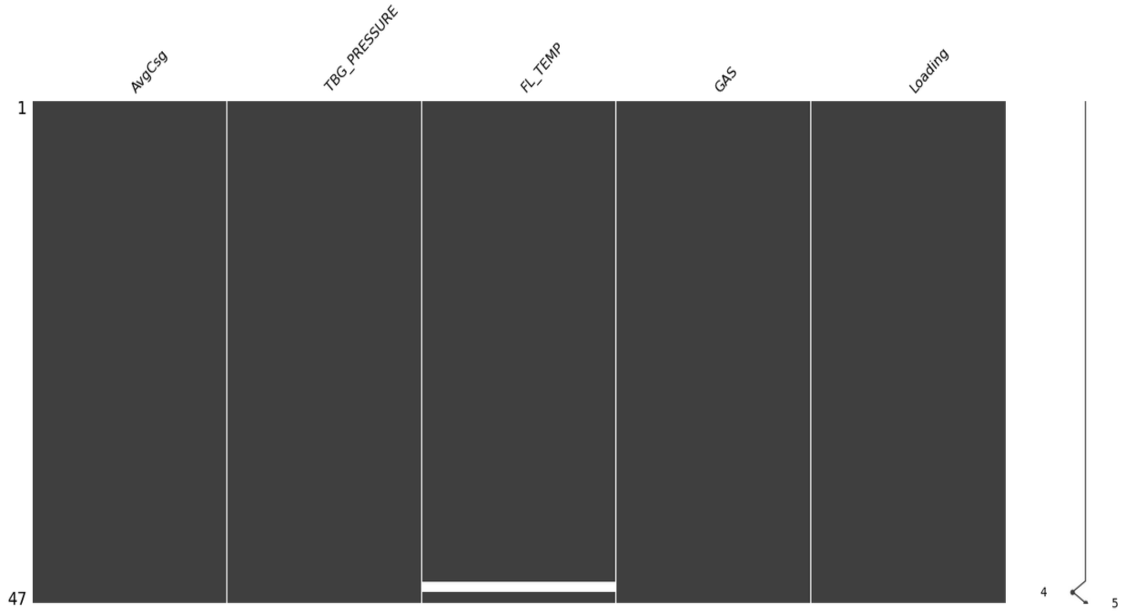


FIGURE 4. 16 FN4-7 DATA CHECK USING MISSINGNO METHOD

In the figure 4:17 the missing data is interpolated.

```
In [39]: df = df.interpolate(method='linear', axis=0, )

In [40]: msno.matrix(df)

Out[40]: <AxesSubplot:>
```

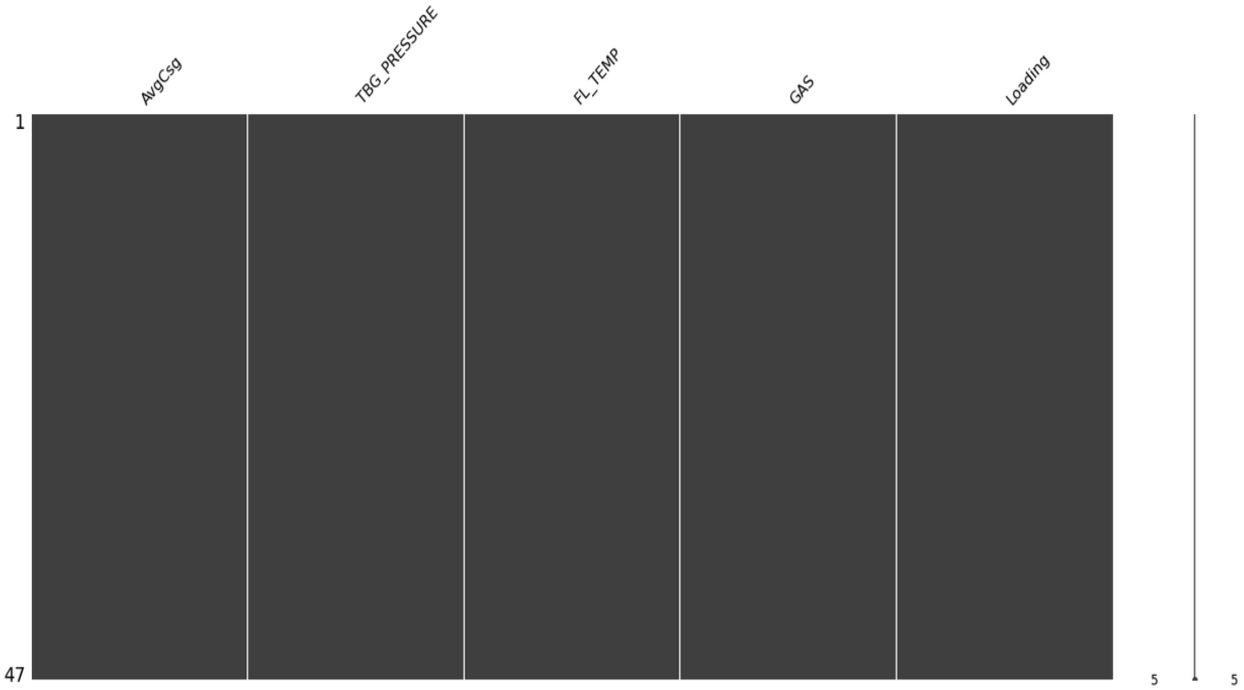


FIGURE 4. 17 DATA INTEGRATION USING INTERPOLATE METHOD

### 4.3.3 Data modeling:

In the figure 4:18 shows the data modeled using K-nearest-neighbors (KNN)

```
In [41]: from sklearn.neighbors import KNeighborsRegressor
         from sklearn.model_selection import train_test_split
         model = KNeighborsRegressor(n_neighbors=3)
         x = df[["GAS", "FL_TEMP" ,"AvgCsg" ,"TBG_PRESSURE"]]
         y = df["Loading"]
         x_train, x_test, y_train, y_test = train_test_split(x,y, test_size=0.3)
         model.fit(x_train, y_train)

Out[41]: KNeighborsRegressor(n_neighbors=3)
```

FIGURE 4. 18 DATA MODELING

### 4.3.4 The results:

In the figure 4:19 the model is run, the accuracy of this model is 0.92 and all the data is tested.

```
In [35]: print("Training set score: {:.2f}".format(model.score(x_train, y_train)))
         print("Test set score: {:.2f}".format(model.score(x_test, y_test)))
         results1 = model.predict(x_test)

         Training set score: 0.92
         Test set score: 1.00

In [36]: predictions = model.predict([[1000, 20, 2000, 1800], [1500, 25,2200, 1500]])
         predictions

Out[36]: array([0.33333333, 0.        ])

In [38]: results1

Out[38]: array([1., 0., 0., 1., 0., 0., 1., 0., 0., 0., 0., 0., 0., 0., 0.])
```

FIGURE 4. 19 MODEL FINAL RESULT

# Conclusions and recommendations

## 5.1 Conclusions:

1- With data contains at least, features of casing pressure, static pressure, tubing pressure, flow line pressure and temperature and the gas flow rate, there will be a chance to build a model that anticipate the status of the well either loading or unloading.

2- Python programing language was used to in visualizing, preprocessing and modelling the data.

3- Hence using K-nearest neighbors regression algorithm, the loading status is predicted.

4- One of the strengths of k-NN is that the model is very easy to understand, and often gives reasonable performance without a lot of adjustments.

5- Building the nearest neighbors model is usually very fast, but when your training set is very large (either in number of features or in number of samples) prediction can be slow. When using the k-NN algorithm, it's important to preprocess your data.

6- This approach often does not perform well on datasets with many features (hundreds or more), and it does particularly badly with datasets where most features are 0 most of the time.

7- Using Missingno method to interpolate data according on how data move and change from one period to another.

## 5.2 Recommendation:

A real data monitoring system is recommended to be fed with a spontaneous data, therefore this model is used to prevent the well from being dead, and we mean by died that well stop producing gas. Real data workflow base hence prevention of:

A- Time loss.

B- Cost loss.

C- The inverse flow to the near wellbore region.

# References

[1] Amir Ansari, Ibrahim Fathi, Fatemeh Belyadi, Ali Takbiri and Hoss Belyadi, 2018, "Data based smart model for real time diagnostics in Marcellus Shales via Machine Learning".

[2] Andreas C. Müller and Sarah Guido, 2016-09-22 first release, "Introduction to Machine Learning with Python (A guide for data scientists)"

[3] Ayush Rastogi, 2019, "onset of liquid loading in large diameter inclined pipes".

[4] Dr. Gioia Falcone, 2008-5, "Decision Matrix for Liquid Loading in Gas Wells for Cost/Benefit Analyses of Lifting".

[5] Kalpesh Patel and Rohit Patwardhan, 2019-3-15, "Machine Learning in Oil & Gas Industry: A Novel Application of Clustering for Oilfield Advanced Process Control".

[6] M.F.Riza, A.R.Hasan and C.S.Kabir, August 18 2016 "A Pragmatic Approach To Understanding Liquid Loading in Gas Wells".

[7] Subhashini Sankar, S. Arul Karthi ,2019-6-6, "Study of identifying Liquid Loading in Gas well Deliquification Techniques".

[8] Tarek Ahmed, 2000, "Reservoir Engineering Handbook".