UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

PREDICTIVE MODELING IN FUGITIVE EMISSIONS TESTING

A THESIS

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

Degree of

MASTER OF SCIENCE

By

ABIGAIL HOVORKA
Norman, Oklahoma
2020

PREDICTIVE MODELING IN FUGITIVE EMISSIONS TESTING

A THESIS APPROVED FOR THE
SCHOOL OF INDUSTRIAL AND SYSTEMS ENGINEERING

BY THE COMMITTEE CONSISTING OF

Dr. Shivakumar Raman

Dr. Zahed Siddique

Dr. Kash Barker

Dr. Pedro Huebner

Dr. Raghu Madhavan

Acknowledgements

This project was completed in collaboration with a team of students from the University of Oklahoma including: Brandon Mansur, Milad Najafbeygi and Abigail Hovorka. The primary contact from Schlumberger who aided the team was Dr. Raghu Madhavan. The primary advisor from the University of Oklahoma was Dr. Zahed Siddique. Contributions from these individuals are mentioned throughout the text.

Contents

List of Tables

List of Figures

List of Equations

Abstract

Greenhouse gases have become an increasingly significant issue in the last few decades. As a response, many organizations have sought to tighten their regulations on their operations to reduce their contributions to greenhouse gases. The International Organization for Standardization has a standard 15848-1 that classifies industrial valves for the oil and gas industry. They too have aimed to tighten their regulations, including this specific standard. However, the current requirements from ISO 15848-1 has made it extremely difficult for manufacturers to get any industrial valves and seals passed. This begs the question, are the new tightness classifications for the ISO standard appropriately relating the test gas, Helium, to the allowable Methane leakage concentration? And with that, is Helium even the best option for a testing gas? To investigate this, a series of experiments were conducted to collect Helium and Argon leak rate data under many temperature and pressure conditions. With this data, a Helium/Argon leak rate ratio model was created with machine learning techniques. Using this model, an Ar/CH_4 multiplier, and diffusion modeling, the ISO 15848-1 tightness classes can be assessed for their accuracy. A disconnect between the ISO 15484-1 Helium and Methane requirements has been identified and there is a call to reconsider the Helium requirements. In addition, a suggestion to investigate Argon as an alternative leakage test gas is also raised.

Keywords: fugitive emissions, methane leak testing, ISO 15848

1. Introduction

Fugitive Methane emission from the oil and gas industry's exploration endeavors along with production and supply lines have raised serious concerns regarding a contribution to greenhouse gases (Daniels & LeBoeuf, 2019). Atmospheric Methane poses a particularly great risk "due to its capacity to trap by volume 28 times more heat than carbon dioxide" (Daniels & LeBoeuf, 2019). "Fugitive emissions from valves account for 60% of the total Methane emissions from a refinery, with as much as 80% of the leakage, per valve, occurring at the valve stem" (Daniels & LeBoeuf, 2019). The primary issue here is that valve leakage of fugitive emissions involving Methane is grossly significant for many years (Daniels & LeBoeuf, 2019). That is until organizations such as the Oil and Gas Climate Initiative (OGCI) and various governmental entities worldwide have decided to intervene (Daniels & LeBoeuf, 2019). It is obvious that there needs to be something done with fugitive emissions across barriers for industrial valves in the oil and gas industry.

The Intergovernmental Panel on Climate Change (IPCC) Guidelines divide the oil and gas industry into three broad categories: oil and gas production; crude oil transportation and refining; and natural gas processing, transportation and distribution. All three categories are responsible for contributing to the issue that is fugitive emissions. IPCC Guidelines for national greenhouse gas emission inventories fugitive emission from oil and gas operations as emissions from all non-combustion sources (IPCC Guidelines for National Greenhouse Gas Inventories, 2019). The general definition of fugitive emissions given in the IPCC Guidelines is "an intentional or unintentional release of gases from anthropogenic activities excluding the combustion of fuels". In general, fugitive emissions from oil and gas activities may be attributed to fugitive equipment leaks, process venting, evaporation losses, disposal of waste gas streams

(e.g., by venting or flaring), and accidents and equipment failure (IPCC Guidelines for National Greenhouse Gas Inventories, 2019). Some of the key factors affecting the amount of fugitive emissions are based on equipment used, design of the systems, maintenance, operating conditions and other factors. Different agencies have regulations, testing and qualification guidelines corresponding to Methane fugitive emissions such as ISO 15848, TA-Luft, and API 622, 624 and 641 (Kazeminia & Bouzid, 2015) (Daniels & LeBoeuf, 2019). All of which are now being called on to reflect the new regulations outlined by new legislation and guidelines.

The focus of this project is the classification of fugitive emissions of Methane from industrial valves in the oil and gas industry outlined by the International Organization for Standardization (ISO) 15848-1 standard. Within this standard, 'fugitive emission' is specifically defined as a "chemical or mixture of chemicals, in any physical form, which represents an unanticipated or spurious leak from equipment on an industrial site" (International Organization for Standardization, 2015). Since fugitive emissions are an unavoidable matter, it is important to understand their behaviors in regard to the oil and gas industry. ISO 15848-1 outlines a leakage measuring procedure, a leak rate classification system (tightness classes) and qualification procedures for fugitive emissions involving industrial valves containing Methane (International Organization for Standardization, 2015). These classifications are used by the valve manufacturers to categorize the valves' industrial applications. Manufacturers often do not use Methane to test these valves in their facilities because it is an expensive gas, can harm the environment, and it is at risk for exploding when under high amounts of pressure (CCOHS, 2020). When Methane is not used, the next choice gas for testing industrial valves is Helium. Helium is a common testing gas due to its small molecular size being easy to work with for leakage tests and it is more affordable than Methane (TQC, n.d.). ISO 15848-1 outlines specific

Helium leak rates for three different tightness classes, each of which are related to an allowable Methane leakage concentration. However, Helium is difficult to store and there is currently a Helium shortage that has lasted for a few decades. Helium is becoming less ideal as a test gas as the need to conserve as much stored harvested Helium as possible increases and the available supply decreases (Vishik). In addition, the relationship between Helium and Methane is not made clear by ISO 15848-1 (Baars). Alternative test gases for leak testing applications are being taken under consideration and thus far, Argon is a great candidate (Chamberlain, 2014). To investigate these claims further, the focus of this project is Methane fugitive emission across barriers on shafts and to investigate how the ISO guidelines for testing relate Helium to Methane. To do this, it is necessary to: integrate permeation through barriers and diffusion of gases in air to compare the proposed guidelines; relate leak rates across barriers, through experimentation, for Helium, Argon and Methane; and develop a Helium/Argon leak rate ratio model to estimate Methane leakages through conversion. Once this model is created, the associated diffusion modeling It is through this that the similarity between Helium and Methane in fugitive emissions valve testing can be assessed and the possibility of Argon as an appropriate test gas can be investigated.

1.1 Problem Statement

The objective of ISO 15848-1 “is to enable classification of performance of different designs and constructions of valves to reduce fugitive emissions” (International Organization for Standardization, 2015). In application, these valves are used by oil and gas companies for the transportation of Methane gas. Hence, in application the fugitive emissions contain Methane leakages. In recent years, legislation around the world have criticized fugitive emissions

standards and have sought to adjust them to make them more strict in an effort to support new environment protection laws (Kazeminia & Bouzid, 2015). ISO 15848-1 provides Helium leak rates for valve testing facilities to meet that will theoretically translate the valve's performance to a specific allowable Methane leakage concentration. The ISO standard also outlines how to take these measurements using Helium as a test fluid. However, with this standard being revised to fit newer legislation, there is not a clear definition of the relationship between Helium and Methane within the ISO standard (Baars). Hence, it cannot be assumed that the valves' behavior in application is being accurately modeled by using the test Helium leak rate targets. This is an issue because manufacturers are having a difficult time getting their valves approved by the international standards when using Helium leakages (Baars). It seems as though the new Helium tightness classifications are too strict and thus not displaying an accurate relationship between the allowable Methane concentrate and the target Helium concentrate used for testing scenarios. This pushes some manufacturers to take the expensive route with safety risks and environmental risks by testing with Methane in an attempt to meet the standard (Baars) (U.S. National Library of Medicine, n.d.). An additional issue is the possibility of improperly calibrated valves being used in oil and gas application, could ultimately contribute to global warming or climate change if they malfunction (U.S. National Library of Medicine, n.d.). Considering that this standard is internationally recognized, these issues apply to companies throughout the world. Countries such as the United States, Europe and Japan are concerned with this standard (Patil, 2013) (Kazeminia & Bouzid, 2015).

The best way to solve this problem is to find the proper relationship between the Helium test leak rate targets and the allowable Methane leak rate concentration. These comparisons can be made starting by collecting Helium and Argon leak rate data. Methane cannot be used for

testing with because of its potential danger and its associated expenses. With Helium and Argon leakages, the individual leak rate ratios for each set of experimental conditions can be determined. These ratios will show a behavioral comparison between the two gases and can be modeled using regression analysis tools. Via third party experimentation, the leak rate ratio behavioral comparison between Argon and Methane produces a ratio coefficient of 1.5 which can be applied to the He/Ar leak rate ratio modeling. With this, He/CH_4 leak rate ratio values can be predicted. These ratios can then be used to draw comparisons across the ISO 15848 tightness classes. Better understanding the necessary Helium leak rate targets for the ISO tightness classes will allow for companies to be able to properly classify their valves and even have a higher likelihood of getting their valves to pass the standards, which has been an ongoing problem since the stricter standards were issued.

2 Literature Review

Some important things to understand for the purposes of this project are: Helium leak testing, why is Helium used, the safety measures taken for testing valves, factors on permeability, the current state of Helium availability, the ISO tightness classifications, diffusion's role in context to this problem, and what other researchers are currently doing to address these issues.

2.1 Helium leak testing

In manufacturing, "Helium is used to find small leaks" in individual parts and in assemblies (TQC, n.d.). The Helium is used as a tracing gas that can help manufacturers identify unwanted leaks (TQC, n.d.). Some factors that can cause an unwanted leak include porosity,

small defects in the welds, micro-cracks, defective seals, incorrect components assembly, etc. Helium is also used as a tracing gas for instances where the manufacturer expects some amount of leakage from their part or assembly and they want to be able to monitor that leakage behavior. Included in this are parts and assemblies that must undergo fugitive emissions testing, which is the focus of this research.

Different methods are used to conduct some of these leak tests: ultrasonic measurement and bubble test, pressure decay, Helium spray, Helium sniff, Helium accumulation, and vacuum systems (TQC, n.d.). For fugitive emissions testing, ISO 15848-1 outlines a specific measuring process for testing valve leakages (International Organization for Standardization, 2015). This method involves a pressure change caused by gas accumulation. In this, there is measurement of Helium that has escaped across a seal and into another chamber. Thus, providing a measurement of Helium leakage across barriers.

2.1.1 Why use Helium or Argon

Helium is a widely used test gas because it is one of the smallest gas molecules and it is an inert gas (TQC, n.d.). Argon is also considered to be an appropriate test gas because, like Helium, it is inert, non-explosive, and non-toxic (Vacuum Instruments Corporation, n.d.). Neither Helium nor Argon are considered to be contributors to greenhouse gases, making them attractive options for leakage testing (World Health Organization, n.d.). Both gases can be safely discharged to the atmosphere without any adverse environmental effects. Some recommend attempting to recapture Helium due to shortages, but it can be rather uneconomical to do so (Chamberlain, 2014).

Both of these inert gases are in the atmosphere but are considered to be trace and extremely rare (World Health Organization, n.d.) (Vacuum Instruments Corporation, n.d.). Hence, “there is little ambient ‘noise’ to interfere with leak measurement” processes (Vacuum Instruments Corporation, n.d.). Both Helium and Argon are non-reactive, so there will not be any negative effect on the part or instruments as a result of any chemical change (Vacuum Instruments Corporation, n.d.) (TQC, n.d.). Helium has a melting point of -272.2˚C and Argon has a melting point of -189.34˚C, making them appropriate choices for industrial tests to sub-zero temperatures (Thomas Jefferson National Accelerator Facility - Office of Science Education, 2020) (Royal Society of Chemistry, 2020).

An important characteristic is that both gases can be detected by spectrometers, making it possible to even track their leakages (TQC, n.d.) (Vacuum Instruments Corporation, n.d.). However, it is easier to collect and keep Argon samples as opposed to Helium (Chamberlain, 2014). Helium is extremely difficult to recapture from the atmosphere and it is difficult to store since it is so small (Chamberlain, 2014).

Argon is more affordable than Helium as a testing gas. For example, to purchase a size 300 cylinder of industrial grade Argon gas costs $39.68/CL whereas the same size cylinder of industrial grade Helium gas costs $229.35/CL, which is 5.8 times more expensive than Argon (AirGas, 2020). Hence, Argon is an even more attractive candidate for being a primary testing gas.

2.1.2 Safety

Though both Helium and Argon are inert gases and can be exhausted to atmosphere, proper safety considerations to prevent asphyxiation risks should be taken. These gases when

bottled contain no Oxygen and therefore are considered asphyxiants (TQC, n.d.). The gases will displace Oxygen increasing the risk of Oxygen deprivation to the user, especially if the surroundings are not well-ventilated.

2.1.3 Factors on permeability and diffusion

The primary test fluids used in industry include Helium, Argon and Methane. Methane (CH_4) is classified in the group of flammable gases, it is hazardous to use it as a testing gas. Hence it is often necessary to use Helium or Argon as alternate test gases. Therefore, it is important to investigate the molecular similarities of these gases when it comes to permeability and diffusion. For this, atomic radius, and molecular weight are factors that have a significant impact on both permeability and diffusion, which can help better understand the gases' similarities. For the purposes of these comparisons, instead of using properties of Methane, the properties of Carbon are used for comparison instead. This is because the Hydrogen atoms are free to move around the Methane molecule, which does not affect the molecule's core size, hence not affecting its permeability. In the Table below, the atomic and covalent radii of Carbon, Helium, and Argon are outlined.

Atomic Number	Element Symbol	Atomic Radius [Å]	Covalent Radius [Å]
2	He	1.400	0.370
6	C	1.700	0.750
18	Ar	1.880	1.010

Table 1 - Molecular radii of He, C, and Ar

(Royal Society of Chemistry, 2020) (Royal Society of Chemistry, 2020) (Royal Society of Chemistry, 2020)

In Table 1, the atomic radius of Carbon is closer to the atomic radius of Argon rather than Helium. Carbon makes up the majority of Methane's substance. With this, it can be argued that

Argon will be more similar to Methane in its behavior. This encourages the investigation of Argon as a testing gas in place of Methane rather than Helium as a testing gas in place of Methane.

2.2 Global Helium shortage

Helium is becoming an increasingly scarce resource, which is concerning considering it is widely used across industries such as oil and gas, space technology, and medical instrumentation (Hope, 2019). Pricing of Helium is suffering in this shortage and it is easily influenced due to there being very few Helium suppliers (Hope, 2019). It is already a difficult element to capture since it floats freely around the atmosphere, but also the Helium reserves discovered by oil companies are running out and scientists are struggling to find new reserves (Carmin Chappell, 2019). This is the third global Helium shortage in the last 14 years and there is not much hope for a future of abundance in Helium (Murphy, 2019). Even if there is easy access to Helium in the future, that supply will also be limited and there is no way to stockpile the briefly available supplies due to Helium being extremely difficult and expensive to store (Murphy, 2019).

Some ways to try to work with the Helium shortage is the implement Helium storage and exploration policies to maintain the reserves and not sell it for below market rates (Vishik). Although this is helpful in the short term by making the supply steadier, the issue with the limited reserves of a resource that is not renewable is not addressed (Vishik). Another suggested solution for preserving Helium use is to limit wasteful applications (Vishik). For research applications, the installation of a recirculatory system is encouraged to recapture the used Helium (Vishik). However the cost to install a system like this is high and not all laboratories may be

able to implement this system or make the proper accommodations even if they see their savings emerge within a few years (Vishik). If Helium continues to be a primary resource for research and manufacturing, having these systems will become not optional (Vishik). It is crucial to find a solution to this because once all of the Helium has been extracted from the Earth, there is no way to recreate it (Vishik). And there are many medical applications where Helium cannot be replaced by another substance, increasing the importance to find some feasible solutions to the world's Helium shortage (Vishik). A realistic solution is to find an alternative for the applications that can accommodate another test fluid. This has become one of the driving motivations for this research.

2.3 ISO 15848-1 leakage tightness classifications

The International Organization for Standardization (ISO) is a worldwide non-governmental organization of national standards bodies (International Organization for Standardization, 2015). Their goal is to make things work by giving world-class specifications for products and systems to ensure quality, safety and efficiency. For the purposes of this report, the focus is on ISO 15848-1, which concerns industrial valves and classifications. "The objective of this part of ISO 15848 is to enable classification of performance of different designs and constructions of valves to reduce fugitive emissions" (International Organization for Standardization, 2015). This is for the application of flammable or inert gas at temperature while under pressure. The tests require for the fluid to be "Helium gas of 97% minimum purity & Methane of 97% minimum purity" (International Organization for Standardization, 2015). The classes vary in criteria due to the difference in valve operating conditions and hazards during industrial use. These varying conditions result in different levels of valve emission performance;

thus, the classes indicate the appropriate minimum leak rates for three different Helium and Methane conditions. The ISO 15848 tightness classes for Helium and Methane are shown in Tables 2 and 3, respectively.

Class	**Measured leak rate (mass flow)**	**Measured leak rate (mass flow)**	**Measured leak rate (volumetric flow)**	**Remarks**
	$mg \cdot s^{-1} \cdot m^{-1}$ stem perimeter (for information)	$mg \cdot s^{-1} \cdot mm^{-1}$ stem diameter through stem seal system	$mbar \cdot l \cdot s^{-1}$ per mm stem diameter through stem seal system	
AH[a]	$\leq 10^{-5}$	$\leq 3.14 \cdot 10^{-8}$	$\leq 1.78 \cdot 10^{-7}$	Typically achieved with bellow seals or equivalent stem (shaft) sealing system for quarter turn valves
BH[b]	$\leq 10^{-4}$	$\leq 3.14 \cdot 10^{-7}$	$\leq 1.78 \cdot 10^{-6}$	Typical achieved with PTFE based packings or elastomeric seals
CH[b]	$\leq 10^{-2}$	$\leq 3.14 \cdot 10^{-5}$	$\leq 1.78 \cdot 10^{-4}$	Typically achieved with flexible graphite-based packings
[a] Measured by the vacuum method [b] Measure by the total leak rate measurement method (vacuum or bagging)				

Table 2 - Tightness classes for stem (or shaft) seals with Helium

(International Organization for Standardization, 2015)

Class	**Measured leakage (sniffing method) ppmv**
AM	≤50
BM	≤100
CM	≤500

Table 3 -Tightness classes for stem (or shaft) seals with Methane

(International Organization for Standardization, 2015)

Table 2 contains units for a leak rate whereas Table 3 is in terms of concentration. Whether the test fluid is Helium or Methane, there are three classes A, B and C specified for leak rates at different conditions with an increasing order of allowable leak rates. Each class allows

for a different level of performance for an intended application. A manufacturer may perform tests for a specific tightness class based on what they want their equipment to do or the conditions they expect their equipment to experience.

2.4 Current and Past Research

ISO 15848-1 has been mentioned frequently in research studies before. Often, teams are trying to abide by the ISO standard with systems they have built or parts they have designed. For example, researchers in Canada were interested in predicting the leak rate through porous compression packing rings for the design of external sealing valves (Kazeminia & Bouzid, 2015). Their goal was to predict the leak rate through these seals and use that information to "design and select suitable compression packing for a maximum tolerated leak for a given application" (Kazeminia & Bouzid, 2015). However, this team explains how recent legislations have become "very strict on the amount of emission that is tolerated" (Kazeminia & Bouzid, 2015). Thus, requiring many "standards such as TA-Luft, ISO 15848-1 and API 622 and 624 and others" to be revised to respect the updated regulations (Kazeminia & Bouzid, 2015). Due to "the ubiquitous use of the yarned packing rings in the sealing of valves, and the strict regulations on fugitive emissions and the new environment protection laws, quantification of leak rate through [the seals] becomes more than necessary and a tightness criteria based design procedure must be developed" (Kazeminia & Bouzid, 2015). Their model consists of two elements: Darcy's model, which is a commonly used "numerical expedient for the simulation of multiphase fluid flow in porous materials", and the Klinkenberg slip effect , which is used to illustrate the material properties (Kazeminia & Bouzid, 2015). The research team used this model to predict values under certain conditions and compared the predicted leak rate values to leak rate values they

obtained experimentally. They graphed the results to observe the accuracy of their methodology as seen in Figures 1 and 2 (Kazeminia & Bouzid, 2015).

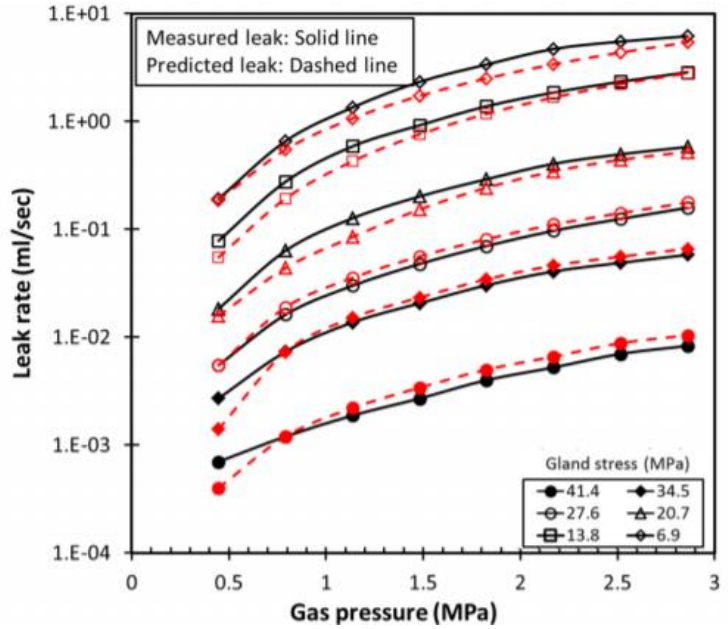


Figure 1 - Measured and predicted leak rate for packing GI versus gas pressure for Journal of Fluid Science and Technology study

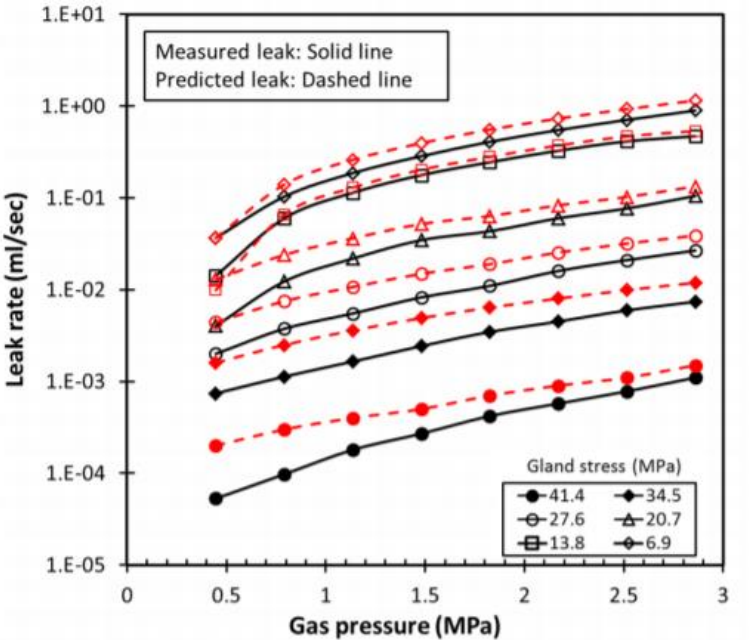


Figure 2 - Measured and predicted leak rate for packing GII versus gas pressure for Journal of Fluid Science and Technology study

(Kazeminia & Bouzid, 2015)

The team concluded that their model was able to predict the leak rate with reasonable accuracy (Kazeminia & Bouzid, 2015).

The significance of this study in relation to this thesis is that the research team found themselves with a similar problem where they needed to go outside of any fugitive emissions regulations to determine their safe and usable leak rates for packing rings. This is due to the strict legislation now surrounding these standards that all have been going through many revisions to meet the legislation (Kazeminia & Bouzid, 2015). This further supports the need to find proper Helium and Methane relationships for ISO 15848-1 to provide realistic standards to fit the new guidelines. Essentially, the team's project scope falls in line with that of this project.

Another group of researchers with Clarke Valve decided to design an entirely new valve to pass the new and improved ISO 15848-1 standard, rather than try to adjust the existing

equipment and seals to the new standards (Daniels & LeBoeuf, 2019). This team decided to focus on API 641 and ISO 15848 due to their common applications across industry. The API 641 standard is specific to quarter-turn valves and focuses on the life span of these valves at different temperatures with a constant pressure (Daniels & LeBoeuf, 2019). Both static and dynamic leakage measurements are recorded for this standard. ISO 15848-1 differs in that it has a range of distinct pressure values and temperature values for both control and isolation valves, which has been described in greater detail in chapter 2.3 of this thesis (Daniels & LeBoeuf, 2019) (International Organization for Standardization, 2015).

The team believes that there is a need for an entirely new valve design to reduce the fugitive emissions. Their goal was to out-perform the industry's 'low emissions' valve that routinely emits 500 ppmv per valve, making the 'low emissions' valve a large contributor to the "annual tonnage of product lost to the atmosphere" (Daniels & LeBoeuf, 2019). The newly designed valve from Clarke Valve is a deviation from current valves in operation currently worldwide, however it serves the same function (Daniels & LeBoeuf, 2019).

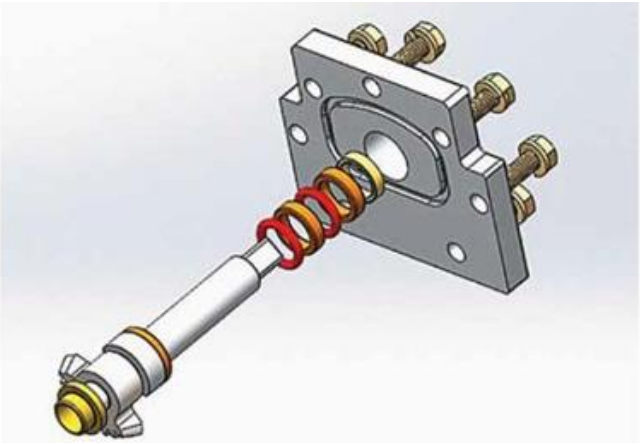


Figure 3 - Clarke Valve's proposed shutter valve bonnet and stem stack.

(Daniels & LeBoeuf, 2019)

During independent testing, in the case of API 641 testing the valve produced average emissions of 6.25 ppmv, which easily passes the API standard of 100 ppmv as a maximum allowable leakage (Daniels & LeBoeuf, 2019). For the ISO testing, the newly-designed valve consistently "allowed less than 10 ppmv over 100,000 mechanical cycles" which is more than appropriate for the ISO standard (Daniels & LeBoeuf, 2019). In conclusion, new valve designs can reduce emissions per valve as much as 95% making it an extremely attractive option to oil and gas producers worldwide and thus earning Clarke Valve millions of dollars in investments toward these new technologies (Daniels & LeBoeuf, 2019).

The significance of this study in relation to this thesis is that it describes a dire need for adjustments to accommodate the changing standards set for ISO 15848-1 in fugitive emissions testing. The industry desperately needs to address the concerns associated with the fugitive emissions standards for industrial valves. Going forward it is important to look at newer technologies to fit into the improved Helium and Methane correlations proposed by this thesis.

3 Data Collection

The purpose of the experiments is to collect static leak rate data of Argon and Helium under a range of pressure and temperature conditions. The collection of this data was a combined effort of Abigail Hovorka, Brandon Mansur and Milad Najafbeygi.

To collect this data, a testing vessel with plastic barrier seals on a stem that separate a high-pressure chamber from a low-pressure chamber was used. Figure 4 shows an example of a seal stack like the one used for the experiments. There are different layers of barrier seal stack that sometimes was used in varying configurations. The stack included, spring energized seals, spacers and v-ring seals. During the experiments, it was found that the spring-energized seals

often failed to seal at low-pressures if that seal had already faced a temperature change from hot to cold or vice-versa.

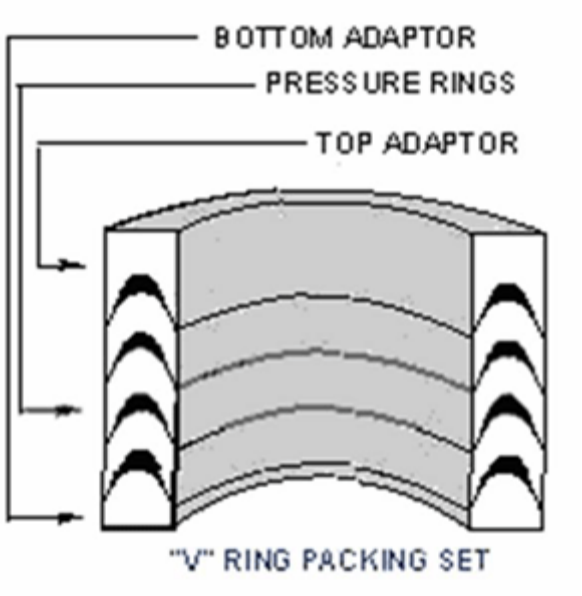


Figure 4 – Example of a V-ring barrier stack like what was used in the experiments

(VAC AERO International, 2013)

Figure 5 shows the position of barrier seals in the testing vessel indicated by the green arrows in the center pointing toward the black shapes in the center. The leakage collection sides A and B are indicated by the orange and blue arrows at the bottom of Figure 5. The high-pressure section is in the center indicated by the red arrow at the bottom of Figure 5. The items labeled “Spacer” with the blue arrows at the top of Figure 5 pointing toward the larger black shapes are metal sleeves that help to position the barrier seals into place in the center of the testing vessel. The shape in in the middle of Figure 5 labeled “Stem/Shaft” represents the shaft, or stem, that goes through the testing vessel and it is what the barrier seal stacks are surrounding.

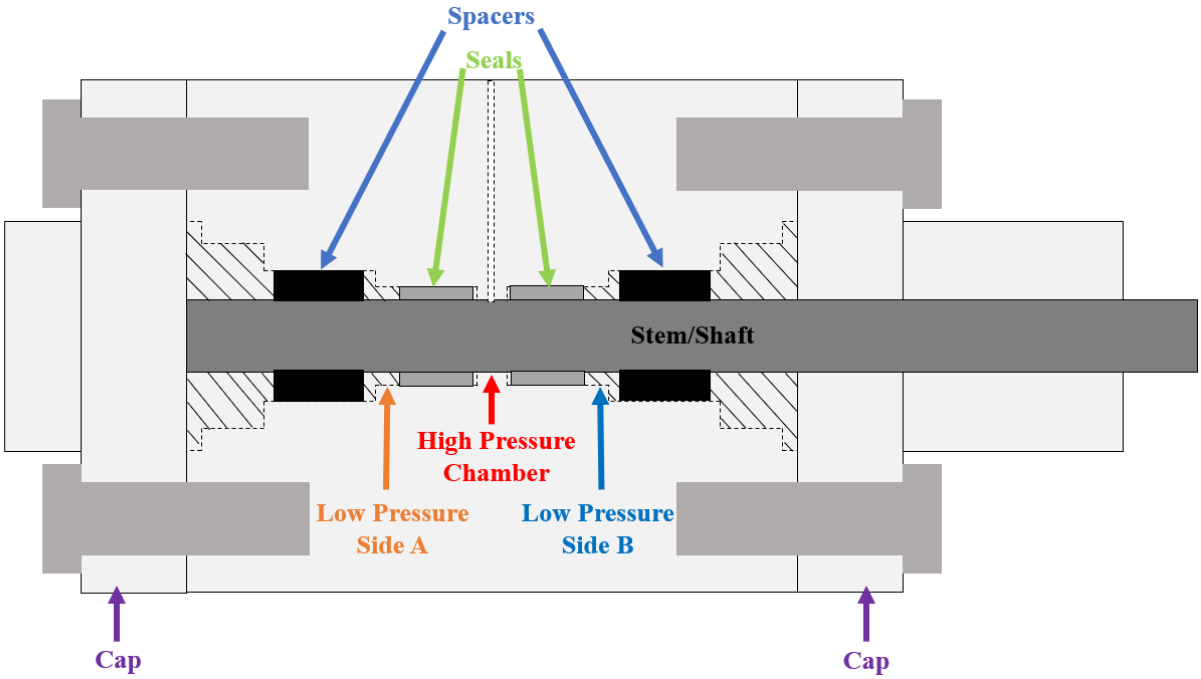


Figure 5 - Shows the position of barrier seals, the high-pressure chamber, and the low-pressure Side-A and Side-B

3.1 The System

The system built for this project was designed and executed by Brandon Mansur, who also helped to collect data to ensure its functionality. The operation and maintenance of this system was performed by Abigail Hovorka, Brandon Mansur and Milad Najafbeygi.

Note: Ideal Gas Laws were used to help produce the leak rate readings because the volume of the system was always a constant and the only variables were pressure and temperature, which were known for any given experiment.

This experimental setup is capable of achieving a temperature range of -70ºC to 240ºC. One of the most important tasks for designing this system was to ensure that the delivery method for temperature change would be consistent, reliable and replicable.

For low temperature tests, a cryogenic chamber that uses liquid Nitrogen as the cooling agent was designed. This chamber is a large container with a metal frame and galvanized metal panels with fiberglass insulation behind the panels (Figure 6).

Figure 6 - High-Pressure Gas Supply, Test and Collection System (front)

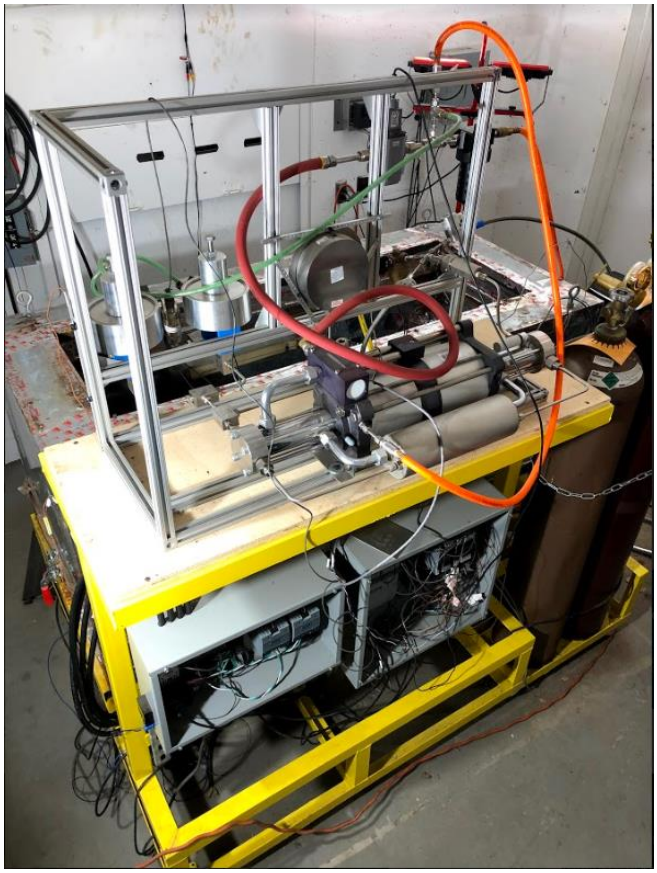


Figure 7 - High-Pressure Gas Supply, Test and Collection System (back)

Liquid Nitrogen, when released into the cooling chamber, vaporizes, and reduces the temperature. To minimize any thermal gradients inside the cooling chamber, two fans were installed on either end of the cooling chamber with the purpose of circulating and mixing the air inside the cryogenic chamber. The cooling process was done in two steps to minimize overshooting. First, reducing and stabilizing the temperature at a target value slightly above (~ 4˚C) the set point and in the second step, allowing the temperature to reach the set point. Once the set point is reached, liquid Nitrogen is sprayed to maintain the temperature. The temperature inside the chamber was monitored with four thermocouples attached to the interior walls. To minimize heat gain from the surroundings, walls of the chamber were lined with insulated layers that including reflective tape and plastic insulation sheets, as shown in Figures 8 and 9. Exit

holes for sensor wires and pressure channels were closed-off with spray foam. It can take nearly 24 hours for the barriers to reach the experimental temperature.

Figure 8 - Inside of the cooling chamber

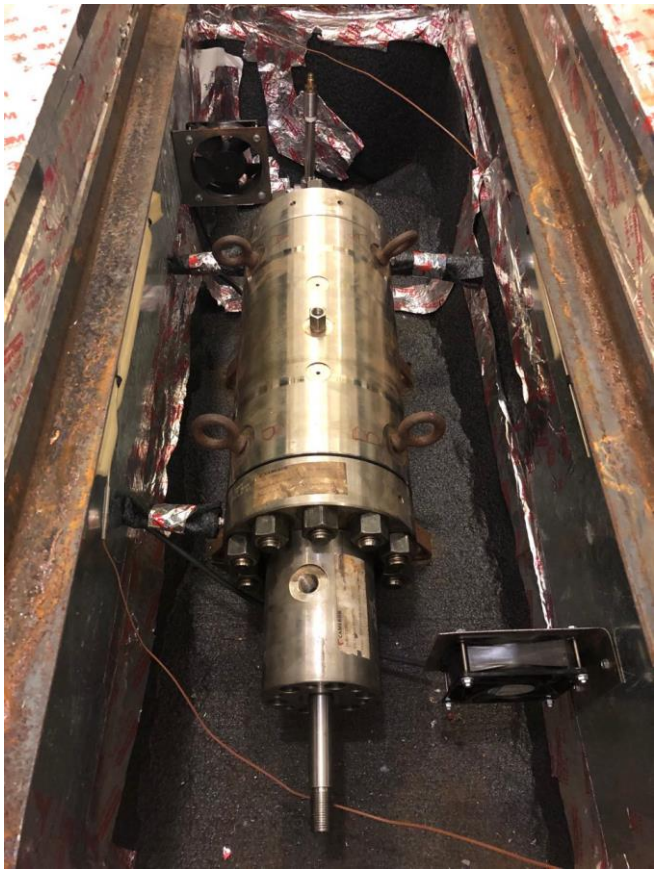


Figure 9 - Tubing of the inside of the box has been covered by insulation

For high temperature tests, four flexible ceramic heater bands were used to heat the test vessel. The heater bands were spaced along the length of the test vessel and they were adjusted using a PI controller on the side of the experimental setup. These heater bands can increase the temperature of the vessel's exterior, which can ultimately reach the center of the vessel within 2˚C of the target temperature. It is there, at the center of the vessel, that the barriers and pressurized test gas are located. It takes an hour to increase the temperature from room temperature to 100˚C, and 2 hours to reach 204˚C in the control volume.

For each experiment, data collection starts after the pressure is stabilized in the high-pressure side of the test vessel. To minimize gas leakage possibilities from the test setup, the use

of tubing connections was limited and instead, the use of long, curved, singular pieces of tube were used.

To pressurize the test vessel, the booster, seen in in the center of the photo in Figure 7 and seen on the schematic in Figure 10, is boosted using controls in the LabVIEW program. This booster increases the pressure in the high-pressure chamber of the experimental set up, which is labeled in Figure 5. The pressure within the high-pressure chamber is monitored by a regulator and the pressure value is displayed in the LabVIEW program. When the desired pressure within 20 psi is displayed in LabVIEW, the solenoid valve, which is labeled in the Figure 10 schematic, that allows the test gas into the chamber is shut and then the system is depressurized, indicated by the large pressure gauge shown in the middle of Figure 6.

Once the high pressure chamber is pressurized, the LabVIEW program logs the data given by the pressure transducers for sides A and B. The measure data is the amount of leakage across the plastic barriers from the high pressure chamber to the low pressure chamber. This is the data that has been analyzed.

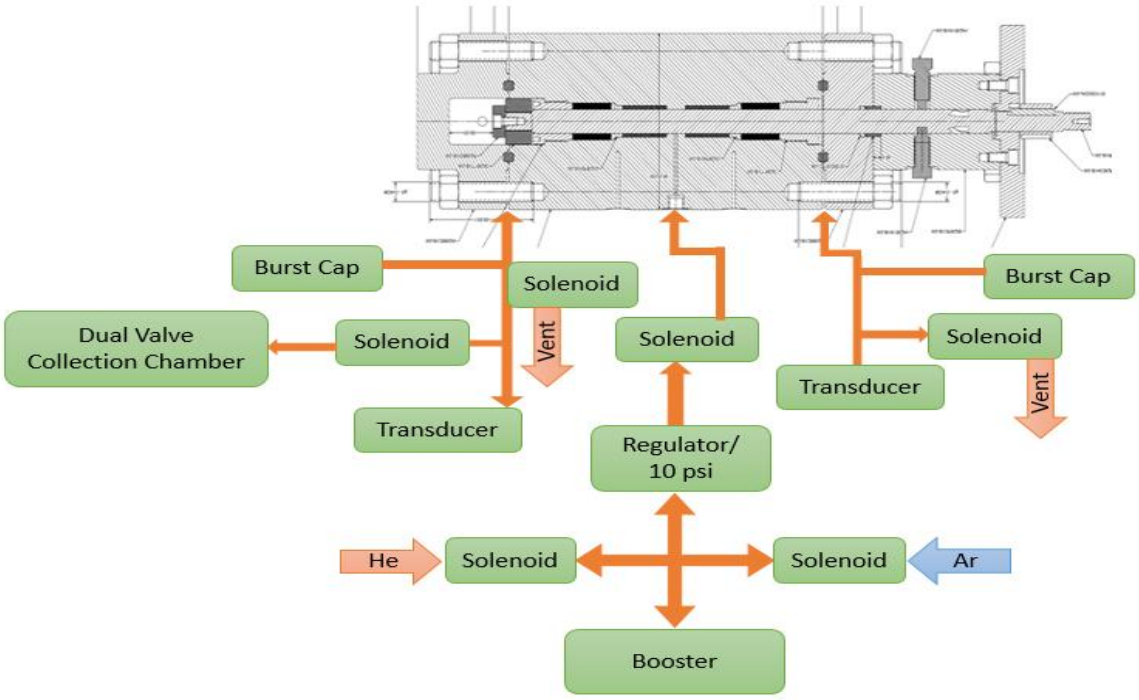


Figure 10 - Equipment schematic

Figure 10, courtesy of Brandon Mansur, shows a schematic of the system. It shows the test gas booster, which is what is used to increase the Argon or Helium pressures in the high pressure chamber, and how it is connected to the solenoid driven routing, and pressure regulator. The schematic also shows the burst caps, transducers, ventilation and safety valves which are all located on the exterior of the cryogenic chamber. To prevent any damage to the low-pressure side transducers, the burst caps and ventilation valves were installed for both sides A and B. In case of a barrier seal failure, causing the highly pressurized test gas to flood into the associated low pressure chamber, the burst caps and valves can vent the gas from the low-pressure sides. This protect the equipment measuring the leakage across the barriers, as they can be severely damaged by the high pressure influx.

3.2 Safety considerations

In order to conduct the experiments safely under high pressures, and with gases that can displace Oxygen, the LabVIEW equipment was placed in a separate bunker space with proper ventilation. This includes all of the equipment used for remotely controlling the experiments. This allowed the pressurization, depressurization, and test gas venting procedures to be controlled safely from the bunker space. The bunker spaces were separated by 50 cm thick concrete walls, making it safe against potential explosions. Cameras were also installed inside the test bunker to allow for continuous monitoring without having to enter the bunker during a high-pressure test.

The risk for overheating the testing vessel was a legitimate concern. So high-temperature experiments, the ceramic heater bands were setup to shut-off power by the LabVIEW program if overheating was observed. There was also a concern for unknowingly flooding the bunkers with Nitrogen, thus displacing Oxygen for anyone who enters the room before, during or after an experiment. To address this, Oxygen sensors were installed both inside the test bunker space and the remote-control bunker space. These sensors trigger alarms and an exhaust fan in the test bunker space when Oxygen levels drop below a set threshold.

3.3 High temperature experiment calculations

To increase the temperature of the vessel for high-temperature experiments, we used 4 heater bands controlled by a proportional-integral (PI) controller. The circuit is connected to a thermocouple and it measures the chamber temperature as feedback. Equations 1 and 2 are expressions for the PI control method:

$u(t)$ is fed into the system as the value of the controller output and manipulated variable input

$$e(t) = SP - PV$$

Equation 1 - PI Control Error Calculation

$$u(t) = u_{bias} + K_c e(t) + K_c \tau_1 \int \tau_0 e(t) dt$$

Equation 2 - PI Control Temperature Calculation

Generally, when the PI controller is switched from manual to automatic mode for the first time, the term u_{bias} as a constant set to the value of manipulated variable input $u(t)$. So, If the error is zero, there could be a "bumpless" transfer. In Equation 2, K_c is a controller gain, and τ_1 is an integral time constant. Both are tunings for the PI controller. The multiplier on the proportional error and the integral term is K_c. The higher the value of K_c makes the controller more aggressive at responding to the errors away from the set point. The setpoint (*SP*) is the target value and process variable (*PV*) is the measured value. PV can deviate from the desired value. Difference between the setpoint and the process variable is the error and defined as Equation 1.

3.4 Low temperature experiment cooling method

Liquid Nitrogen was used as the cooling agent. Liquid Nitrogen, with a boiling temperature of -195.8°C, is commonly used in cryogenic systems for its availability and cost effectiveness. Two commonly used methods with liquid Nitrogen cryogenic chambers are: one that uses a liquid Nitrogen pool at the bottom of the chamber, and one that applies a spray of liquid Nitrogen into the cryogenic chamber. The liquid Nitrogen pool method requires more Nitrogen, and it drops the temperature faster. To keep the Nitrogen in its liquid state and facilitate controlled evaporation, elaborate insulation tools and techniques are necessary. In

contrast, the Nitrogen spray method consumes less Nitrogen, and controlling temperature is much easier. The latter method was chosen for these experiments. To reduce the temperature, liquid Nitrogen is sprayed toward the top of the box. The two fans that circulated the inside air were very effective in minimizing temperature gradients, although some pooling did result at the bottom of the chamber.

3.5 Data collection process

In order to produce a data set that was reliable, appropriate for the scope, replicable, and consistent between experiments, the following procedure was executed for each experiment. After installing new seals, the first experiment was always a high-pressure run at 10,000 psi. This was done to have the same prestressed condition for all seals in all experiments. The seal stacks were replaced after they experience 204˚C for a full set of pressures with a single test gas. The experimental procedure details are shown, outlined by Abigail Hovorka and Milad Najafbeygi:

1) Recording data: Start the experiment by starting the recording of the data. The data gets recorded in a LabVIEW output file format. For any test, data recording starts automatically by running the LabVIEW program. File nomenclature was based on seal type on side A and side B, temperature, pressure, and number of experiments in that condition.
2) Setting temperature:
    a. Heating: connect the heater bands and wiring, connect thermocouple to controller feedback port, set the temperature on the PI controller, and check it with LabVIEW program.

b. Cooling: connect the housing valve to the liquid Nitrogen tank. Set the temperature to 4ºC higher than the target temperature in the LabVIEW program. Turn on the fans and stay on until the temperature stabilizes. Reduce the temperature gap and allow the temperature to stabilize at the target valve.

3) Purging: Open the valves and apply bottle pressure to the high and low-pressure chambers to remove air from the system.
4) Zero reading: To eliminate the effect of atmospheric pressure and temperature change in pressure measurement, before pressurizing the system, open the valves and take zero-point readings of the transducers.
5) Pressurizing system: Open the air regulator to start actuating gas booster for pressurizing. In cold experiments, set the pressure in 100 psi higher than the target pressure because of the effect of the cold temperature in gas.
6) Close: After setting the pressure on the high-pressure side, close the low-pressure side valves.
7) Depressurizing the system: For depressurizing the system, reduce the pressure on the high-pressure side and open the low-pressure valves.
8) Stop the experiment: Finish the test by stopping the recording in LabVIEW.

3.6 Data collection matrix

Table 4 shows the data collection matrix. During these experiments, leak rates were measured at low and high temperatures for a range of pressure between 600 psi to 10,000 psi. The main goal was to capture the most extreme conditions and to obtain repeatable leak rate data points. The 10,000 psi tests were used to not only observe high pressure behavior, but to also

stress the seal for the remaining experiments. Once the seals have been stressed, their condition is constant; if the experiments were run with the pressure increasing between the experiments, the stress on the seal would be different each time. Whereas if the seal is subjected to high pressure in the beginning, the stress that is caused by the high pressure is the same amount of stress the seals will exhibit for every experiment.

The 600 psi tests were used not only to observe leak rate behavior as low pressures, but they were also used as a standard part of procedure to check the energizing and sealing capabilities of the seals. Issues were observed routinely at this low pressure, which are most likely caused by the V-rings in the seal stack not functioning as expected. Table 4 indicates which pressure and temperature experiment combinations were conducted.

		Pressure (psi)							
		600	**720**	**1000**	**1500**	**2250**	**3750**	**6250**	**10,000**
Temperature (°C)	**-46**	✓				✓	✓	✓	✓
	-29	✓				✓	✓	✓	✓
	0	✓				✓	✓	✓	✓
	20	✓	✓	✓	✓	✓	✓	✓	✓
	121	✓	✓	✓	✓	✓	✓		✓
	204	✓					✓		✓

Table 4 - Data collection matrix

4. Modeling Approaches

The following proposed models and their methodology were produced by Abigail Hovorka, except for data processing assistance from Milad Najafbeygi which is specifically mentioned in chapter 4.1 Linear Modeling. There were two data sets from the experiments. A set of Helium leak rate data and a set of Argon leak rate data. Each of these sets contain the same combinations of pressure and temperature conditions. There were anywhere from 2 to 5 experiments completed for the different sets of conditions on the matrix (Table 4) that were

covered. This resulted in dozens of data sets and millions of data points. When these low pressure data points were graphed, they showed a linear behavior, see Appendix B. Thus, the goal with this this data set was to create a linear model for the purposes of predictive modeling. These model iterations are described in chapter 4.1 in depth. After creating this models, there was an interest in researching a more complex model with high data inclusivity. That model is described in chapter 4.2 in depth.

4.1 Linear Modeling

Following data collection, between 1 and 3 individual leak rate ratios of He/Ar have been calculated for the individual pressure and temperature combinations. The goal with this linear modeling was to find a He/Ar regression model that was representative of the data, using the experimental leak rate ratios. Two methods for data outlier elimination were tested. To pre-process the data, which is where Milad Najafbeygi aided, the low pressure data for both sides A and B of the test vessel were plotted using MATLAB, see Appendix B for some of those low pressure graphs. The slope from those graphs for each experiment were calculated and used as the leak rate for that specific gas under the specific experimental conditions. The average of the leak rates was calculated for each run number; multiple experiments were conducted with new sets of seals, so there were multiple first run experiments for each set of conditions. These averaged leak rates were then used to create the ratios. The averaged leak rate for the Helium was divided by the averaged leak rate for the Argon, thus creating the He/Ar leak rate ratio. Table 5 displays the leak rate ratios calculated for the different experimental conditions. The production of this Table and the necessary calculation involved were completed with the aid of Milad Najafbeygi.

Pressure (psi)	-46C side A	-46C side B	-29C side A	-29C side B	0C side A	0C side B	Room Temp Side B (B)	Room Temp Side A (W)	Room Temp (PTFE)	204 C - Side A	204 C - Side B	121 C - Side A	121 C - Side B
600		2.686		2.179		0.823	14.823	4.102	2.881	2.831	5.522	66.982	2.386
750							24.073	4.807					
1000							17.956	4.158	2.839				
1500							25.095	7.596					2.709
2250	12.928	21.127	8.143	19.957			14.348	13.176	2.763				
3750	34.621		18.718		22.790		9.787	16.491	4.263	3.750	2.929		3.956
6250	64.262		60.181		39.553		26.487	9.364	7.876			148.471	
10000							16.500		18.223	8.571			54.489

Table 5 - Data Matrix for Linear Modeling

This set of ratios with the different experimental combinations is the data set used to create a regression model for He/Ar leak rates. The linear models include coefficients calculated from the experimental He/Ar ratios to produce regression models. With these regression models, the probable He/Ar leak rate ratio for any temperature and pressure can be determined. Using this model, and an Ar/CH_4 multiplier, a He/CH_4 model can be determined. This multiplier has been determined from third-party select experiments as a conservative Ar/CH_4 leak rate ratio. Using the regression model and ISO industry standards for allowable Methane leak rate concentrations, an equivalent Argon leak rate and Helium leak rate can be found. Due to the molecular behavior of Argon, it is found to model Methane leak rates more closely than with Helium, while providing a conservative estimate since Argon leaks faster than Methane.

4.1.1 Method 1 with data bounds from IQR

This method used for eliminating outliers involved an interquartile range, denoted IQR, to summarize the variability (NIST/SEMATECH e-Handbook of Statistical Methods, 2012). The interquartile range is the difference between the first and third quartiles of the data (NIST/SEMATECH e-Handbook of Statistical Methods, 2012). The first quartile, denoted Q1, is the value in the data set that contains 25% of the data below the median and the third quartile, denoted Q3, is the value in the data set that contains 25% of the data above the median (NIST/SEMATECH e-Handbook of Statistical Methods, 2012). This does not necessarily mean that the value for the first quartile is an equal magnitude of distance from the median as is the third quartile, it means that the amount of data points between the median and the specific quartile are equal. This is represented in Figure 11. The IQR is the area of the data spread that encompasses the bulk of the data around the median.

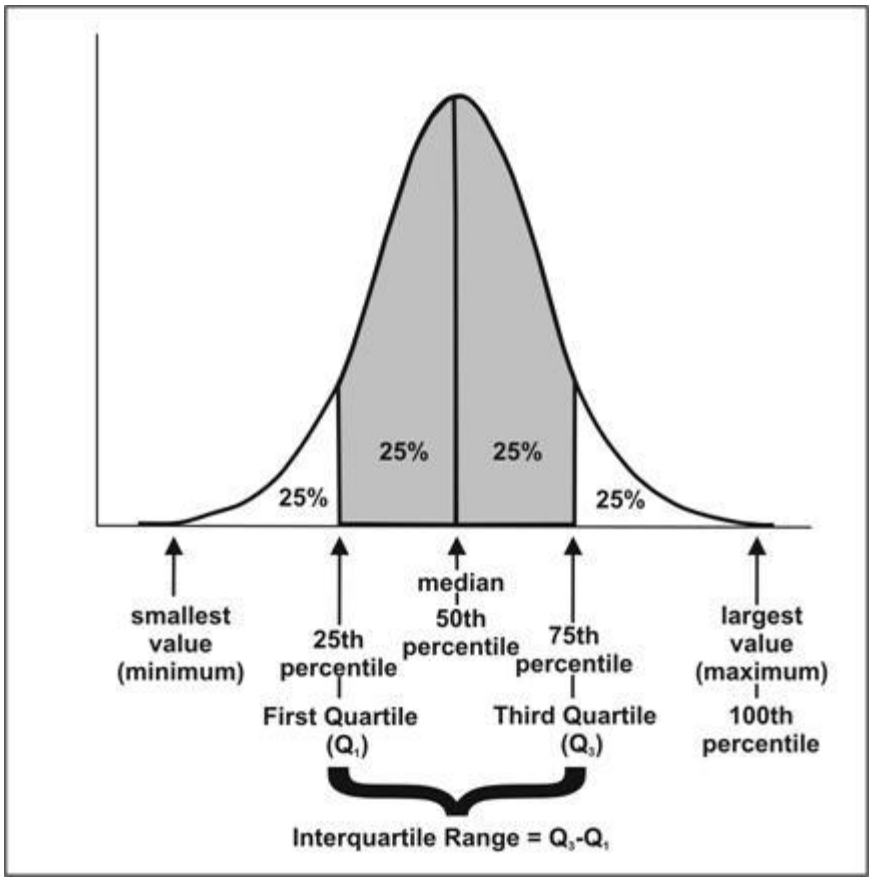


Figure 11 – The Middle Half of the Observations in a Frequency Distribution Lie within the Interquartile Range

(Centers for Disease Control and Prevention, 2012)

The lower-bound for acceptable data is represented by Q1 and the upper-bound is represented by Q3. These bounds are $<0.75\sigma\pm$ on a normal probability plot, so this method has been named Method 1. Any data point that falls outside of these bounds is considered an outlier and is removed. The values for the bounds are calculated to be:

Q1 (lower bound)	3.904
Q3 (upper bound)	21.543

Table 6 - Method 1 bounds

Therefore, any He/Ar ratios below 3.904 or any data above 21.543 are considered to be outliers. When this method was applied it removed nearly 50% of the data set. The final data set for Method 1 is displayed in Table 7.

T	P	Ratios
-46	2250	21.127
-46	2250	12.928
-29	2250	8.143
-29	2250	19.957
-29	3750	18.718
0	2250	11.496
121	3750	3.956
204	600	5.522
204	10000	8.571
25	600	14.823
25	600	4.102
25	750	4.807
25	1000	17.956
25	1000	4.158
25	1500	7.596
25	2250	14.348
25	2250	13.176
25	3750	4.263
25	3750	9.787
25	3750	16.491
25	6250	7.876
25	6250	9.364
25	10000	18.223
25	10000	16.500

Table 7 - Method 1 Filtered Data Set

After removing data points from the set, a regression analysis was performed with the remaining data. The regression analysis produced residuals, residual plots, statistical inferences, intercept coefficients and variable coefficients for the regression model. Before moving forward with these coefficients to build the regression model, an analysis on the residuals was required to confirm the validity of the coefficients produced. A way to do this is to test for normality among the residuals. The Shapiro-Wilk test for normality is a useful tool for determining a data set's

normality. This test calculates a W-test statistic from the data set. In this case it is the set of residuals (NIST/SEMATECH e-Handbook of Statistical Methods, 2012). The W-test statistic is calculated as follows:

$$W = \frac{\left(\sum_{i=1}^{n} a_i x_{(i)}\right)^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

Equation 3 - Calculation for W-test statistic

Where x_i represents the ordered sample values ($x_1$is the smallest data point) and the $a_i$values are constants calculated using the mean, variances and covariances of the ordered statistics using the sample size *n* from a normal distribution. With this test statistic, a p-value can be determined from data Tables (NIST/SEMATECH e-Handbook of Statistical Methods, 2012). For simplicity, the statistical software R has been used to produce this test statistic and p-value. Using a Shapiro-Wilk test on the residuals, it can be determined that there is normality among the residuals with 90% confidence (p-value=0.162).

Another way to assess model validity is to look at the Significance F-value in the analysis of variance (ANOVA) Table produced by the regression analysis. In an ANOVA Table, the experimental response measurements are separated into components that correspond to different sources of variation (NIST/SEMATECH e-Handbook of Statistical Methods, 2012). An ANOVA Table includes five columns of data: degrees of freedom (df), sum of squares (SS), mean squares (MS), F-test statistic (F), and significance F value (significance F). Degrees of freedom is the number of data points that can be assigned to a particular distribution. The sum of squares is the squared sum of each data point's variation from the mean. This allows the computation of variance displayed in the ANOVA Table. The mean squares are the sum of squares divided by their respective degrees of freedom. Relating these variances to the number of data points

provides an understanding on the population variance. The F-test statistic is the mean square of the regression divided by the mean square of the residuals. An F-test statistic is a test statistic on the F-distribution. Finding this value produces a ratio of explainable and unexplainable variance. With this F-test statistic and the degrees of freedom for the data set, the significance F-value is produced and can be analyzed like a normal p-value (Simon Fraser University, 2011).

ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	2	219.819	109.910	4.514	0.023
Residual	21	511.286	24.347		
Total	23	731.105			

Table 8 - Method 1 ANOVA Table

In this model, the Significance F-value is less than 0.05, which shows statistical validity.

Lastly, one of the more well-known methods for validating model is R^2, which is a "widely used goodness-of-fit measure whose usefulness and limitations are more or less known to the applied researcher" (Cameron & Windmeijer, 1997). It is also described as the variance of the predicted values divided by the variance of the data (Gelman, Goodrich, Gabry, & Vehtari, 2019). The R^2 for the model produced by Method 1 is 0.301 with an adjusted R^2 of 0.234, It is generally better to look at the adjusted R^2 value as it is an unbiased estimator that makes corrections for the sample size and the number of variables (Nau, 2019). To many, these R^2 values would be considered "too low" since they are below the classic 0.7 recommendation (Grace-Martin, 2020). However this is not always the case because an R^2 value, no matter what size, is meant to explain the explained variance in a model for a given data set (Nau, 2019). It is still showing that there is some significant effect on the output values by the input variables, in terms of this linear model; these variables showcase a small, but reliable relationship with the

leak rate ratio values (Grace-Martin, 2020). Considering there are other metrics to help validate this model, it is still important to assess the predictions it produces.

Regression Statistics	
Multiple R	0.548
R Square	0.301
Adjusted R Square	0.234
Standard Error	4.934
Observations	24.000

Table 9 - Regression Statistics for Method 1

With this in mind, and considering the model has been validated in two other ways, this model can proceed in making predictions.

Now that the model has been validated, the coefficients produced from the regression analysis can be used to produce a regression model for He/Ar leak rate ratios.

These are the upper 95% coefficients:

Upper 95%
14.0884
-0.0125
0.0013

Table 10 - Method 1 regression values

When using these coefficients to calculate He/Ar leak rate ratios for the same inputs as the experimental data, the predicted ratios plotted alongside the experimental ratios can be seen in Figures 12 and 13. The same general trend is followed with the predicted ratios.

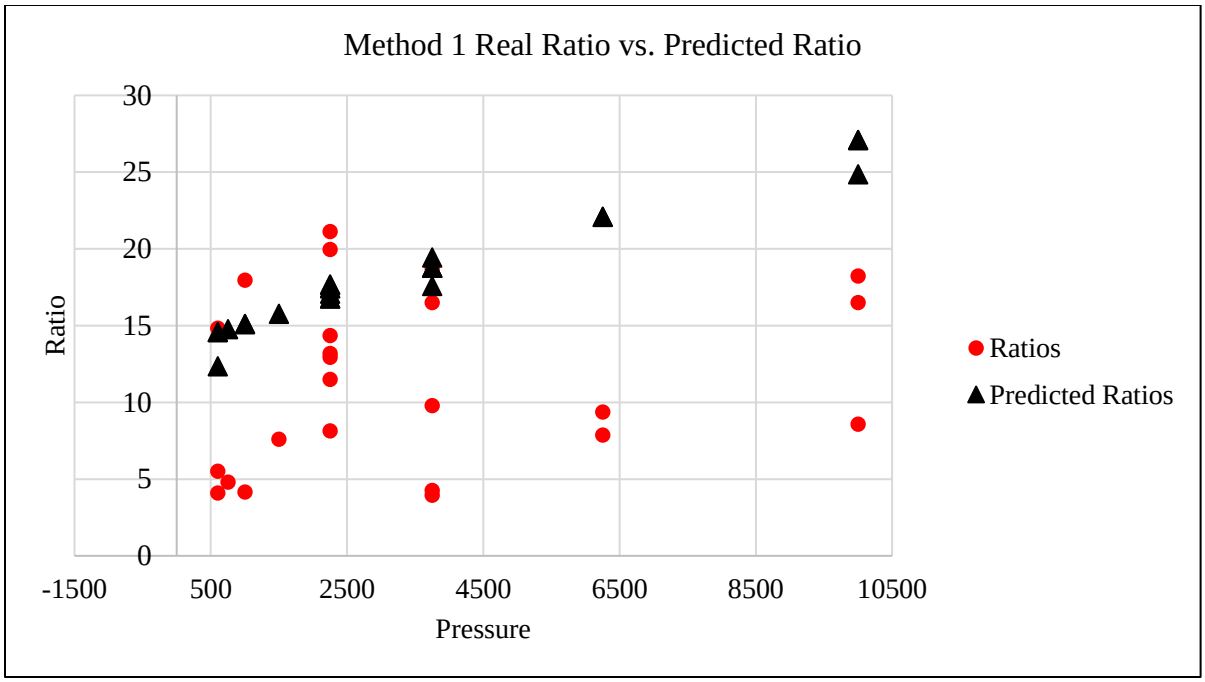


Figure 12 - Method 1 Experimental Ratios vs. Predicted Ratios (plotted by pressure)

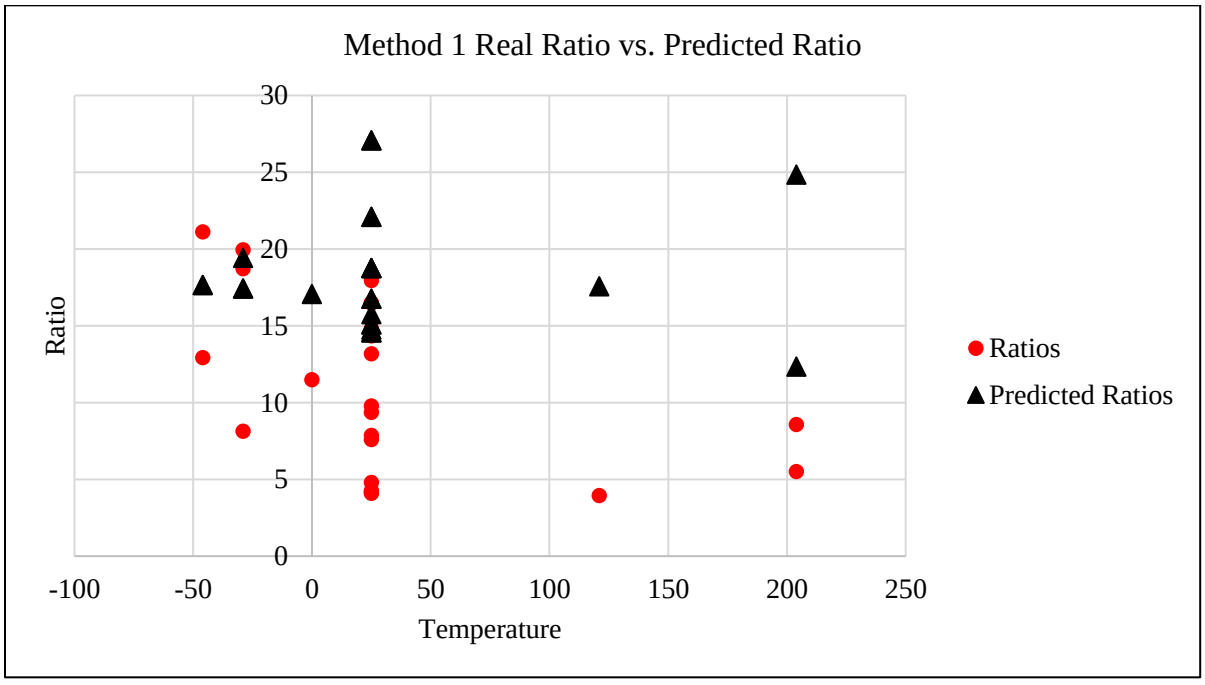


Figure 13 - Method 1 Experimental Ratios vs. Predicted Ratios (plotted by temperature)

In Figures 14 and 15, Method 1 is plotted to provide a visual for values predicted between 600 psi and 10,000 psi.

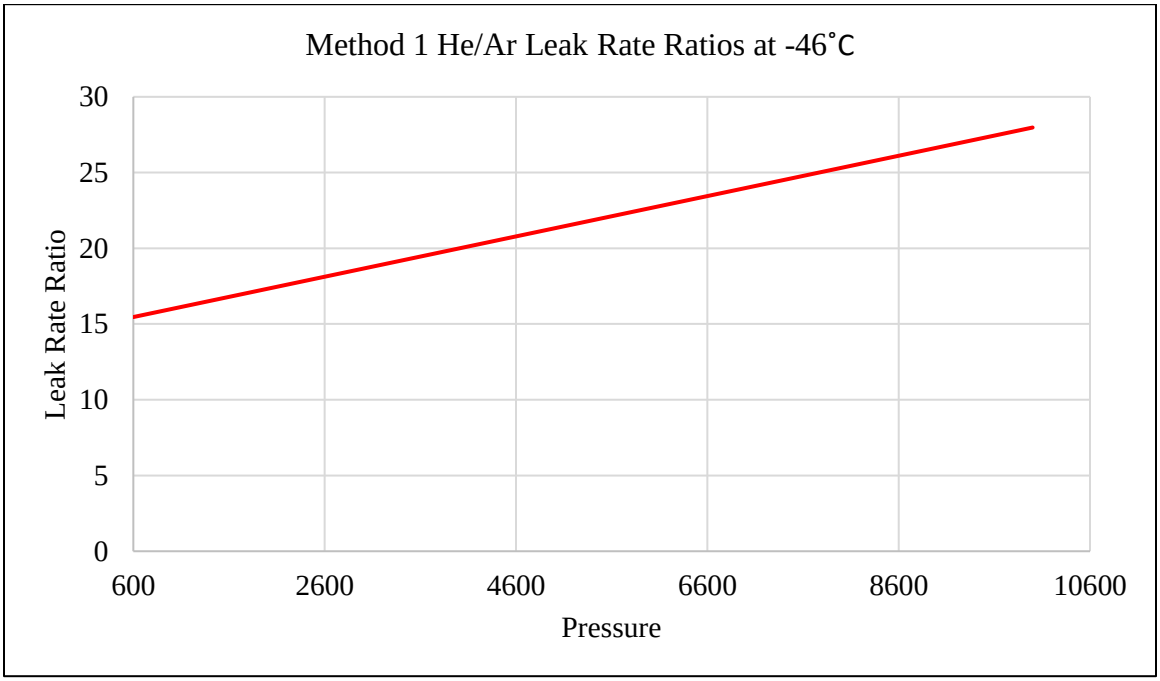


Figure 14 - Graphical representation of Method 1's He/Ar regression model for one temperature (-46˚C)

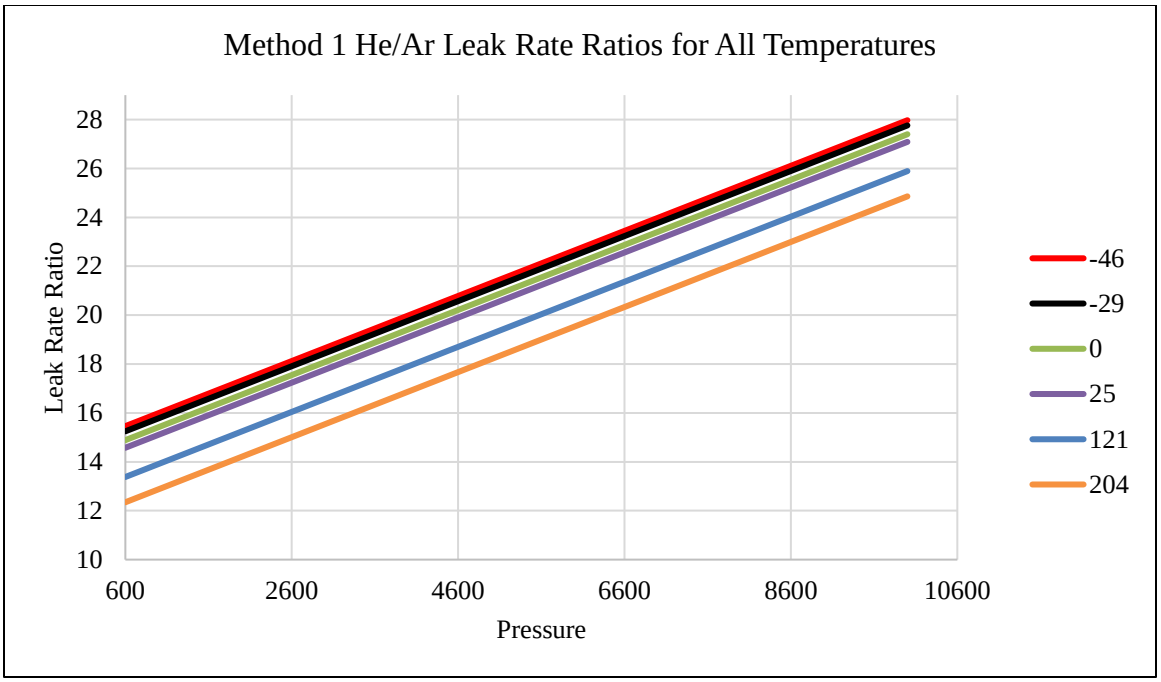


Figure 15 - Graphical representation of Method 1's He/Ar leak rate ratio regression model for various temperatures

The regression analysis provides multiple values and coefficients for the regression model: intercept coefficient, variable coefficients, and coefficient values at the upper and lower bounds of a 95% confidence interval. The first set of coefficients consist of the least squares estimate for the regression model variable coefficients and the regression model intercept. The "least squares regression" technique is used to create values for a best fit line for a set of data. This, theoretically, is as close to the true values for a regression model to which the analysis tool can be. The upper and lower 95% values show what the coefficients are at the upper and lower bounds of the 95% confidence interval for the best fit line values. The 95% confidence interval is a range of values that we can be 95% certain contains the true regression model values. The upper 95% values were selected to build the regression model shown below. This predicts lower and more conservative Ar leakage threshold to meet a given Helium leak rate. If the intent is to

define equivalent Helium leak rate to meet a given CH_4 leakage threshold, the lower 95% values for the regression model is likely a better and more conservative option. The resulting He/Ar leak rate ratio model is (X_1 is temperature in ˚C and X_2 is pressure in psi):

$$f(P,T) = 14.0884 - 0.0125X_1 + 0.0013X_2$$

Equation 4 - Method 1 He/Ar Leak Rate Ratio Multivariate Linear Regression Model

The He/CH_4 leak rate ratio model is shown below, estimated by multiplying the He/Ar leak rate model (Equation 12) by 1.5. The selection this factor is explained in chapter 4.2.

$$f(P,T) = 21.1326 - 0.0187X_1 + 0.0020X_2$$

Equation 5 - He/CH_4 Leak Rate Ratio Multivariate Linear Regression Model

4.1.2 Method 2 with Tukey Fences

This method for eliminating outliers used an interquartile range, similar to Method 1, but the upper and lower bounds were found differently. In this case, the Tukey Fences were used.

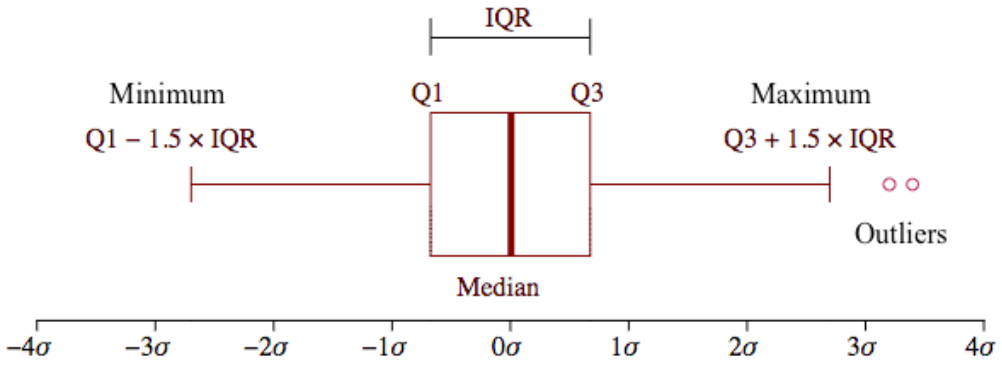


Figure 16 – Box and Whisker Plot Example

(Zheng, 2019)

An IQR is useful to eliminate outliers because it identifies the behavior of the data's overall spread in the middle half of the data set. The IQR narrows the focus to the consistent and reliable

data points. In this case, the term “reliable” means a set of data that could be repeated if performed again, hence providing an accurate representation of the true behavior of the experiment’s dependent variables. Once there is a measure of the data’s spread, it can be used to identify how far that reliability can reach in the data set. The mathematical method to determine this new reliable spread was founded by John Tukey, a famous statistician. This method creates a lower bound and an upper bound, referred to as “Tukey Fences”, for the data considered to not be an outlier. Equation 6 shows the calculation for the lower bound and Equation 7 shows the calculation for the upper bound (NIST/SEMATECH e-Handbook of Statistical Methods, 2012).

$$Q_1 - 1.5 \times IQR = LB$$

Equation 6 - Tukey Fence Lower Bound Calculation

$$Q_3 + 1.5 \times IQR = UB$$

Equation 7 - Tukey Fence Upper Bound Calculation

The significance of the 1.5 coefficient is that it extends the data set spread toward the extremes of the probability density function, denoted as PDF. Any areas in the PDF that are outside of $>3\sigma\pm$ are areas that are considered to be “extremes” in the data spread. Hence, the 1.5 value is the standard coefficient used for finding IQR outliers (Simon Fraser University, 2011). When the theory was applied to this ratio data set, the results were as follows:

Q1	3.904
Q3	21.543
IQR	17.639
LB	-22.554
UP	48.000

Table 11 - Method 2 bounds

Therefore, any He/Ar ratios below -22.554 or any data above 48.000 are considered to be statistical outliers. This eliminated five ratios. The data set after outlier elimination for Method 2 is included in Table 12.

T	P	
-46	600	2.686
-46	2250	21.127
-46	2250	12.928
-46	3750	34.621
-29	600	2.179
-29	2250	8.143
-29	2250	19.957
-29	3750	18.718
0	600	0.8233
0	2250	11.496
0	3750	22.790
0	6250	39.553
121	1500	2.709
121	3750	3.956
121	600	2.386
204	600	2.831
204	600	5.522
204	3750	3.750
204	3750	2.929
204	10000	8.571
25	600	2.881
25	600	14.823
25	600	4.102
25	750	24.074
25	750	4.807
25	1000	2.839
25	1000	17.956
25	1000	4.158
25	1500	25.095
25	1500	7.596
25	2250	2.763
25	2250	14.348
25	2250	13.176
25	3750	4.263
25	3750	9.787
25	3750	16.491
25	6250	7.876
25	6250	26.487
25	6250	9.364
25	10000	18.223
25	10000	16.500

Table 12 - Method 2 Filtered Data

After removing five data points from the set, a regression analysis was performed with the remaining data. Just like Method 1, the regression analysis produced residuals, residual plots, statistical inferences, intercept coefficients and variable coefficients for the regression model. Before moving forward with these coefficients to build the regression model, an analysis on the residuals was required to confirm the validity of the coefficients produced. A Shapiro-Wilk test was used again to check for normality among the residuals. It can be determined that there is normality among the residuals with 99% confidence (p-value=0.0167). Model validity is integral, so the Significance F-value has also been found for this model to assess validity.

ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	2	1127.541	563.770	8.593	0.001
Residual	38	2493.180	65.610		
Total	40	3620.721			

Table 13 - Method 2 ANOVA Table

In this model, the F-value is less than 0.05, which shows statistical validity. Like with Method 1, the final metric to observe is the adjusted R^2 value for the regression model using the Method 2 filtered data set. There is something similar happening with Method 2. The R^2 value is 0.311 and the associated adjusted R^2 value is 0.275, again bringing up the concern with lower R^2 values. The same idea goes for this situation, these metrics are still indicating that there is a small but noticeable relationship between these variables and the leak rate ratios being predicted (Grace-Martin, 2020). Although, it is worth noting that the Method 2 adjusted R^2 is greater than that of Method 1, indicating that the relationship between pressure, temperature and the leak rate ratios is more reliable in Method 2.

Regression Statistics	
Multiple R	0.558
R Square	0.311
Adjusted R Square	0.275
Standard Error	8.100
Observations	41.000

Table 14 - Regression Statistics for Method 2

Now that the model has been validated in multiple ways, the coefficients produced from the regression analysis can be used to produce a regression model for He/Ar leak rate ratios. These are the upper 95% coefficients:

Upper 95%
13.758
-0.022
0.002

Table 15 - Method 2 regression model coefficients

When using these coefficients to calculate He/Ar leak rate ratios for the same inputs as the experimental data, the predicted ratios plotted alongside the experimental ratios can be seen in Figures 17 and 18. The same general trend is followed with the predicted ratios.

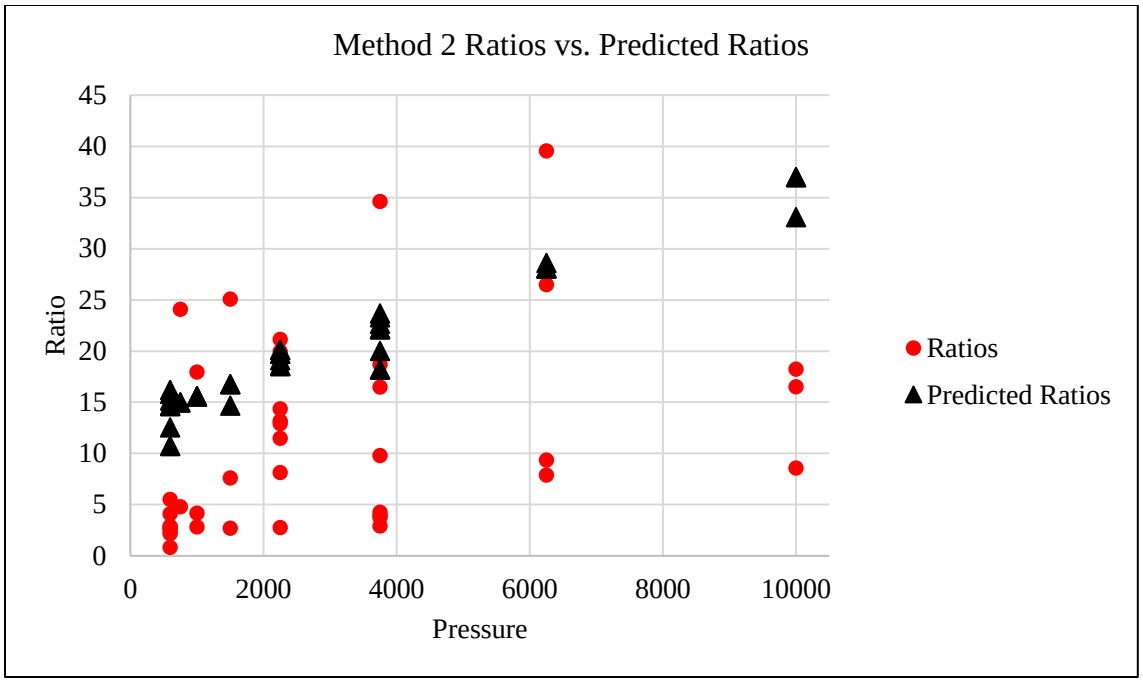


Figure 17 - Method 2 Experimental Ratios vs. Predicted Ratios (plotted by pressure)

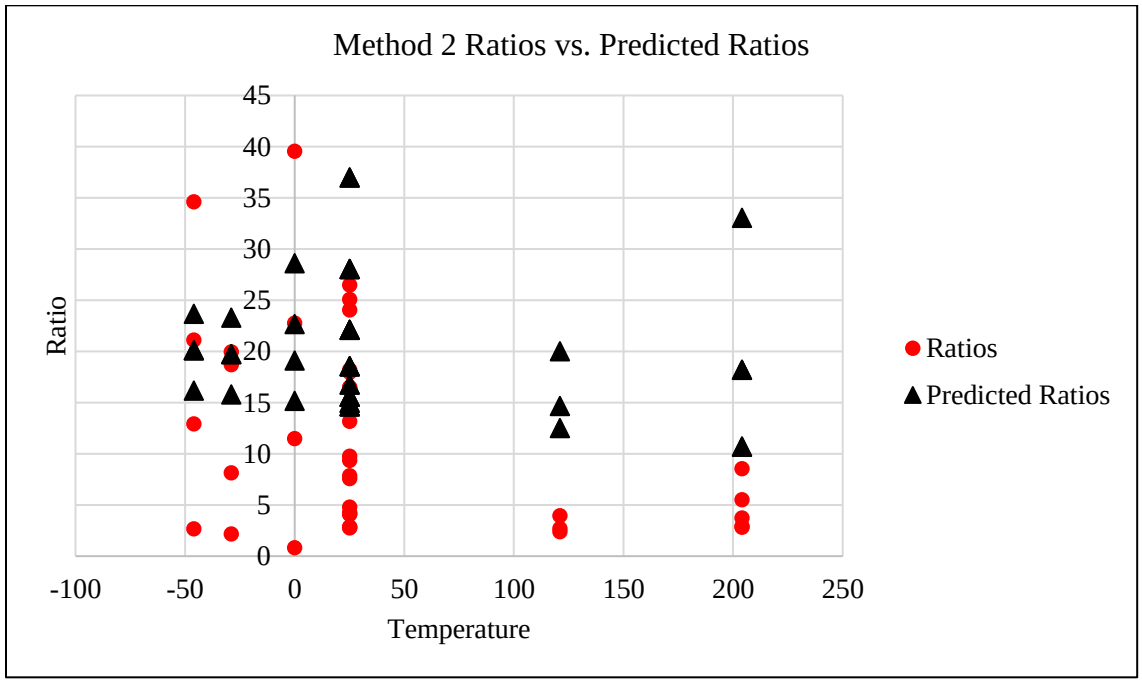


Figure 18 - Method 2 Experimental Ratios vs. Predicted Ratios (plotted by temperature)

In Figures 19 and 20, Method 1 is plotted to provide a visual for values predicted between 600 psi and 10,000 psi.

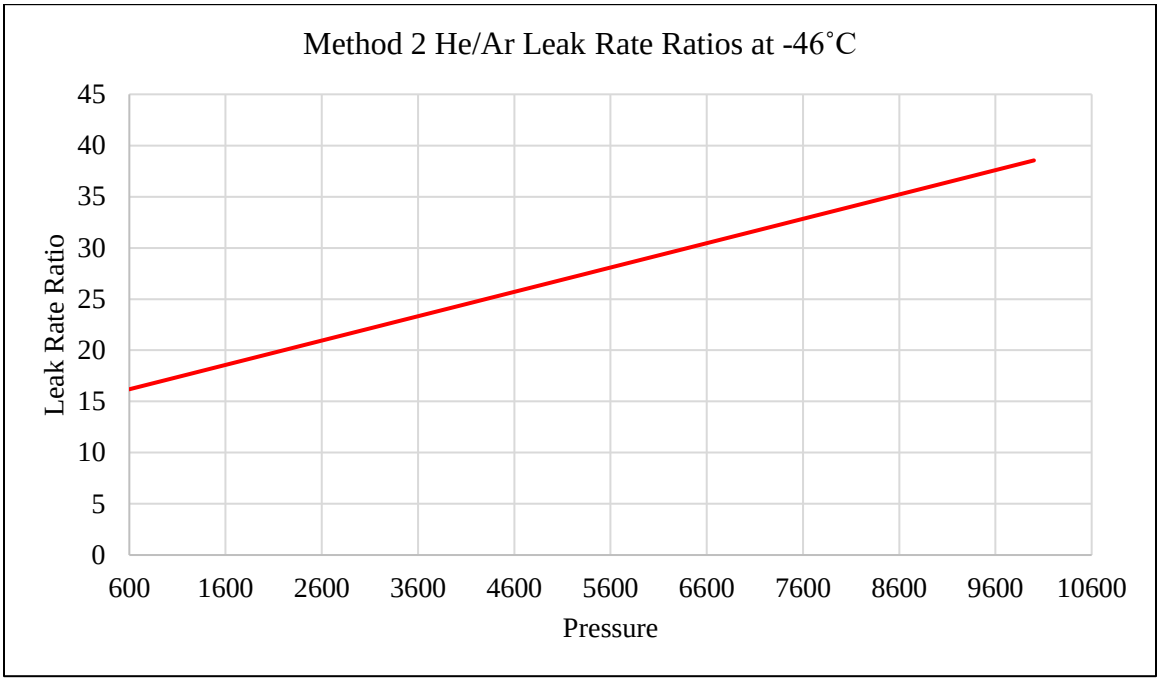


Figure 19 - Graphical representation of Method 2's He/Ar leak rate ratio regression model for one temperature (-46˚C)

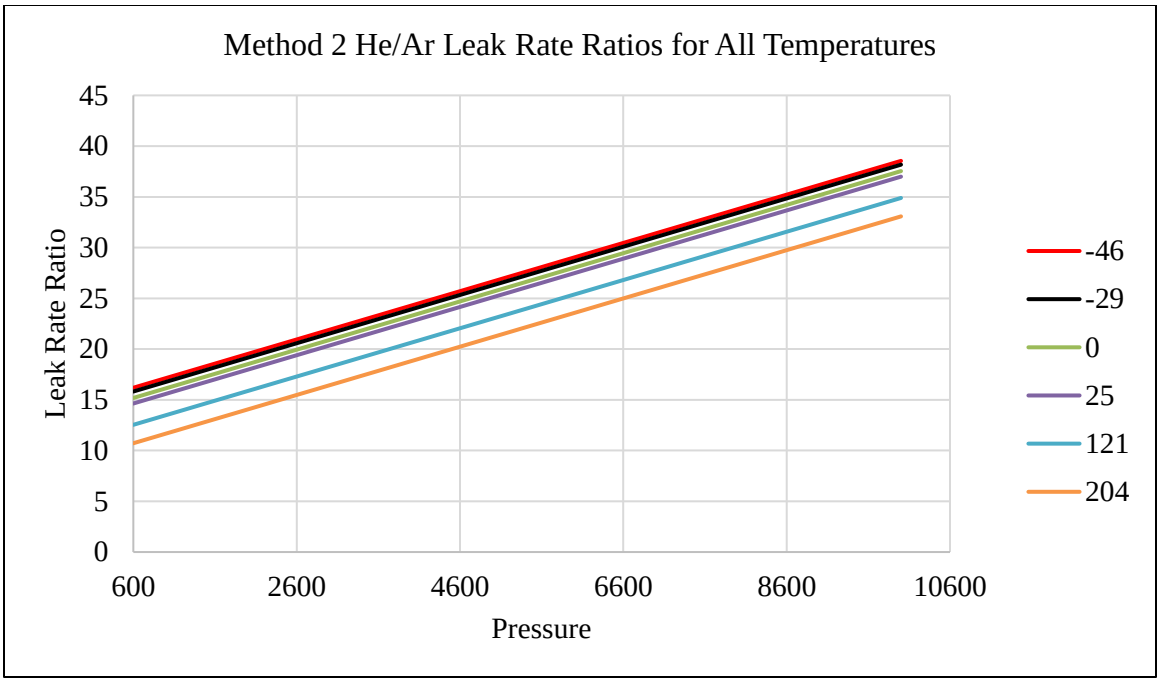


Figure 20 - Graphical representation of Method 2's He/Ar leak rate ratio regression model for various temperatures

For the same logic used for Method 1, the upper 95% values are used for the Model 2 regression model. The resulting He/Ar leak rate ratio model is (X_1 is temperature in deg C and X_2 is pressure in psi):

$$f(P,T) = 13.7577 - 0.0219X_1 + 0.0024X_2$$

Equation 8 - Method 2 He/Ar Leak Rate Ratio Multivariate Linear Regression Model

The He/CH_4 leak rate ratio model is shown below, estimated by multiplying the He/Ar leak rate model (Equation 16) by 1.5. The selection this factor is explained in chapter 4.2.

$$f(P,T) = 20.6365 - 0.0328X_1 + 0.0036X_2$$

Equation 9 - Method 2 He/CH_4 Leak Rate Ratio Multivariate Linear Regression Model

4.2 Random Forest Modeling

For this method, the goal was to use more of the available data than in the previous methods to capture more of the data's behavior. The previous two methods had a shorter original data set including only 1 to 3 leak rate ratios for each set of conditions in the data collection matrix. This next method uses the same raw data sets that were collected from the experimental set up. However, the pre-processing method was quite different. Rather than just producing a few leak rate ratios per set of conditions as displayed previously in Table 5, for each run for each set of conditions, 15 leak rates were produced and thus 15 leak rate ratios per run per set of conditions were produced.

The data collected included millions of data points. The most important task was to process the raw data in such a way that the data could be used to create a model that would make predictions based off of temperature and pressure inputs. To do this, there needed to be a series of leak rate changes associated with each set of temperature and pressure experimental conditions. And with that larger data set, a Helium/Argon leak rate ratio model tracking the ratio change could be made. This model would then be able to make appropriate ratio predictions for any set of temperature and pressure inputs within the experimental bounds.

4.2.1 Preliminary Data Processing

The first step was cleaning the data. In each of the experiments, there was a clear pressurization period before the leakage occurred, a clear leakage period and a clear depressurization period after the leakage occurred from the raw data shown in Figure 21.

Date & Time	Time (Seconds)	High Pressure	LP Side A	LP Side B	Temp Side A	Temp Side B	Box Thermocouple 1	Box Thermocouple 2	Box Thermocouple 3	Box Thermocouple 4	Box Temp Average	Ambient
08/29/2019 01:10:56.174 PM	0	279.8651435	1.010916893	0.035044779	-0.530682132	0.079467377	-23.76552014	-28.94088723	-17.00441367	-15.37551479	-21.27158396	23.4705703
08/29/2019 01:10:57.176 PM	1	279.6833338	1.010000945	0.034502655	-0.597968876	0.137546315	-23.74341433	-28.81953532	-16.81275429	-15.32372204	-21.17485649	23.48780627
08/29/2019 01:10:58.175 PM	2	279.5395815	1.008880672	0.035065142	-0.655236382	0.069167327	-23.75768445	-28.7763735	-16.82273794	-15.31511284	-21.16797718	23.46299549
08/29/2019 01:10:59.175 PM	3	279.4013888	1.008753755	0.035511049	-0.571755272	0.032058081	-23.62202845	-28.78804101	-16.92145913	-15.29745511	-21.15724592	23.51021613
08/29/2019 01:11:00.176 PM	4	279.4987253	1.00905537	0.035742163	-0.535260363	-0.018099687	-23.54939474	-28.64579963	-16.82166141	-15.23819434	-21.06376253	23.43703813
08/29/2019 01:11:01.175 PM	5	279.6559526	1.009398344	0.035555079	-0.563720149	-0.007772022	-23.51196076	-28.51372244	-16.71089385	-15.15256818	-20.97228631	23.45572006
08/29/2019 01:11:02.176 PM	6	279.8386021	1.00992114	0.034069757	-0.536700043	-0.009911233	-23.48174149	-28.41301333	-16.63689122	-15.11599352	-20.91190989	23.35701302
08/29/2019 01:11:03.175 PM	7	280.2894051	1.009832166	0.034079616	-0.555063298	0.046786102	-23.51684005	-28.39243502	-16.62310098	-15.01656848	-20.88723613	23.35232107
08/29/2019 01:11:04.176 PM	8	279.7973026	1.011289316	0.034177822	-0.621169537	0.032590347	-23.47406119	-28.43188756	-16.56766531	-15.00172953	-20.8688359	23.46771012
08/29/2019 01:11:05.175 PM	9	279.5068499	1.011229363	0.035848549	-0.644976365	0.028344221	-23.39517129	-28.35067178	-16.53420927	-14.97050689	-20.81263981	23.49494385
08/29/2019 01:11:06.175 PM	10	279.4945201	1.012324408	0.036754593	-0.57122874	0.043926481	-23.3574491	-28.25676422	-16.42597713	-14.94461785	-20.74620208	23.49693324
08/29/2019 01:11:07.175 PM	11	279.3626157	1.011374819	0.035082559	-0.505295756	0.105376117	-23.28604295	-28.14731125	-16.38534031	-14.86203252	-20.67018176	23.48250959
08/29/2019 01:11:08.175 PM	12	200.7214891	1.010126829	0.034757072	-0.500687388	0.124516471	-23.34302854	-28.01205362	-16.3844795	-14.81339145	-20.63823828	23.55406315
08/29/2019 01:11:09.177 PM	13	88.21646739	1.009569658	0.036877101	-0.529947102	0.10926073	-23.24589368	-27.93484483	-16.31897942	-14.67510755	-20.54370637	23.53399453
08/29/2019 01:11:10.176 PM	14	-1.718499374	0.607379464	0.036485211	-0.524536461	0.024311182	-23.07172879	-27.87112114	-16.48945193	-14.61660517	-20.51222676	23.51203439
08/29/2019 01:11:11.176 PM	15	-1.299964077	0.203011061	0.025570097	-0.618770104	-0.014489133	-23.03148463	-27.88269625	-16.5242842	-14.64562026	-20.52102134	23.48151629
08/29/2019 01:11:12.178 PM	16	-1.236245828	0.002140399	0.013191998	-0.627995426	-0.020678333	-23.02307091	-27.84164323	-16.44104592	-14.59785169	-20.47590294	23.46126732
08/29/2019 01:11:13.177 PM	17	-1.254239001	0.002917106	0.006099665	-0.60833489	0.010387971	-22.95217315	-27.83063537	-16.18193033	-14.53485391	-20.37489819	23.45251816
08/29/2019 01:11:14.177 PM	18	-0.93609346	0.00284719	0.005363203	-0.605193285	-0.000584441	-22.94658102	-27.66067025	-16.05286217	-14.47041592	-20.28263234	23.55836562
08/29/2019 01:11:15.177 PM	19	-0.726820501	0.001645409	0.005397968	-0.566251129	0.064968387	-22.92443311	-27.54302316	-15.98920255	-14.44161202	-20.22456771	23.57780288
08/29/2019 01:11:16.177 PM	20	-1.177815143	0.00188586	0.005099301	-0.531610562	0.077049775	-22.88001275	-27.51020785	-16.09818378	-14.39609641	-20.2211252	23.52159464
08/29/2019 01:11:17.178 PM	21	-1.545769486	0.001826801	0.004974793	-0.556921252	0.090108799	-22.90406582	-27.47401419	-16.13379107	-14.3225696	-20.20861017	23.57611949
08/29/2019 01:11:18.177 PM	22	-1.827078042	0.00195322	0.00444205	-0.554533422	0.1039536	-22.76372987	-27.49599599	-16.03197875	-14.24871792	-20.13510563	23.48941739
08/29/2019 01:11:19.178 PM	23	-1.600019523	0.001162493	0.004306543	-0.582063856	0.089929385	-22.65861835	-27.37803249	-15.9406494	-14.17564213	-20.03823559	23.50068496
08/29/2019 01:11:20.177 PM	24	-1.60434377	0.00234093	0.005031603	-0.62716512	0.087644047	-22.60913129	-27.27736839	-15.98162952	-14.20286706	-20.01774907	23.53350063
08/29/2019 01:11:21.177 PM	25	-1.961461941	0.002734023	0.006520715	-0.644784816	0.002381509	-22.60926014	-27.1895931	-15.92681329	-14.155186	-19.97021313	23.55150636
08/29/2019 01:11:22.178 PM	26	-2.283349403	0.002067298	0.004816224	-0.716474161	0.023784117	-22.58582322	-27.10429926	-15.83620121	-14.11240772	-19.90968285	23.50686514
08/29/2019 01:11:23.178 PM	27	-1.118258545	0.002051159	0.004553636	-0.697101994	-0.024743828	-22.56100073	-27.04853047	-15.88222565	-14.05166417	-19.88585526	23.51478804
08/29/2019 01:11:24.178 PM	28	55.42662722	0.002969551	0.005941294	-0.664871671	-0.040907472	-22.54457148	-26.9939955	-15.7766268	-13.97142057	-19.82165359	23.51448981
08/29/2019 01:11:25.178 PM	29	176.3859585	0.003840216	0.00725691	-0.648005855	-0.015664869	-22.48155735	-26.87887615	-15.75708194	-13.9150498	-19.75814131	23.43367518
08/29/2019 01:11:26.178 PM	30	316.3771106	0.00657866	0.006114581	-0.583463887	-0.044233792	-22.38913756	-26.71548928	-15.6976925	-13.85368852	-19.66400197	23.47533016
08/29/2019 01:11:27.178 PM	31	393.5304703	0.008529901	0.004801817	-0.607715167	0.042129148	-22.32423259	-26.70653362	-15.72332279	-13.81347695	-19.64189149	23.53807887
08/29/2019 01:11:28.178 PM	32	425.7376726	0.011378833	0.003878982	-0.579352719	-0.030119182	-22.17227907	-26.70769749	-15.61292432	-13.80237175	-19.57381816	23.57579458
08/29/2019 01:11:29.178 PM	33	503.4186645	0.011646738	0.00386763	-0.612827689	0.036871741	-22.09219053	-26.69865481	-15.53435897	-13.78993103	-19.52878383	23.54370914
08/29/2019 01:11:30.179 PM	34	579.7126031	0.01269547	0.00527112	-0.601059616	0.016558577	-22.09715836	-26.58558117	-15.47242485	-13.68885399	-19.46100459	23.59936248
08/29/2019 01:11:31.178 PM	35	614.9195515	0.013471217	0.005825421	-0.500049848	0.055370389	-22.05694038	-26.54417841	-15.42439507	-13.56977316	-19.39882176	23.55813743
08/29/2019 01:11:32.178 PM	36	603.6286916	0.014417819	0.006221115	-0.591622106	0.133285947	-22.061527	-26.46609203	-15.33587694	-13.51008911	-19.34339627	23.58743779
08/29/2019 01:11:33.178 PM	37	600.1115003	0.015080877	0.004078603	-0.64000603	0.166651675	-22.014284	-26.49994496	-15.32257869	-13.54911495	-19.34648065	23.54985813
08/29/2019 01:11:34.179 PM	38	598.7533091	0.01464064	0.004462007	-0.656439615	0.060905497	-21.96352562	-26.34586285	-15.22827994	-13.55812613	-19.27394863	23.45383366

Figure 21 - Screen capture of raw data set's first 40 rows

Those are shown in Figure 22 in regions A, B and C, respectively. Region B is the region that encapsulates the necessary data for the modeling. Hence, removing regions A and C were necessary.

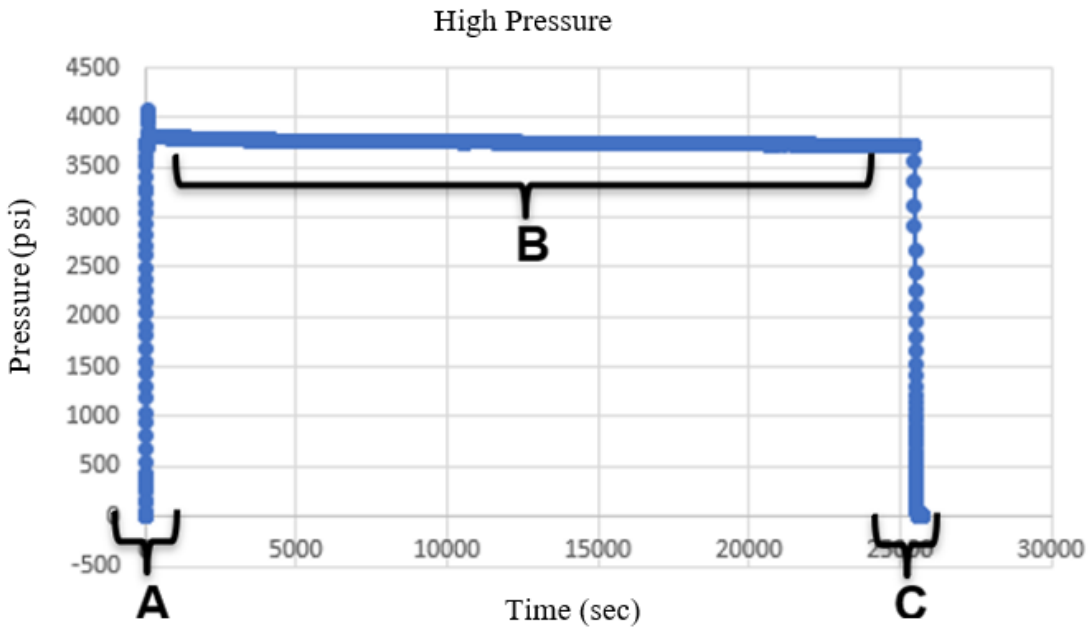


Figure 22 - Experimental Regions (depicted with results from a random experiment)

There were a few cases where the pressurization time was long due to the team manually resetting the pressure due to a system failure or possibly having to depressurize and go into the room to fix something and then immediately pressurize again (Figure 23).

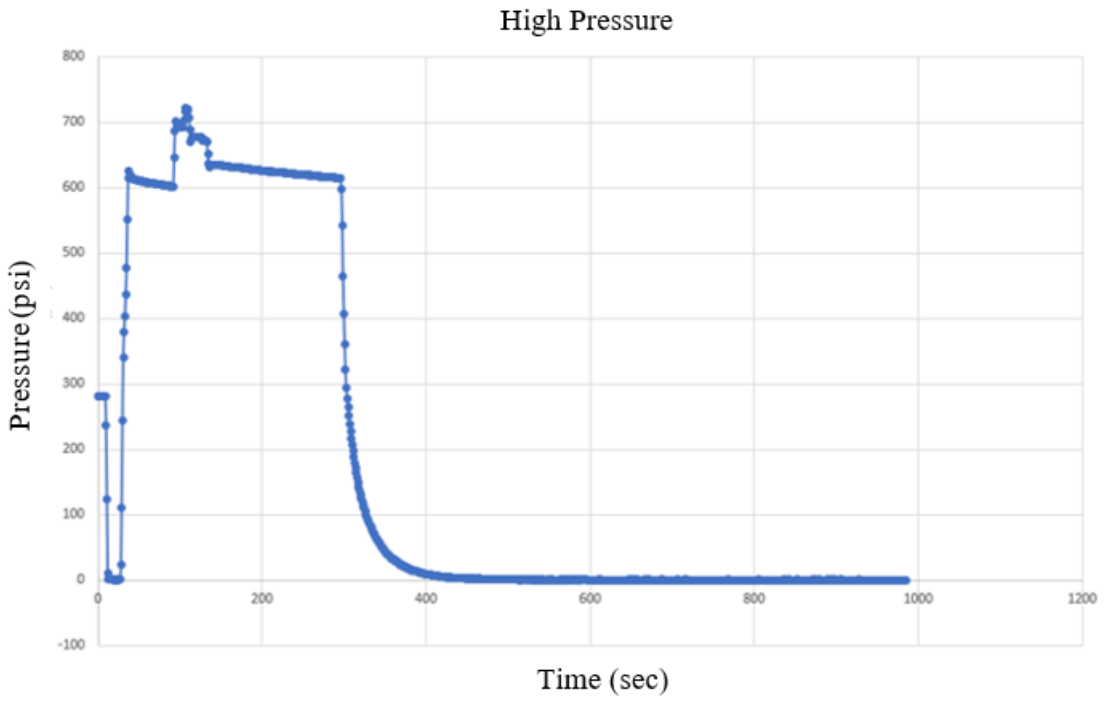


Figure 23 - 0˚C 600 psi Run 1 Argon High Pressure Data

In these cases, region B was considered the section of the graph where there was stable leakage after the last pressurization attempt. For example, in Figure 23, the region B would be from around 150 seconds until around 300 seconds. Sometimes the depressurization time, region C, made up the majority of the data sets, like in Figure 23. This is most likely because of a system fail-safe and the team not realizing it until the next morning. This means that the depressurization time could have lasted for up to eight hours and in this time the data shows zero psi for those eight hours, again explaining the behavior in Figure 23. These fail-safe are mechanically induced by the pressure transducers when a certain pressure is reached in the low-pressure chambers to avoid damage to the transducers. There is not a downside to these experiments other than the fact that they sometimes contain data for 200,000 seconds, making them difficult to open and manipulate due to their file size. The best way to identify regional shifts (i.e. when one region ends and another begins) was to graph the leak rate data like in Figure 22 or 23, then removing

the data associated with regions A and C. When the data would approach a regional shift, there was a significant difference in leak rate values which were then followed by a series of leak rate values all within a small range. In total, there were 321 experiments that needed to be cleaned. When the regions A and C were eliminated, the high-pressure leakage data looked like this:

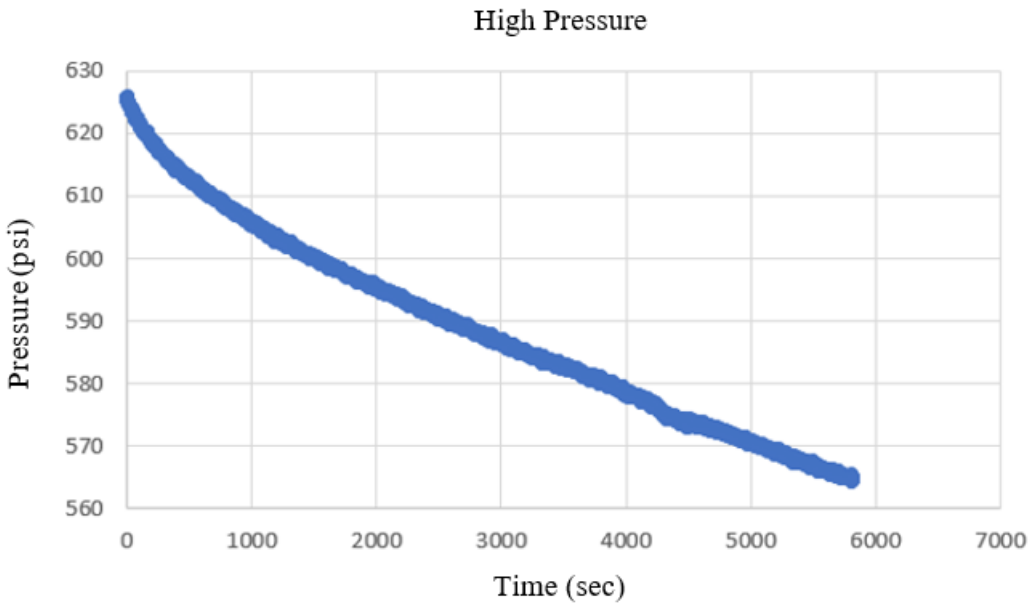


Figure 24 - high-pressure data post region A and C elimination

Then the data for low pressure side A would look like this:

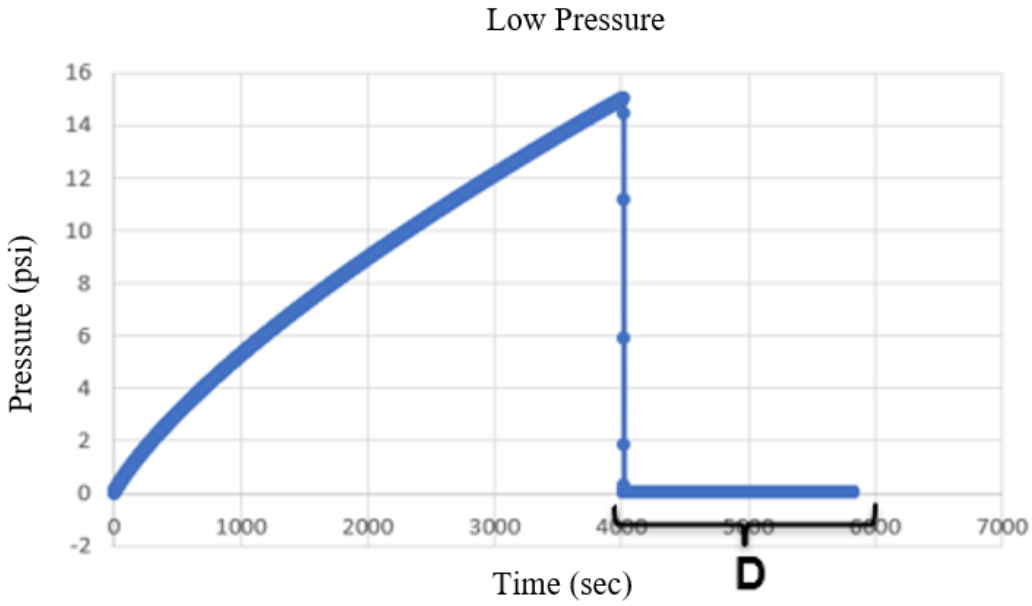


Figure 25 - low pressure data post region A and C elimination

For the purposes of building a model for the He/Ar leak rate ratios, the data collected by low pressure side A would be used. Therefore, the additional depressurization region labeled “D” needed to be eliminated. This region “D” is showing the point where the low-pressure chamber reached a maximum capacity and was programmed to release the gas to protect the pressure transducer.

The purpose for using the low-pressure side A data is so that a true leak rate from the high-pressure chamber could be determined. If the leak rate from the high-pressure chamber were used for model building, there would be no way to tell how much was leaking into low pressure side A, low pressure side B, or the atmosphere. Hence, tracking the gas that leaks across the barriers from the high-pressure chamber into the low-pressure side A allows for an appropriate leak rate across the barrier to be observed.

One of the major challenges to overcome with the 321 data sets was that they were all different sizes. Some experiments lasted 72 hours and some lasted 1 hour, it depended on the type of gas, the pressure and temperature conditions. The experiments would run until the system would release the gas from the low-pressure chamber to protect the equipment or until the experiment duration reached 3 days. The purpose of these experiments was to capture the fact that some of these leak rates were faster and some were slower. So rather than comparing the leak rates minute-by-minute or second-by-second, there would be a comparison of percent leakage. This means that the same percent of leakage would be compared to the corresponding experiment with the opposite gas. This allows for a more direct comparison between the leak rates of the different gas data sets.

For this method, VBA coding was the main tool. It was necessary to come up with a way to produce equally sized data sets from the varied experiment lengths. This was done by dividing the row count of each data set by the number 15, producing 15 groups of data per experiment. The number 15 was chosen because of the size of the smallest data set being 30 points, and at least 2 values needed to be in each group to produce a slope. Since the interest here is to model the leak rate, the slope of the values in each of the 15 groups was taken using linear estimation VBA coding as well. This method allows for the entire leakage behavior to be compared between the different gases. The data was then consolidated to create the individual experimental conditions' leak rate ratios. An example of what this data looked like finally is presented in Table 16:

RUN 2		
Helium	Argon	Ratio
LP A Slopes	LP A Slopes	LP A Ratio
0.0121	0.0019	6.3271
0.0117	0.0009	13.1694
0.0114	0.0012	9.8345
0.0112	0.0010	10.7926
0.0110	0.0010	11.2320
0.0109	0.0009	11.5853
0.0108	0.0009	11.8445
0.0106	0.0009	11.9435
0.0105	0.0009	12.2570
0.0104	0.0008	12.6427
0.0103	0.0002	52.6423
0.0103	0.0009	11.5293
0.0101	0.0008	12.7575
0.0101	0.0008	12.9133

Table 16 - Final Ratio Calculation Example for Run 2 0˚C 2250 psi

When the data was consolidated, there were some sets of ratios that were obviously compromised. The experimental ratio of Helium to Argon is theoretically around 2.0 and some of the experimental leak rate ratios were around 0.5 or even 20.0, but some data sets were showing ratios over 2000. These sets of data were considered errors as a result of instrument malfunction and were removed.

Once these ratios were determined for each set of conditions experiments, the ratios were compiled into one large column consisting of over 1000 rows, along with their corresponding temperatures and pressures. A preview of that data set is shown in Figure 26:

Temp	Pressure	Run	Ratio
-46	500	1	2.736534
-46	500	1	6.980444
-46	500	1	7.568846
-46	500	1	4.842116
-46	500	1	5.644376
-46	500	1	5.18069
-46	500	1	5.018066
-46	500	1	5.096454
-46	500	1	5.043045
-46	500	1	4.500734
-46	500	1	-3.56245
-46	500	1	8.255863
-46	500	1	6.762815
-46	500	1	5.549516
-46	500	1	4.998628
-46	600	1	3.074189
-46	600	1	3.069898
-46	600	1	3.061385
-46	600	1	3.297487
-46	600	1	3.258821
-46	600	1	3.086102
-46	600	1	3.243191
-46	600	1	3.255927
-46	600	1	3.236937
-46	600	1	3.314743
-46	600	1	3.407469
-46	600	1	3.322282
-46	600	1	3.35986
-46	600	1	3.370257
-46	600	1	3.43941
-46	2250	1	15.00071
-46	2250	1	14.21891
-46	2250	1	13.97256
-46	2250	1	13.82314
-46	2250	1	14.47391
-46	2250	1	14.38245
-46	2250	1	11.99532
-46	2250	1	11.7443
-46	2250	1	11.80833

Figure 26 - screen capture of final data consolidation's first 40 rows

With all of the leak rate ratios in one place, any ratio that was negative was removed since it is inappropriate to model a negative leakage. These negative values are likely due to noise in the data or the system instruments. The outliers were then removed using the same Tukey Fence method from model Method 2. The data set initially contained 1,148 data points and after the removal of negatives and outliers, the final data set contained 1,004 values, resulting in a 12.5% loss. This was deemed an acceptable loss due to the highly variable nature of the data and the difficulty associated with the data collection process. Also, a 12.5% loss still yielded a reasonably large enough data set to which modeling could occur; considering significantly

smaller data sets were used for the modeling Methods 1 and 2, it was believed that 1,004 data points were more than sufficient for modeling purposes.

4.2.2 Building the caret model

Once the data had been cleaned, it was time to start modeling the data. The first modeling attempt was a multivariate linear regression like in Methods 1 and 2, but it did not exhibit normality in the residuals. This was also the first attempt because the low pressure leak rates exhibited linear behavior, see Appendix C. This meant a different model type would be necessary but knowing which one to use was not immediately obvious. The final data when plotted did not have a particular shape (Figures 27-28).

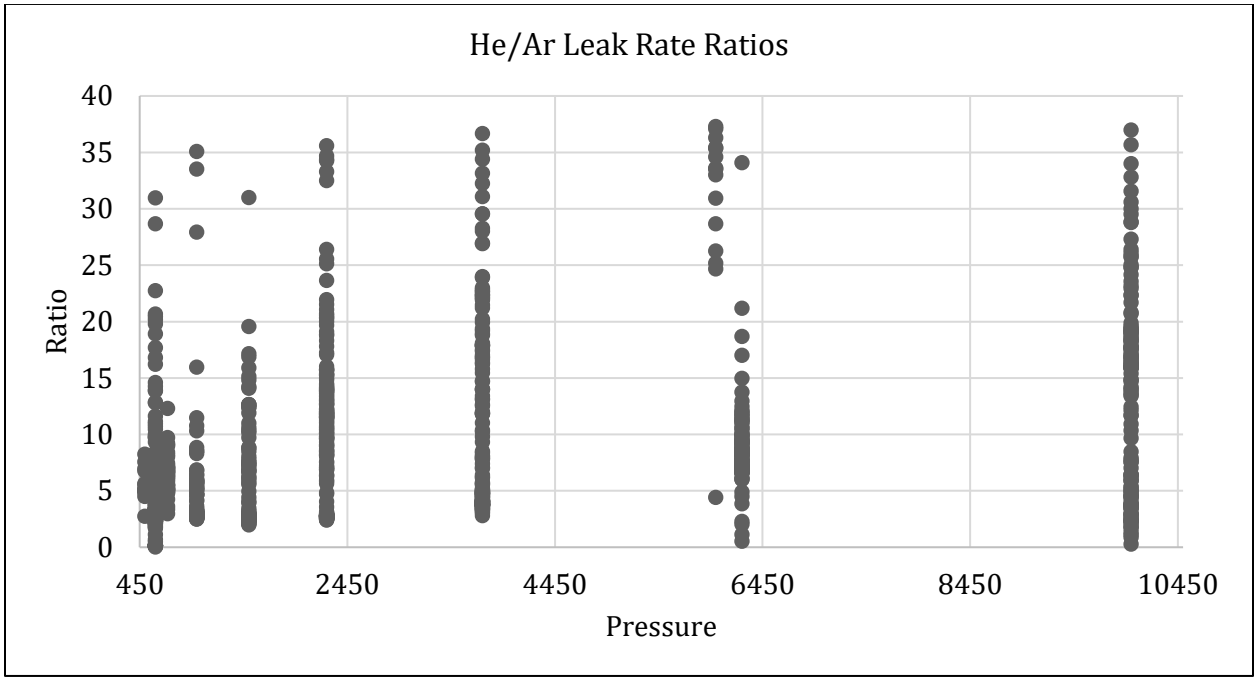


Figure 27 - He/Ar Leak Rate Ratios Plotted against Pressure

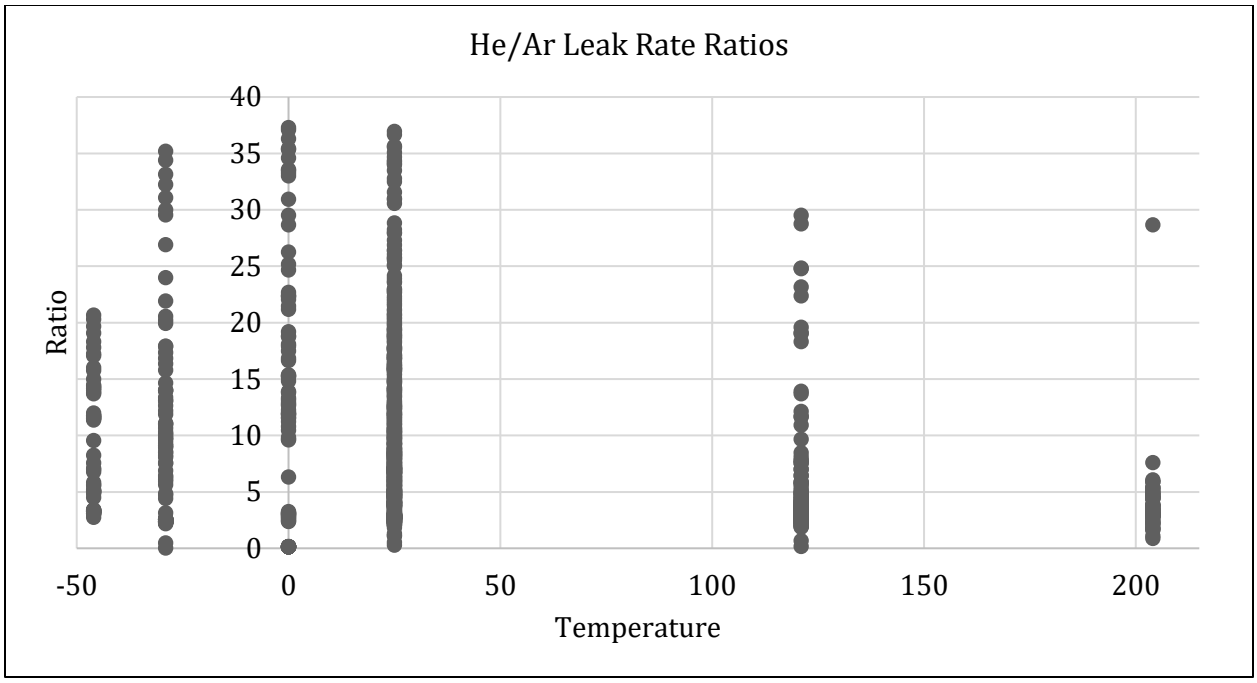


Figure 28 - He/Ar Leak Rate Ratios Plotted against Temperature

To investigate the best fitting model, the caret package in R studio was used. The caret (Classification And Regression Training) package "is a set of functions that attempt to streamline the process for creating predictive models" (Kuhn, 2019).This package has tools for data splitting, pre-processing, feature selection, model tuning using resampling, and variable importance estimation (Kuhn, 2019). Using caret, a data set can be split into a training data set and a testing data set, which then are used to train a model and validate that model. For this data set, the split was 80% training data and 20% testing data. These values are chosen randomly, but with a set.seed() function, the same random values are selected each time the code is run in order to maintain reproducibility. Hyperparameter tuning is needed in order to adjust the predictive model and optimize it to increase its validity. Some models have more tunable hyperparameters than others. An incredible benefit to the caret package is that it can train, test and tune 238 different types of models, making it an extremely useful tool for a situation such as this where

the proper model to use is unknown or not obvious initially (Kuhn, 2019). The caretList feature is one that allows multiple models to be tested in parallel, when using the function modelList, to determine which is the best-fitting model without having to test individual models one-by-one (see Appendix G). Initially, 7 models were created with the training data at a time to get an idea of what the modeling could look like; the goal here was to see if there was one model that stood out from the rest (Appendix G).

The scatter plot matrix below consists of the training data set (Figure 29). Again, creating a visualization to look for any obvious trends or relationships. The top left box shows the relationship of temperature on the leak rate ratio and the top middle box shows the relationship of pressure on the leak rate ratio. Both show a denser correlation of points toward the lower end of the x axis values.

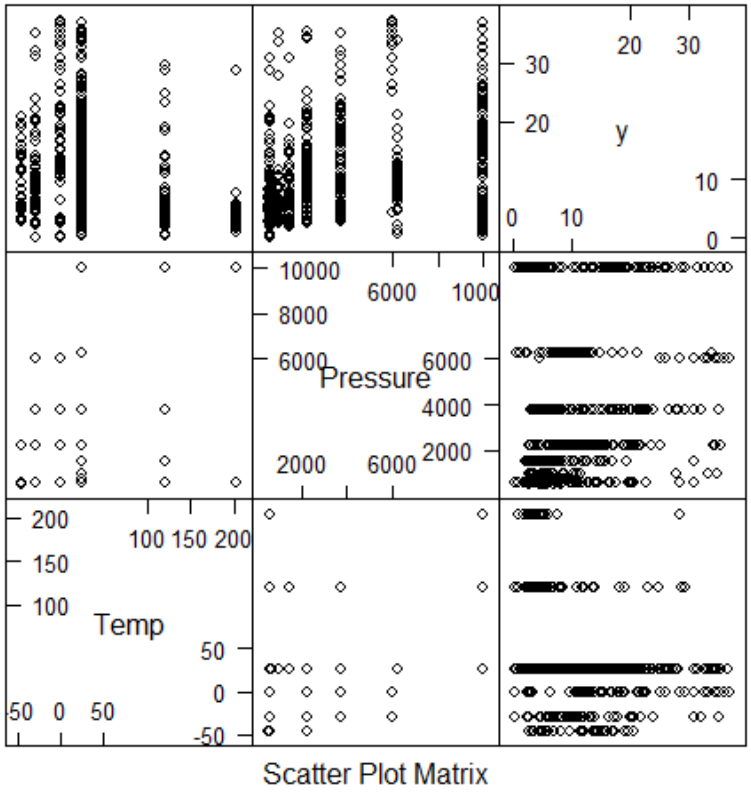


Figure 29 - Scatter Plot Matrix of Train Data

With these observations, there is a suspicion of possible skewing. The histogram shown in Figure 30 shows that the raw training data is skewed to the left significantly.

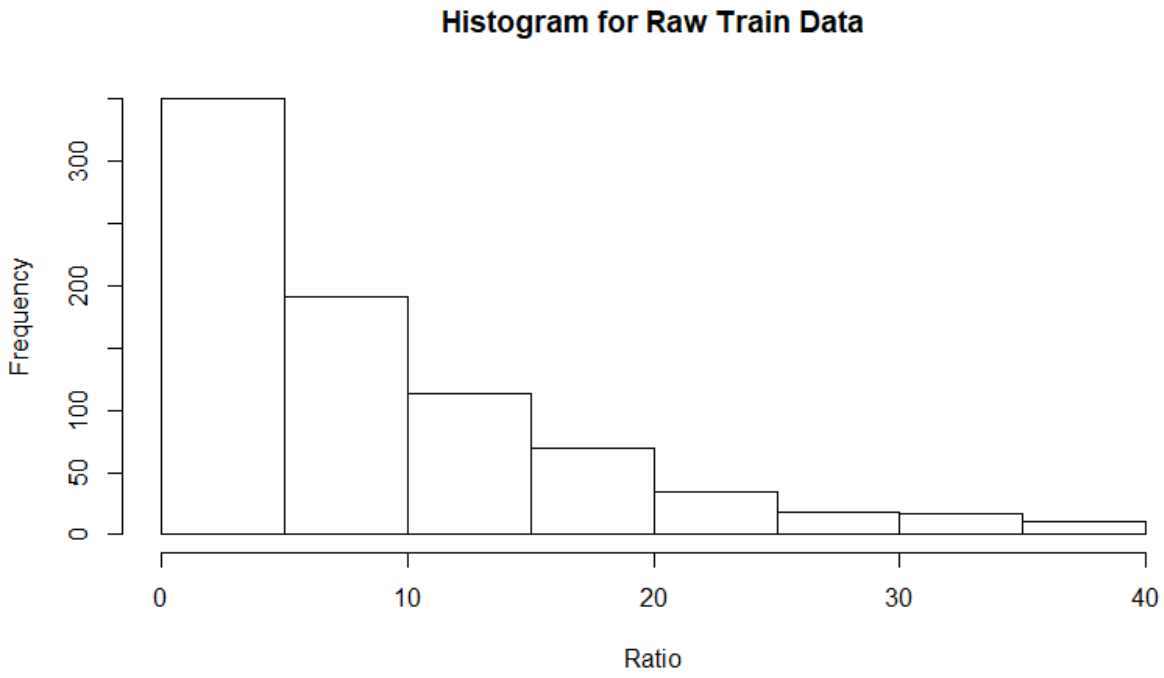


Figure 30 - Histogram for Raw Train Data

Models built in the caret package are often centered and scaled to increasing model strength and validity. In Figure 31, the train data after R automatically centered and scaled it is shown. There is still a significant skew toward the left side of the graph.

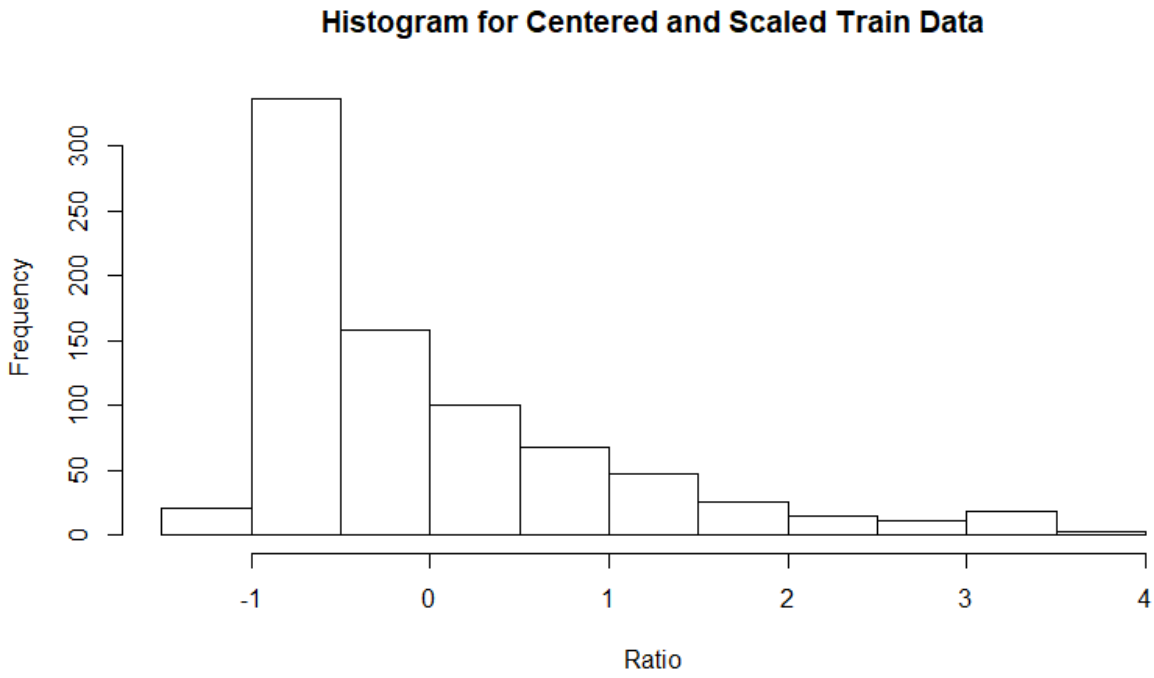


Figure 31 - Histogram for the Train Data after being centered and scaled

Despite the small amount of centering and scaling, this still helps to produce a more robust model. This same processed data is used for each of the models being tested using caret.

Of the 238 available models to model this data, about half of them were appropriate to use for the shape of the data and the type of data included in this set. Below are results for some of the better performing models produced with caretList:

Model Type	RMSE	Rsquared	MAE
Random Forest	6.109	0.399	3.916
Linear Model	6.999	0.205	5.123
Classification Tree	6.598	0.303	4.447

Table 17 - Model Results from caretList

4.2.3 The Random Forest Model

From this, there was one model that regularly stood out: Random Forest. Random Forest is a substantial modification of bagging, "a technique for reducing the variance of an estimated prediction function" (Hastie, Tibshirani, & Friedman, 2009). Much like decision trees or classification trees, Random Forests grows many classification trees (Berkeley, 2004). This bagging method build this large collection of trees so that they are de-correlated and then averages them (Hastie, Tibshirani, & Friedman, 2009). These forests are grown to the largest extent possible without any pruning (Berkeley, 2004). The basic idea of Random Forests is to "average many noisy and approximately unbiased models, and hence reduce the variance (Hastie, Tibshirani, & Friedman, 2009). Using the trees for this bagging technique is ideal because they can "capture the complex interaction structures in the data" (Hastie, Tibshirani, & Friedman, 2009). Random Forests are ideal for large data bases, can handle thousands of inputs, do not overfit the data, are easy to train and tune, and offers an experimental method for detecting variable interactions (Hastie, Tibshirani, & Friedman, 2009) (Berkeley, 2004).
Random Forest for Regression or Classification:

1. For $b = 1\ to\ B$:
    a. Draw a bootstrap sample $\boldsymbol{Z}^*$of size N from the training data.
    b. Grow a random-forest tree T_b to the bootstrapped data, by recursively repeating the following steps foreach terminal node of the tree, until the minimum node size n_{min} is reached,
        i. Select m variables at random from the p variables.
        ii. Pick the best variable/split point among the m.
        iii. Split the node into two daughter nodes.

2. Output the ensemble of trees $\{T_b\}_1^B$.

To make a prediction at a new point x:

$Regression$: $\hat{f}_{rf}^B(x) = \frac{1}{B}\sum_{b=1}^{B} T_b(x)$.

$Classification$: Let $\widehat{C_b}(x)$ be the class prediction of the bth random-forest tree.

Then $\hat{C}_{rf}^B(x) = majority\ vote\ \{\hat{C}_b(x)\}_1^B$.

Equation 10 – Algorithm for Random Forest for Regression or Classification

(Hastie, Tibshirani, & Friedman, 2009)

The deeper and the greater the forest sufficiently grows, the lower the model's bias will be (Hastie, Tibshirani, & Friedman, 2009). From the caretList operations, Random Forest displayed the highest R^2 value of any model tested.

To build a better Random Forest model, it was important to train and tune it by itself outside of a caretList function. So, a default Random Forest model was created using the caret package and the methods for tuning it included hyperparameter for loops within caret training packages (Appendix G). The hyperparameters of interest for tuning included: mtry, ntree, maxnode, and nodesize. The hyperparameter "mtry" is the "number of variables randomly sampled as candidates at each split"; "ntree" is the "number of trees to grow"; "maxnode" is the "maximum number of terminal nodes trees the forest can have"; and "nodesize" is the "minimum size of terminal nodes" (CRAN, 2018). A series of for loops outside of caret Random Forest training models to test and assess the optimal value was done for each hyperparameter (Appendix G). Through this, the optimal values for each hyperparameter that produced the highest R^2 value for the model were determined. Once these hyperparameters were determined,

the number of folds for the cross-validation in the trainControl function was adjusted manually until there was a point where the R^2 value began to level-off.

4.2.4 How the Random Forest caret Model Performs

This resulting Random Forest model found an interaction affect between the two input variables: temperature and pressure with a higher variable importance associated with pressure. “Two independent variables interact if the effect of one of the variables differs depending on the level of the other variable” (Glimo, n.d.). In terms of a Random Forest, this means that if a split on one variable in a tree makes a split to the other variable either systematically less possible or more possible (Berkeley, 2004). This is determined “under the hypothesis that the two variables are independent of each other” (Berkeley, 2004). This does not change how the input variables are presented to the model, but it does explain some interesting behaviors within the model.

Before any of the hyperparameter tuning, the default Random Forest trained caret model returned an R^2 value of 0.470 with a cross validation fold number of 20; after the hyperparameter tuning and increasing the fold value to 250 the R^2 increased to 0.722 with an adjusted R^2 value of 0.721, which is a significant increase. A tool called postResample is offered in the caret package for assessing the fitting ability of the models. With this, using the tuned Random Forest model, the testing data in the remaining 20% of the data are used to predict the associated output values. With those output values assigned to a variable, plug that variable and the actual experimental output data from the 20% training data to assess how closely related the predicted output values are to the actual output values in the testing data set. The R^2 associated with the postResample is 0.404, which is smaller than the 0.722 from the tuned model, but this is a common occurrence when using the postResample function. The 0.722 is associated with how closely the variance

associated with the Random Forest model matches that of the training data whereas the 0.404 is how closely the variance of the Random Forest model aligns with the variance of the test data. This is likely due to the high variance nature of the data set. For the purposes of comparing the linear models to this non-linear model later-on, the metric of focus is the adjusted R^2 of 0.721.

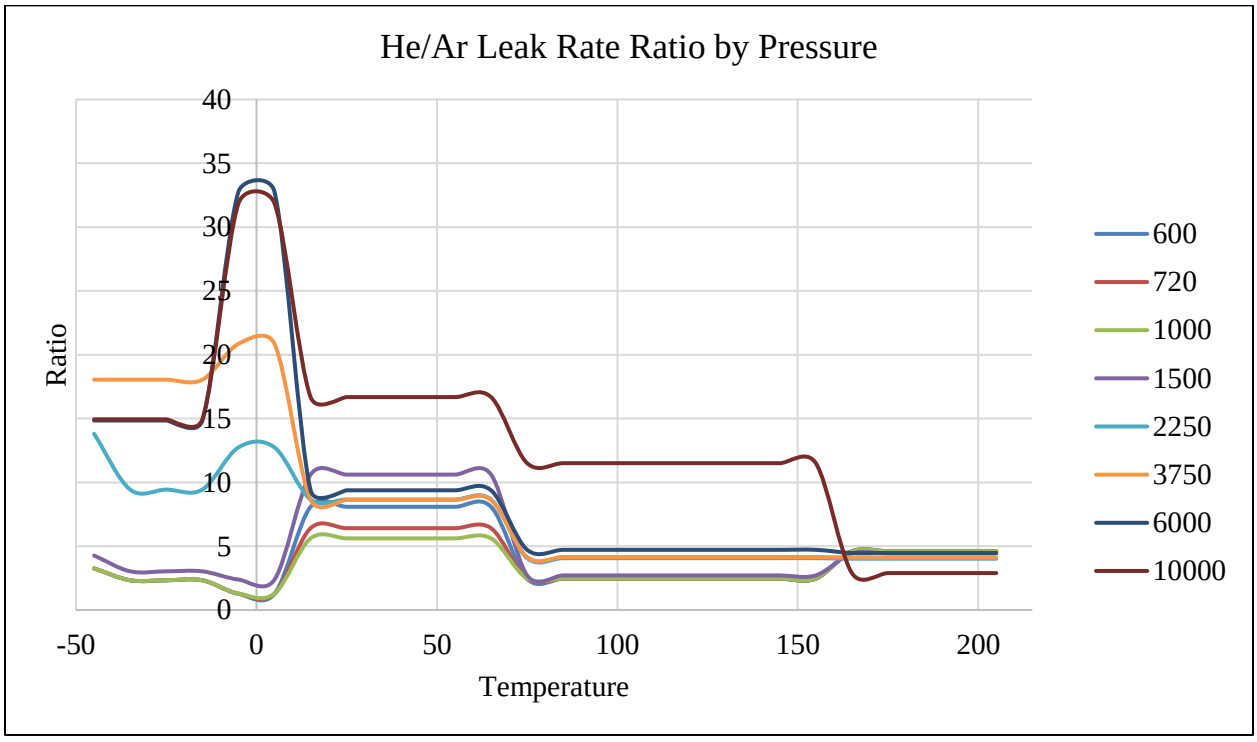


Figure 32 - Random Forest He/Ar Leak Rate Ratio Model (by pressure)

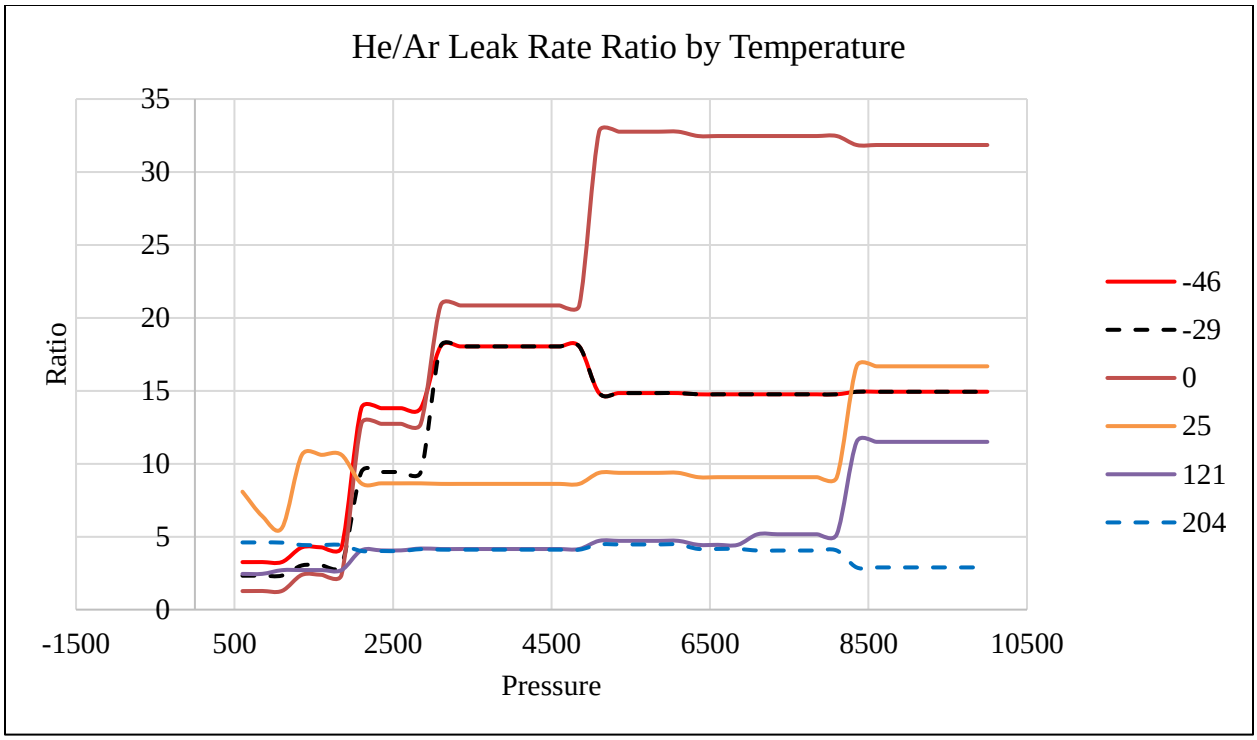


Figure 33 - Random Forest He/Ar Leak Rate Ratio Model (by temperature)

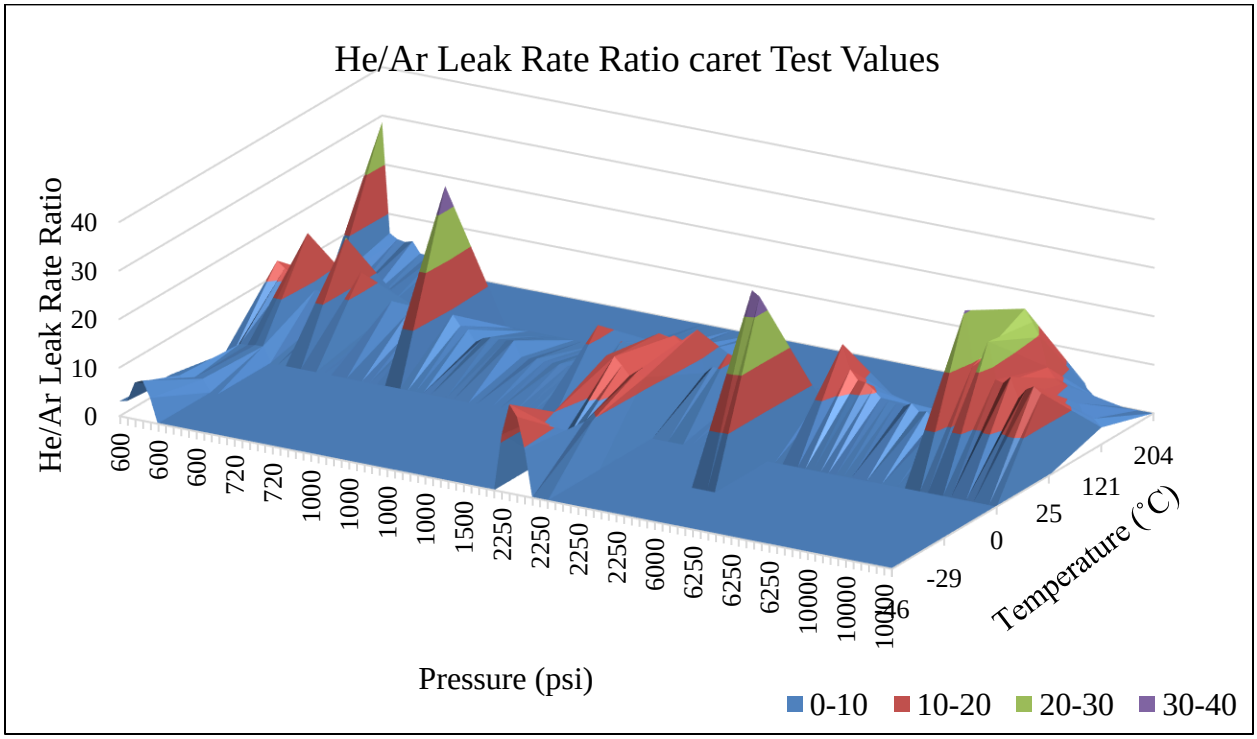


Figure 34 - Surface Plot of the Random Forest Test Data

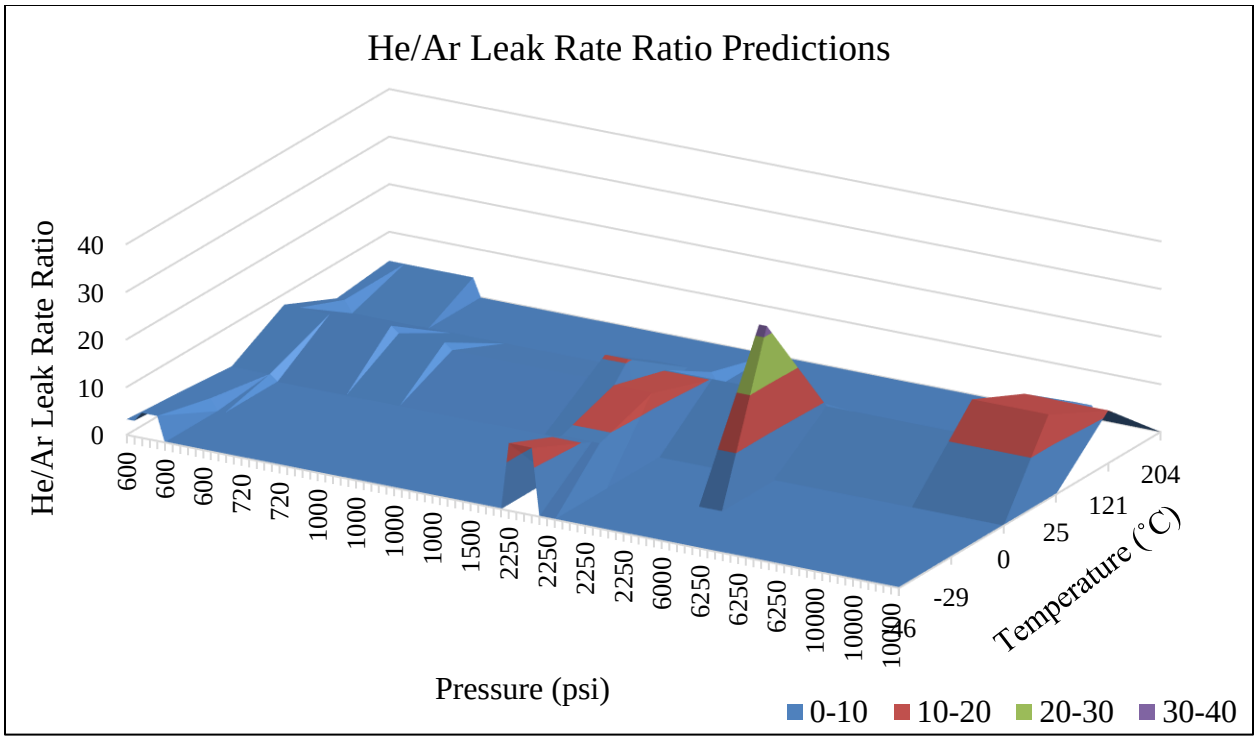


Figure 35 – Surface Plot of the Predictions made from Test Data Inputs

The surface plot in Figure 34 shows the actual output values for the test data input variables. The surface plot in Figure 35 shows the predicted output values using the Random Forest with the input variables from the test data set. The R^2 value of 0.404 is describing the amount of explained error between Figure 34 values and Figure 35 values. Figure 35 helps to show how difficult it is to fit a model to this data set. There is a higher concentration on the left side, but there are still significant values spread out toward the right side of the graph seen as "spikes" in the data. Considering this challenging model behavior, the predicted values actually do recognize the concentrated areas of the graph and dip/rise with those heavier clusters while also helping to smooth the dramatic spikes and provide a reasonable and reliable estimate. Hence, it helps to explain the behavior of the highly variable experimental data.

Another way to observe the Random Forest's ability to predict was to manually test data instead of randomizing it with the caret function. The data points including the pressure 3750 psi

were selected to be tested. They were removed from the data set used to build the Random Forest model, a total of 166 points. With this, a similar Random Forest model was created. The input variables for the removed points were put into the new Random Forest model in order to predict new outputs for those inputs. These outputs were then compared to the experimental outputs associated with those inputs. This was a way to see how closely a similar Random Forest model could predict the ratios for input variables that were not used to build the model. Figure 36 shows the predicted outputs plotted over the experimental outputs.

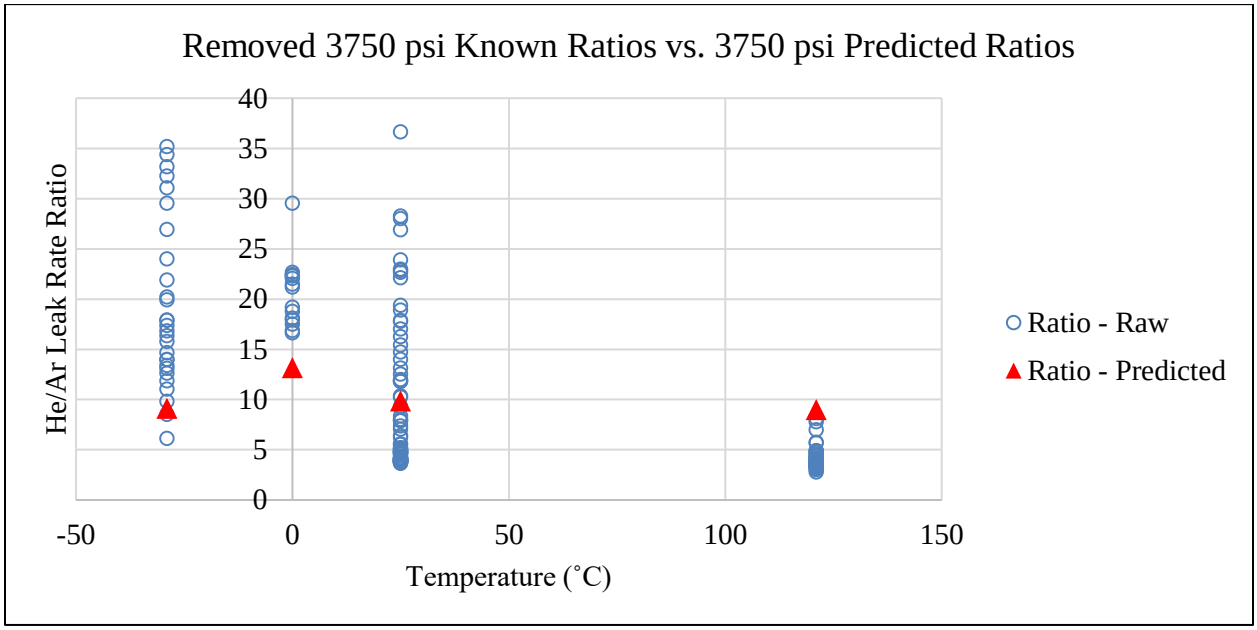


Figure 36 - Removed 3750 psi outputs compared to the predicted 3750 psi outputs

Figure 36 shows there is a close relationship between the predicted outputs and the experimental outputs. Hence, the Random Forest model performs well when predicted outputs for input variables not included in the model. This helps to increase confidence in a Random Forest model's abilities with this data set.

4.3 How the Models Compare

The three models have some differences but many similarities. They all show a positive effect due to pressure; they all take both temperature and pressure inputs, and generally exhibit the same kind of behavior, just at different magnitudes (Figure 37).

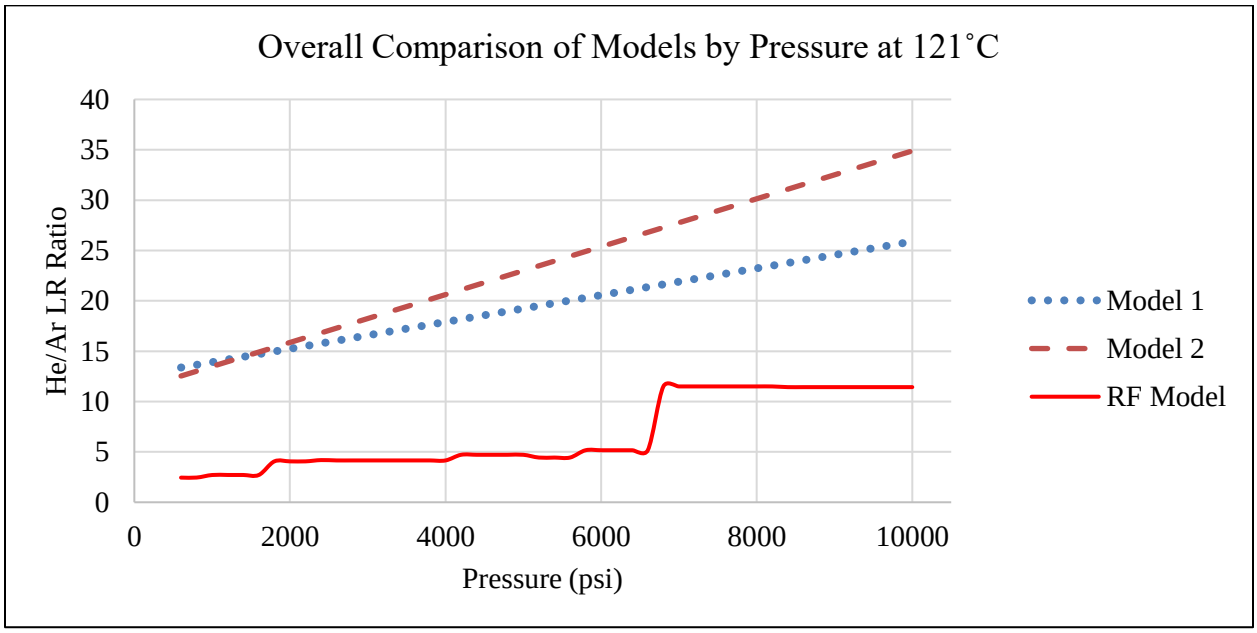


Figure 37 - Models 1 and 2 plotted with the Random Forest model at 121˚C

Perhaps one of the most significant differences between the linear models and the Random Forest model is the pre-processing of the data set. They came from the same raw data set. They explain the same data set to in different ways to different degrees. Even though Method 1 had stricter upper and lower bounds for acceptable data and hence fewer data points than Method 2, they both came from the same pre-processed data set where there is one He/Ar leak rate ratio associated with each individual set of experimental conditions. Thus explaining their similar shape and behavior. The Random Forest model considered multiple He/Ar leak rate ratio possibilities for each set of conditions. The RF model contained 250% more data for model building than Model 2 and 500% more data than Model 1. In terms of Methane leakages, Model

1 is more conservative in its estimate than Model 2; and the RF model is significantly more conservative than Model 1's estimates. This is determined by the magnitude of the plotted graphs in Figure 37.

Model	R^2	Adjusted R^2
Method 1	0.301	0.234
Method 2	0.311	0.275
Random Forest	0.722	0.721

Table 18 - Model Metric Comparison

According to Table 18, the adjusted R^2 of Method 1 is 0.234, which is less than that of Method 2, which is 0.275. And Method 2's adjusted R^2 is less than the Random Forest model's adjusted R^2 of 0.721 which is a more significant difference when compared to the difference between the first two methods. Although the lower adjusted R^2 values are still worth noting and it has been determined that it is not always going to have an adverse effect on assessing a model's validity, the fact that the adjusted R^2 for the same data set has been increased to 0.721 when changing to a Random Forest model encourages the selection of the RF model. Having the highest adjusted R^2 value of all the potential models, the highest data inclusivity, and the more conservative nature of the RF model, it has been selected as the model of choice for investigating the ISO allowable leakage standards.

5 Diffusion modeling

Diffusion is the movements of particles through a concentration gradient (Dickson, 2020). ISO 15848-1 provides allowable Methane leakage concentrations and this diffusion modeling allows the concentrations to be translated into a leak rate. This allows for the comparison of the testing Helium leak standards to be compared to the allowable Methane

amounts. Hence, the selected predictive model must be combined with diffusion modeling in order to get ISO leak rate comparisons. This diffusion modeling was completed by Abigail Hovorka with the aid of Milad Najafbeygi and Schlumberger advisor, Dr. Raghu Madhavan.

5.1 Point source model

For this, a diffusion model that can calculate leak rate from concentration at given point is applied. The assumptions for this model include steady state, three-dimensional free diffusion, continuous point source of leak rate, and the atmospheric concentration can be simplified to zero. The 3D diffusion Equation reads:

$$\partial_t \vec{c} = \nabla^2 \vec{c}$$

Equation 11 – 3D-Diffusion (Part I)

$$\text{SS} \rightarrow \frac{\partial \vec{c}}{\partial t} = 0 = \nabla^2 \vec{c}$$

Equation 12 - 3D-Diffusion (Part II)

From this, the solution is:

$$\vec{c}(r) = \frac{\dot{m}}{4\pi D r} + c_\infty$$

Equation 13 - Final Diffusion Equation

In this, $\dot{m}$ represents a leak rate, D is the diffusion coefficient, which is a function of temperature, and c_∞ is the atmospheric concentration, which can be simplified to zero. The published diffusion co-efficient values in air at 20ºC are given in Table 19.

Diffusion Coefficients in Air at ~20˚C (m^2/s)	
Argon	0.0000189
Helium	0.0000697
Methane	0.0000196

Table 19 - Diffusion coefficients

(Engineering ToolBox, 2001)

Estimated equivalent CH_4 leak-rates for a given concentration at 1" from the point source at 20^0 C are given in Table 20.

Class	CH_4 (Sniffer) concentration	Equivalent leak rate (mg/s)	mol/s
AM	≤50	0.000205	1.277E-08
BM	≤100	0.000410	2.553E-08
CM	≤500	0.002048	1.2774E-07

Table 20 - Allowable maximum leak rate vs concentration limits

(International Organization for Standardization, 2015)

5.2 Effect of ambient temperature on concentration vs allowable leak-rate

Diffusion is affected by the ambient temperature. Higher the temperature, faster the diffusion. By applying the results from Equations 11, 12, 13 from this report, a diffusion co-efficient multiplier has been estimated for the 20ºC reference value.

The following Equation 14 shows the relationship between diffusion coefficient and temperature. D_{AB} is the diffusion coefficient between A and B gases.

$$D_{AB} = \frac{2}{3}\sqrt{\frac{k_B^2}{\pi^3}}\sqrt{\left(\frac{1}{2m_A}+\frac{1}{2m_B}\right)\frac{4T^{3/2}}{P(d_A-d_B)^2}}$$

Equation 14 - Diffusion Coefficient Multiplier for 20˚C

In this Equation, P is the atmospheric pressure, T is the temperature., k_B is Boltzmann constant. It is equal to 1.3807(79)×10−23 J/K. m_A and m_B are molecular mass, and d_A and d_B is the atomic diameter. Since we measure the diffusion at the outside of the control volume, T = 293.15 K (20ºC, environmental temperature) and the pressure is equal to atmospheric pressure, P = 1.01325x105 Pa. Generally, 78% of air is considering Nitrogen and 21% Oxygen. So, for any gas diffusion on-air, 78% of total diffusion is on Nitrogen and 21% of that is on Oxygen. Equation 14 can be reduced to Equation 15 by applying the known parameters.

$$D_{\mathrm{AB}} = 2790 \frac{T^{1.622}\left(\frac{1}{M_A}+\frac{1}{M_B}\right)^{0.5}}{P\left(V_A{}^{1/3}+V_B{}^{1/3}\right)^2}$$

D_{AB} = diffusion coefficient (cm^2/s)
T = temperature (K)
P = pressure (atm)
M_A, M_B = molecular weight
V_A, V_B = molecular volume at the normal boiling point (from boiling point density) (cc/gmol)

Equation 15 - Diffusion Coefficient Equation

$$\frac{D_1}{D_0} = \left(\frac{T_1}{T_0}\right)^{1.622}$$

Equation 16 - Equation for Finding the Diffusion Coefficient

Equation 16 shows how the diffusion coefficient D_1 at a new temperature T_1 can be estimated from a known reference value D_0 at T_0.

Calculated correction factors R for a range of ambient temperatures T (in ˚C), defined by R(T), are given in tabulated and graphical forms below (Figure 38). This translates to allowing a temperature dependent leak-rate for CH_4 to meet a given concentration based CH_4 tightness

class. For example, the multiplier of 0.89 at 0ºC ambient temperature allows a lower CH_4 leak-rate to meet a given CH_4 class, while at 50ºC, a multiplier of 1.17 allows a higher CH_4 leak-rate to meet the same tightness class.

T (˚C)	R(T)
-20	0.79
-15	0.81
-10	0.84
-5	0.87
0	0.89
5	0.92
10	0.95
15	0.97
20	1.00
25	1.03
30	1.06
35	1.08
40	1.11
45	1.14
50	1.17
55	1.20
60	1.23

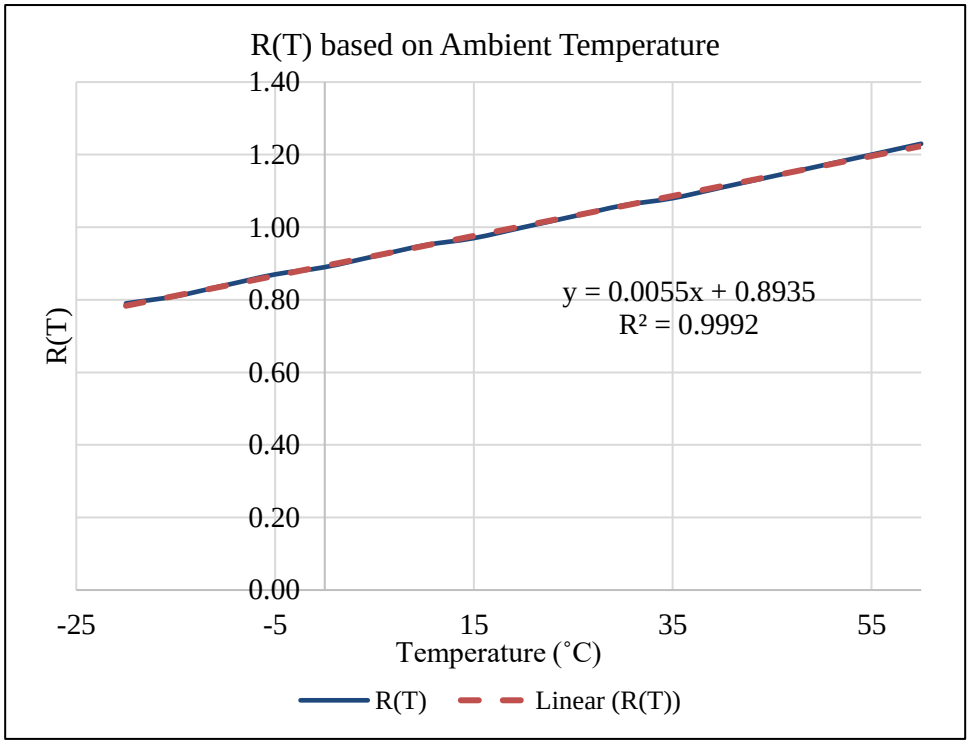


Figure 38 - Diffusion correction factor based on ambient temperature

5.3 Effect of distributed leakage from around a valve stem

The differences between a single point leakage (point source) from the stem against the likely scenario of distributed leakage from around the stem are shown below. Stem to barrier seal interface was modelled as a set of 8-point sources equally distributed along the circumference, and the leak-rate was evenly distributed among the point sources. Sniffer measurement was at a 1” point away from the stem circumference.

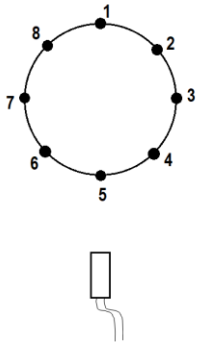


Figure 39 - Stem circumference is modelled as 8 discrete point sources with equal spatial and leak-rate values.

With distributed leakage as modelled in Figure 39, a higher total leak-rate that is a function of the stem diameter can be allowed and still meet the concentration threshold at a point. These results are shown as diffusion correction factor in Figure 40. For example, a 3" stem with distributed leakage gives a concentration of 0.65 times that of a point source. In converse, for a 3" stem, a higher leak rate that is 1.0/0.65= 1.54 times that of a point source can be allowed and still meet a select concentration threshold.

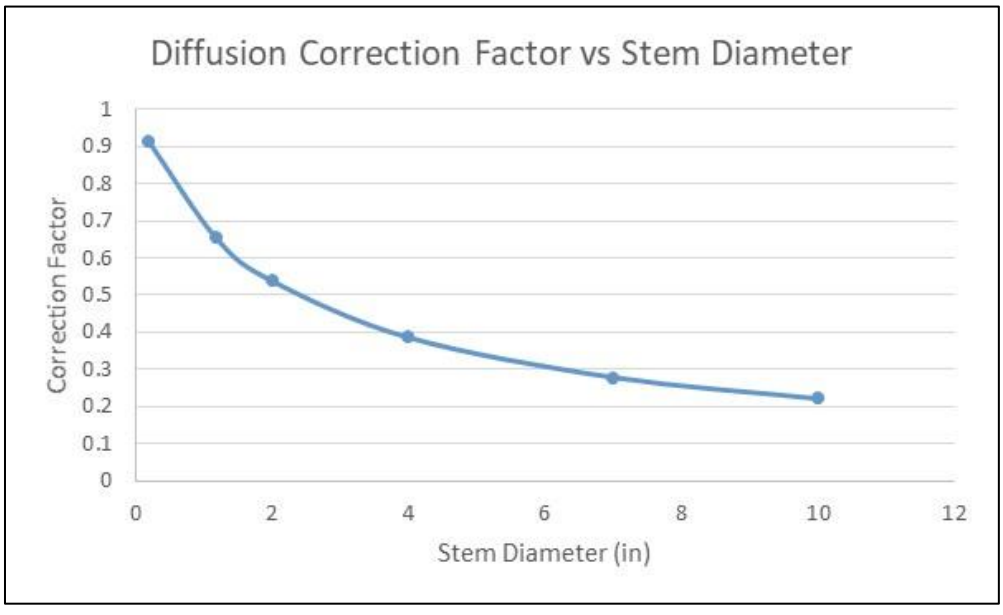


Figure 40 - Diffusion correction factor considering stem diameter (mm)

6 Model leak rate prediction in relation to the ISO tightness classes

ISO 15848-1 standard tightness classifications are based on the different allowable concentrations of Methane. There are three classes that increase in severity as the class increases. The classes are A, B and C, the allowable air pollutant concentration in parts per million by volume for the classes are 50, 100, and 500 respectively. This is outlined in Table 21.

Class	Measured leakage (sniffing method) ppmv
AM	≤50
BM	≤100
CM	≤500

Table 21 - Tightness classes for stem (or shaft) seals with Methane

(International Organization for Standardization, 2015)

The maximum allowable CH_4 leak rates estimated from the given threshold concentrations are given below (Table 22) for 20˚C atmospheric temperature with a point source.

Class	CH_4 (Sniffer) concentration (ppmv)	Equivalent leak rate (mg/s)	mol/s
AM	50	0.000205	1.277E-08
BM	100	0.000410	2.553E-08
CM	500	0.002048	1.2774E-07

Table 22 - Allowable maximum leak rate vs concentration limits

(International Organization for Standardization, 2015)

The Methane sniffer data is converted into equivalent Helium and Argon leak rates using the experimentally determined leak rate ratios, regression model, and the diffusion model. The RF model has been selected to establish threshold Helium leak rates with which the ISO 15848-1 standards will be compared.

The assumptions for converting the Methane concentrations into Helium and Argon leak rates in the examples given below (Tables 23 through 25) include: point source diffusion, 20˚C free diffusion in air, Ar/CH_4 leak rate ratio of 1.5 from experiments, and a representative He/Ar leak rate ratio Random Forest model.

To illustrate the implementation of these methods, three examples have been created for a temperature rating of 121˚C using the Random Forest model from chapter 4.2 of this report.

Class measured leakage (ppmv)	CH_4 Leak Rate (mol/s)	Modeled Helium Leak Rate (mol/s)	Argon Leak Rate (mol/s)	ISO 15848-1 Helium (mol/s, 30mm seal)
50	1.277E-08	7.780E-08	1.915E-08	5.873E-11

Table 23 - Class A example at 2250 psi and 121˚C

Class measured leakage (ppmv)	CH_4 Leak Rate (mol/s)	Modeled Helium Leak Rate (mol/s)	Argon Leak Rate (mol/s)	ISO 15848-1 Helium (mol/s, 30mm seal)
100	2.553E-08	9.394E-08	3.830E-08	5.873E-10

Table 24 - Class B example at 600 psi and 121˚C

Class measured leakage (ppmv)	CH_4 Leak Rate (mol/s)	Modeled Helium Leak Rate (mol/s)	Argon Leak Rate (mol/s)	ISO 15848-1 Helium (mol/s, 30mm seal)
500	1.277E-07	2.203E-06	1.915E-07	5.873E-08

Table 25 - Class C example at 10,000 psi and 121˚C

The final He/CH_4 leak rate ratio model using the Random Forest modeling determined in R Studio using leak rate ratio trend data has been used to calculate the Modeled Helium Leak Rate (mol/s) values.

In Tables 23-25, the Helium leak rate is calculated by using the He/Ar Random Forest model and the Ar/CH_4 leak rate ratio of 1.5, which creates the He/CH_4 relationship. In every

example, the ISO Helium leak rate is significantly smaller than the modeled Helium leak rate determined by the regression model. In Table 26 the percent of leak rate increase from the ISO standard's leak rate to the leak rate modeled with experimental data are shown.

Class	Modeled Helium Leak Rate (mol/s)	ISO Helium Leak Rate (mol/s)	Difference	Percent Increase
A	7.780E-08	5.873E-11	7.774E-08	132434.465%
B	9.394E-08	5.8703-10	9.335E-08	15902.791%
C	2.203E-06	5.870E308	2.144E-06	3652.666%

Table 26 - The Percent Increase from the ISO Leak Rate to the Modeled Leak Rate

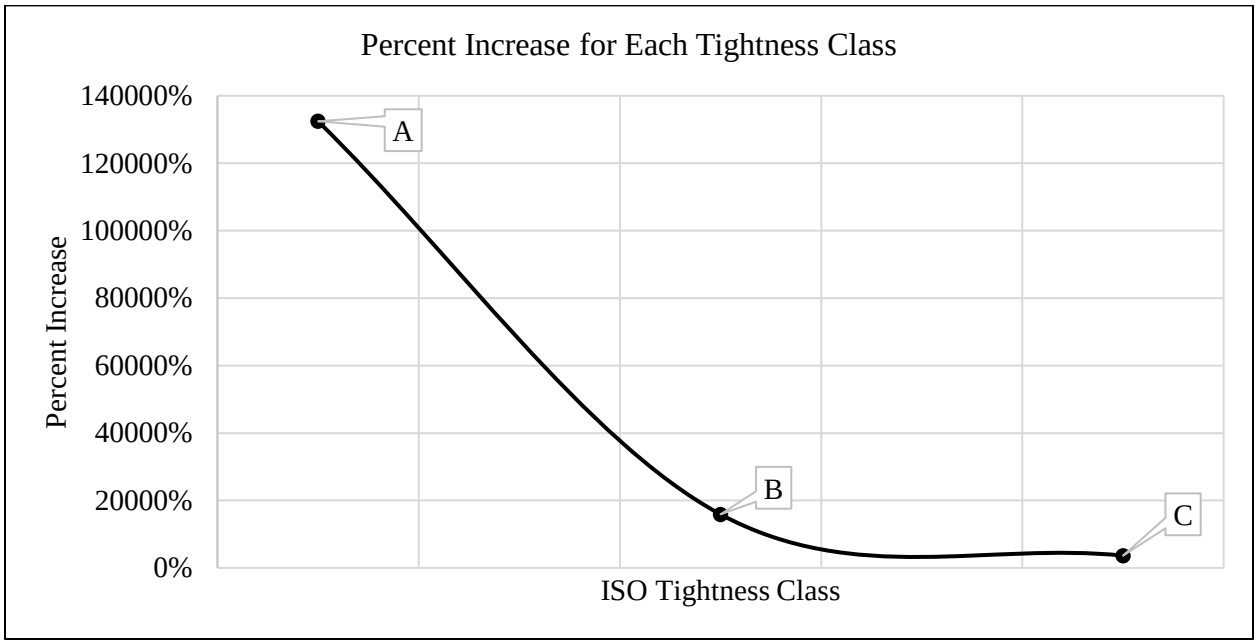


Figure 41 - Percent Increase from ISO to Model per Helium Tightness Class

In Table 26, it is shown that as the ISO class decreases, the percent increase (or change) from the standard's required Helium leak rate to the modeled leak rate increases significantly. Figure 41 helps to show the large jump in percent increase from Class B to Class A.

In Tables 23-25, the equivalent Argon leak rate is also the calculated that has been converted from the Methane leak rate using the Ar/CH_4 leak rate ratio. The ISO 15848-1

standard has never addressed Argon as a testing gas so there is no comparison to be made from the ISO standard's requirements.

7. Discussion

Multiple methods were tested to model the He/Ar leak rate ratio data in order to describe better the relationship between Helium and Methane. Like the researchers from the 2015 Journal of Fluid Science and Technology study on porous compression packing rings, the predicted model values were graphed with the original measured data to assess the different models' behavioral accuracies (Kazeminia & Bouzid, 2015). From these different leak rate ratio models, there was one that stood above the rest, the Random Forest model. This model's ability to model complex behavior through cross validation and advanced decision tree analysis pushes it above the rest of the models. Unlike the same research team who studied the porous compression packing rings, performance metrics were used to assess the Random Forest models' quality of predictions in addition to graphing the predicted data alongside the original data, which is similar to the research team's methods (Kazeminia & Bouzid, 2015). The Random Forest model was able to capture the data's chaotic behavior and large variance and produce a model with an R^2 value of 0.722, higher than those of the other methods with values of 0.301 and 0.311.

The 2015 study took into consideration the general issues regarding stricter regulations surrounding fugitive emissions testing; they mentioned observing issues among "standards such as TA-Luft, ISO 15848-1 and API 622 and 624 and others" and their potentially unnecessary needs to be revised (Kazeminia & Bouzid, 2015). However a study done by researchers from Clarke Valve did specifically focus on ISO 15848-1 and one other standard, coming to the same conclusion that the standards are extremely difficult to meet (Daniels & LeBoeuf, 2019). This

project and thesis focus specifically on the legislation-inspired revisions associated with ISO 15848-1. When the Random Forest model predictive abilities were applied to the ISO 15848-1 standard and conditions, this predictive model showed that the actual behavior of Helium does not match the theorized behavior from the ISO standard. As shown in Table 26 and Figure 41, the percent increase from the allowable Helium leak rate concentration values from ISO 15848-1 to the predicted Helium leak rate concentration values from the Random Forest model is incredibly significant. This proves that the ISO 15848-1 Helium tightness classifications do not have a proper correlation to the allowable Methane leakage associated with the tightness classes. Had there been a percent increase been less than 10%, there may be a chance to argue that there is some variation there and that the ISO standard does have some kind of correlation between Helium and Methane. However with percent changes of 132,434.465%, 15,902.791%, and 3,652.666% for Helium tightness classes A, B, and C, respectively, there is obviously a disconnect between the tightness classes' associated Methane leak rate and the testing Helium leak rate and thus, an urge to change the classifications. Unlike the 2019 Clarke Valve study, a call to make adjustments to the ISO 15848-1 standard is being made rather than designing an entirely new valve that should pass the existing, more strict tightness classes (Daniels & LeBoeuf, 2019). Although Clarke Valve has responded to the needs of the ISO 15848-1 standard, it can be seen that the standard itself is not in line with its intended purpose of correlating the Helium test leak rates to the allowable Methane leak rate concentrations in application. An entirely new valve may not be the best solution to the problem like they suggest since the standard itself is flawed.

Something that neither of these pieces of literature address is the possibility of having a more appropriate test gas for these crucial fugitive emissions standards. This project has found

that Argon performs more like Methane, which was theorized based on the similarities between the Argon and Carbon molecules. Which begs the question, is Helium really the better test gas? Both Argon and Helium can be detected by spectrometers, making both of their leakages traceable; both gases are non-reactive and have extremely low melting points, all of which make them feasible choices for industrial testing (TQC, n.d.) (Royal Society of Chemistry, 2020) (Vacuum Instruments Corporation, n.d.) (Thomas Jefferson National Accelerator Facility - Office of Science Education, 2020). However, it is easier to capture and store Argon from the atmosphere than it is to capture and store Helium as discussed in the Leak Specialists report from 2014 (Chamberlain, 2014). Considering that there is a global Helium shortage and there are many uses for Helium that cannot use another gas (i.e. MRI machines or research involving liquid Helium), it is beneficial to open the discussion on switching to Argon in areas that will allow for it (Chamberlain, 2014). In addition, Argon is more cost-effective than Helium. When purchasing industrial grade cylinders of testing gas, Helium costs 5.8 times more per cylinder than Argon (AirGas, 2020). With this, Argon is a better option for test gas selection. Hence, it is necessary to begin the investigation of implementing more industry-wide applications of Argon as a test gas, beginning with standards such as ISO 15848-1.

8. Conclusions

A concern that environmental organizations have become increasingly more aware about in the past decade is the environmental risk associated with fugitive emissions and how to reduce their potential environmental harm. Fugitive emissions from the oil and gas industry's many explorative endeavors as well as from their production and supply lines have become a main focus for these environmental organizations (Daniels & LeBoeuf, 2019). This is because a

common gas that runs through pipes and machinery is Methane. Methane, when released to the atmosphere, becomes a significant contributor to the greenhouse gases "due to its capacity to trap by volume 28 times more heat than carbon dioxide" (Daniels & LeBoeuf, 2019). Hence, there is significant interest in preventing Methane from escaping to the atmosphere in order to reduce environmental harm. There are many standardization organizations in the world who have been called to adjust their standards to be stricter in an effort to reduce fugitive emissions of Methane in the oil and gas industry. One organization in particular that has been affected by this is the International Organization for Standardization (ISO). ISO 15848 is concerned with the fugitive emissions of Methane from industrial valves in the field (International Organization for Standardization, 2015). Manufacturers have been struggling to get their valves to pass the updated ISO standards and have either had to resort to risky Methane tests in their facility, redesign their entire product, or contribute to the global Helium shortage through extensive testing to get their valves to pass the ISO 15848-1 standard (Daniels & LeBoeuf, 2019) (Kazeminia & Bouzid, 2015).

So why is it that so many manufacturers are struggling to pass their equipment when testing with Helium (Kazeminia & Bouzid, 2015)? The ISO 15848-1 standard does not effectively correlate Helium to Methane within its guidelines in these new endeavors to make the standards stricter to reduce global Methane fugitive emissions. In Tables 23-25, the Modeled Helium Leak Rate is greater than the ISO mandated Helium leak rate. In other words, the ISO Helium leak rates are stricter than the modeled Helium leak rates created using experimental data. Table 26 and Figure 41 show that the percent change from the Helium leak rates required by ISO 15848-1 in a testing facility and the Helium leak rate determined through modeling experimental data is very large. With this, it can be determined that there is little to no

correlation between the allowable Methane concentration tightness classes and the Helium test leak rates for the tightness classifications defined by the standard.

Since there is this disconnect between Helium and Methane, a suggestion can be made to use Argon as a replacement test gas. It has similar leakage characteristics compared to CH_4 and is a better test gas than Helium to substantiate Methane tightness classes, seen in Tables 23-25. Argon is a readily available and it is a cheaper inert test gas, meanwhile Helium is facing a supply shortage and due to that has become a less economical test gas.

In essence, the primary conclusion for this report is that there is not a clear correlation between Helium and Methane in ISO 15848-1 and there needs to be a better relationship showcased between Helium and Methane in ISO 15848-1, if Helium continues to be the primary test gas. Protecting the environment is the highest priority, so adjusting these standards must be done carefully, but there is a need to reassess the relationship between the test gas and Methane. And if there is no opposition to changing test gases, then a deeper investigation of Argon as a test gas is strongly suggested.

9. Future Work

The claim with this work is not that the new Helium leak rates for testing industrial valves have been found, but that there needs to be a deeper investigation for how the Methane concentrations correlate to Helium test leak rates. There are inconsistencies and they need to be addressed. Future investigations should involve more complex models or even a more involved factor-effect analysis to rule out any data effects external to the system.

And besides this, Argon needs to be investigated further as well. The third party experimentation needs to be validated with other experiments like those outlined in ISO 15848-1

for passing industrial valves. Switching to Argon is a much more sustainable option in terms of environmental concerns, human health and finances.

The next step for this research will be the investigation of these leak rates in a dynamic scenario. This report is concerned with just static leak rates, but ultimately there is a need to also understand the relationship between these gases in a dynamic set up. This research can be a starting point for any dynamic tests in the future.

10 References

AirGas. (2020, July). Industrial Grade, Size 300 Cylinder. Oklahoma. Retrieved July 2020

Baars, A. (n.d.). *White paper: Fugitive Emission testing, Methane vs. Helium.* Retrieved May 2020, from Ventil: https://ventil.nl/wp-content/uploads/2019/06/Website-White-paper-1-Emission-testing-Methane-vs-Helium.pdf

Berkeley. (2004). *Breiman and Cutler's Random Forests*. Retrieved 2020, from https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm

Cameron, A. C., & Windmeijer, F. A. (1997). An R-squared measure of goodness of fit for some common non-linear regression models. *Journal of Econometrics*, 329-342. Retrieved May 2020

Carmin Chappell, J. (2019, June 21). *The worldwide Helium shortage affects everything from MRIs to rockets - here's why*. Retrieved November 2019, from CNBC: https://www.cnbc.com/2019/06/21/Helium-shortage-why-the-worlds-supply-is-drying-up.html

CCOHS. (2020). *OSH Answers Fact Sheets*. Retrieved May 2020, from https://www.ccohs.ca/oshanswers/chemicals/chem_profiles/methane.html

Centers for Disease Control and Prevention. (2012, May 18). *Principles of Epidemiology in Public Health Practice, Third Edition; Section 7: Measures of Spread*. Retrieved April 2020, from Centers for Disease Control and Prevention (CDC): https://www.cdc.gov/csels/dsepd/ss1978/lesson2/section7.html#ALT27

Chamberlain, P. (2014, May 8). *Are We Running Out of Helium*. Retrieved November 2019, from Leak Testing Specialists Inc: http://www.leaktestingspec.com/NewsEvents/Details?id=1259

Cotton, F., & Wilkinson, G. (1988). Advanced Inorganic Chemistry (5th ed.).

CRAN. (2018, March 25). *Package 'randomForest'*. Retrieved June 2020, from CRAN.R-Project: https://cran.r-project.org/web/packages/randomForest/randomForest.pdf

Daniels, K., & LeBoeuf, C. (2019). Quarter-Turn Control Valve Design for Low Emissions to Pass API 641 and ISO 15848 Testing. *Fugitive Emissions Journal*. Retrieved June 2020

Dickson, L. (2020, May 18). *Diffusion*. Retrieved June 2020, from LibreTexts: Chemistry: https://chem.libretexts.org/Bookshelves/Physical_and_Theoretical_Chemistry_Textbook_Maps/Supplemental_Modules_(Physical_and_Theoretical_Chemistry)/Kinetics/Diffusion

Engineering ToolBox. (2001). *Air - Diffusion Coefficients of Gases in Excess of Air*. Retrieved July 2020, from https://www.engineeringtoolbox.com

Gelman, A., Goodrich, B., Gabry, J., & Vehtari, A. (2019). R-squared for Bayesian Regression Models. *The American Statistician*, 307-309. Retrieved May 2020

Glimo. (n.d.). *What is an interaction effect?* Retrieved May 2020, from Glimo: http://glimo.vub.ac.be/downloads/interaction.htm

Grace-Martin, K. (2020). *Can a Regression Model with a Small R-squared Be Useful?* Retrieved May 2020, from The Analysis Factor: https://www.theanalysisfactor.com/small-r-squared/

Hastie, T., Tibshirani, R., & Friedman, J. (2009). Chapter 15.3.4: Random Forests. In T. Hastie, *The elements of statistical learning* (pp. 587-604). New York: Springer. doi:10.1007/b94608_15

Hope, K. (2019, September 17). *Helium Shortage: Prices just keep going up and up.* Retrieved November 2019, from BBC: https://www.bbc.com/news/business-49715838

International Organization for Standardization. (2015). Industrial valves — Measurement, test and qualification procedures for fugitive emissions — Part 1: Classification system and qualification procedures for type testing of valves (ISO Standard No. 15848-1). Retrieved October 2019, from http://www.iso.org/iso/catalogue_detail

IPCC Guidelines for National Greenhouse Gas Inventories. (2019). CHAPTER 4: FUGITIVE EMISSIONS. Retrieved 2019, from https://www.ipcc-nggip.iges.or.jp/public/2019rf/pdf/2_Volume2/19R_V2_4_Ch04_Fugitive_Emissions.pdf

Kazeminia, M., & Bouzid, A.-H. (2015). Prediction of gas flow in compressible and low permeability porous media. *Journal of Fluid Science and Technology*, 1-12. Retrieved June 2020

Kuhn, M. (2019, 3 27). *The caret Package*. Retrieved April 2020, from Github: http://topepo.github.io/caret/

Murphy, H. (2019, May 16). *The Global Helium Shortage is Real, but Don't Blame Party Balloons*. Retrieved November 2019, from https://www.nytimes.com/2019/05/16/science/Helium-shortage-party-city.html

Nau, R. (2019). *What's a good value for R-squared?* (Duke University: Fuqua School of Business) Retrieved May 2020, from Statistical forecasting: notes on regression and time series analysis: https://people.duke.edu/~rnau/rsquared.htm

NIST/SEMATECH e-Handbook of Statistical Methods. (2012, April). *Anderson-Darling and Shapiro-Wilk tests*. Retrieved November 2019, from Engineering Statistics Handbook: https://doi.org/10.18434/M32189

NIST/SEMATECH e-Handbook of Statistical Methods. (2012, April 10). *One-Way ANOVA overview*. Retrieved November 2019, from Engineering Statistics Handbook: https://doi.org/10.18434/M32189

NIST/SEMATECH e-Handbook of Statistical Methods. (2012, April). *What are outliers in the data?* Retrieved July 2020, from Engineering Statistics Handbook: https://doi.org/10.18434/M32189

Patil, N. (2013). Effective solutions to beat Fugitive Emission. *Emission Control*, 1-3. Retrieved June 2020

Royal Society of Chemistry. (2020). *Carbon.* Retrieved from Royal Society of Chemistry: https://www.rsc.org/periodic-table/element/6/carbon

Royal Society of Chemistry. (2020). *Helium*. Retrieved April 2020, from Royal Society of Chemistry: https://www.rsc.org/periodic-table/element/2/helium

Royal Society of Chemistry. (2020). *Periodic Table: Argon*. Retrieved April 2020, from Royal Society of Chemistry: https://www.rsc.org/periodic-table/element/18/argon

Saylor Academy. (2012). *Introductory Statistics.* Retrieved November 2019, from https://saylordotorg.github.io/text_introductory-statistics/s15-03-f-tests-for-equality-of-two-va.html

Simon Fraser University. (2011). *Inter-Quartile Range, Outliers, Boxplots.* Retrieved November 2019, from Simon Fraser University - California: https://www.sfu.ca/~jackd/Stat203_2011/Wk02_1_Full.pdf

Thomas Jefferson National Accelerator Facility - Office of Science Education. (2020). *The Element Helium*. Retrieved 2020, from JLab Science Education: https://education.jlab.org/itselemental/ele002.html

TQC. (n.d.). *Guide to Helium Leak Testing*. Retrieved May 2020, from TQC Automation & Test Solutions: https://www.tqc.co.uk/our-services/leak-testing/helium/guide-to-helium-leak-testing/

U.S. National Library of Medicine. (n.d.). *Methane*. Retrieved April 2020, from ToxTown: https://toxtown.nlm.nih.gov/chemicals-and-contaminants/methane

VAC AERO International. (2013, August 13). *Shaft Seals for Rotating Shafts*. Retrieved May 2020, from VAC AERO International Inc: https://vacaero.com/information-resources/vacuum-pump-technology-education-and-training/9363-shaft-seals-for-rotating-shafts.html

Vacuum Instruments Corporation. (n.d.). *Argon Leak Detection*. Retrieved from Vacuum Instruments Corporation: Leak Detection Solutions: https://vicleakdetection.com/page/1057/argon-leak-detection

Vishik, I. (n.d.). Why We Are Running Out of Helium and What We Can Do About It. *Forbes*. Retrieved November 2019, from https://www.forbes.com/sites/quora/2016/01/01/why-we-are-running-out-of-helium-and-what-we-can-do-about-it/#45ebe12357ad

World Health Organization. (n.d.). *Climate change and human health - risks and responses. Summary*. Retrieved 2020, from World Health Organization: https://www.who.int/globalchange/summary/en/index13.html

Zheng, J. (2019, March 19). *What is box plot?* Retrieved June 2020, from https://jingwen-z.github.io/what-is-box-plot/

11 Appendices - Supplementary information

Appendix A – Issues encountered and Challenges

Combination of issues both Milad Najafbeygi and Abigail Hovorka experienced

1) Prior to writing this, my experience with machine learning was minimal and caret is a machine learning technique. Teaching myself this package was a significant challenge. My previous experience with R studio and my basic statistical knowledge helped a lot, but I did not know the basics of machine learning before this.
2) My VBA code went through many iterations. In the beginning it took nearly 30 minutes to run. By the end I had it running in 3 seconds, but it took 4 or 5 different attempts at writing the program.
3) It was challenging to seal the equipment when working with Helium. Helium Leakage was often very high with select barrier configurations to generate back-to-back Helium and Argon leak-rates and meaningful ratios.
4) We used a liquid Nitrogen setup to cool the experimental chamber to below ambient and up to -46˚C. Introduction of liquid Nitrogen (at ~ -198.6˚C) led to many valve and connection failures during the course of this effort.
5) Reducing temperature to -46˚C took more than 18 hours. Though it was setup on a closed loop controller, the system needed to be watched to prevent overshooting the setpoint, staging the temperature reduction in 2 steps turned out to be more practical.
6) After each experiment with one gas, a delay of at least 24 hours was needed to fully depressurize the system and remove any trapped gas in the seal barrier. Trapped gas tends to slow down the permeation of the second gas in back-to-back experiments and affect leakage results.

7) The available seal barrier stack did not work well after any thermal cycles. Hence, barriers needed to be changed for experiments between high and low temperatures.
8) With each new assembly and at the start of a test, the system needed to be leak tested.

Appendix B – Individual raw data plots for Methods 1 and 2

These plots are courtesy of Milad Najafbeygi

Examples of raw data are presented in this section. For a given test pressure and temperature, pressure increase at the low-pressure collection side show the leakage behavior. By applying ideal gas law along with the volume of the collection sides and the temperature, mass leakage flow rate was estimated.

Figure 42 is an example of the high and low-pressure variation graphs at a set 0˚C low temperature and 6250 psi. The black dotted line is high pressure and the red and blue lines are low pressure.

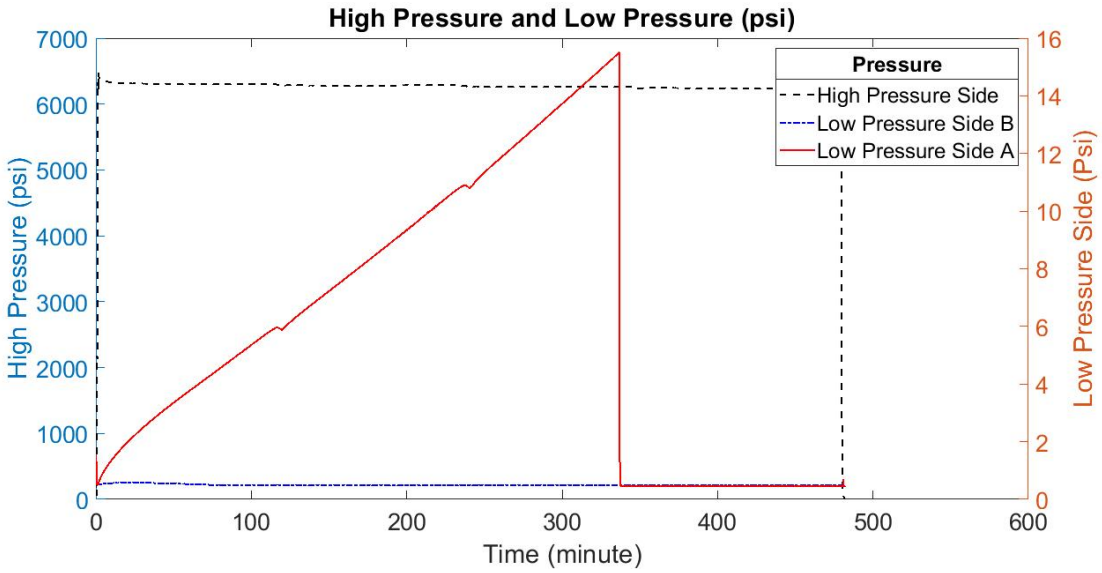


Figure 42 - low-pressure changes at 0˚C and 6250 psi for Argon

Figures 43-48 are results of leak rate respectively at -46˚C, -29˚C and 0˚C. The red line is a low pressure and the green line is showing temperature changes.

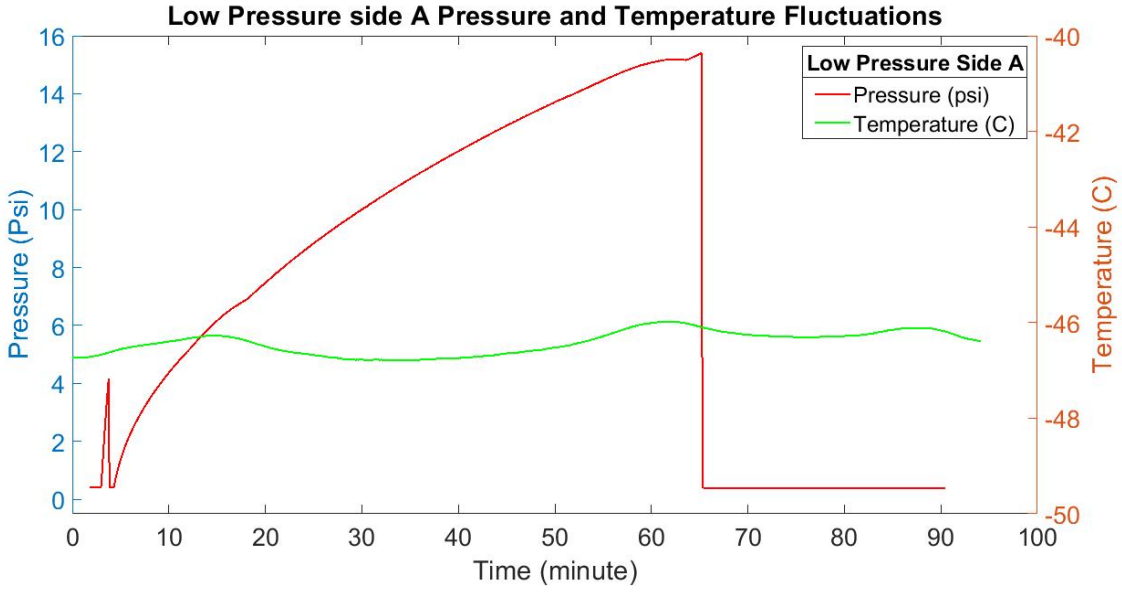


Figure 43 - plastic seals at -46˚C and 6250 psi for Argon

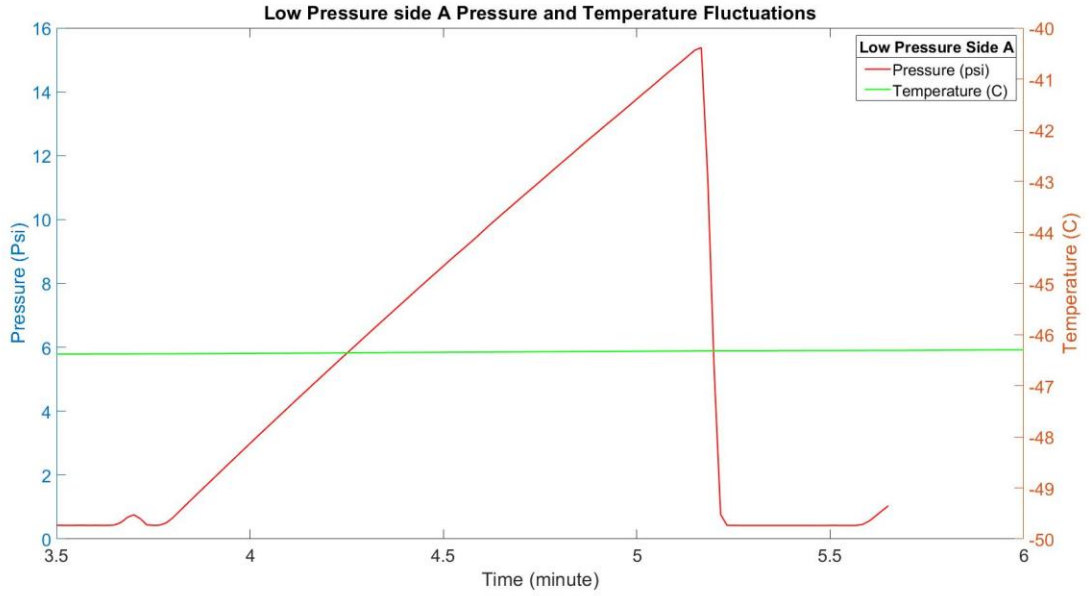


Figure 44 - plastic seals at -46˚C and 6250 psi for Helium

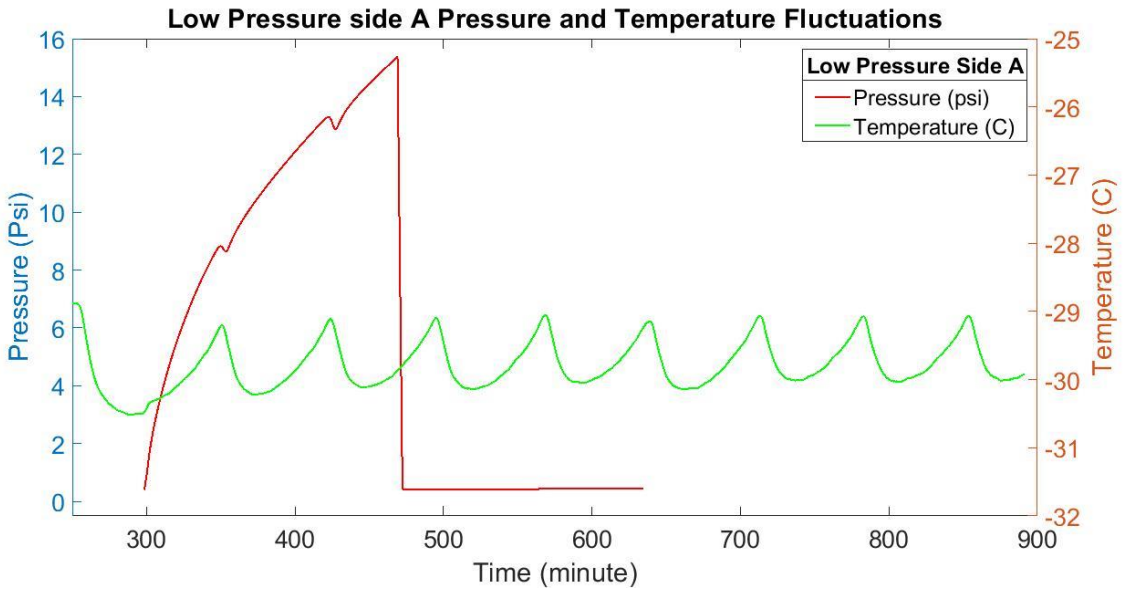


Figure 45 - plastic seals at -29˚C and 6250 psi for Argon

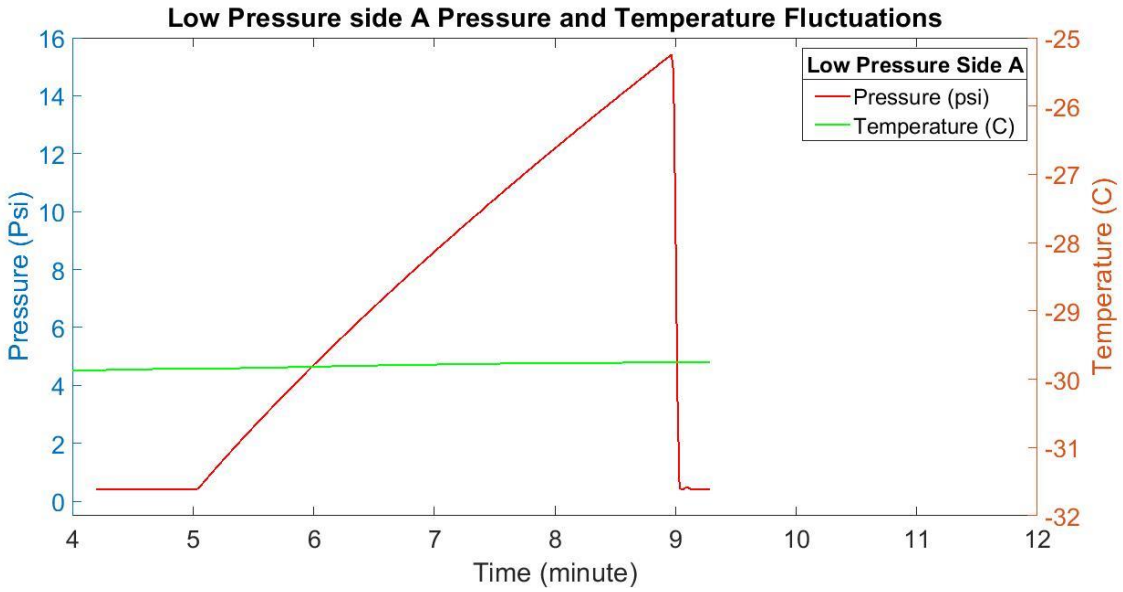


Figure 46 - plastic seals at -29˚C and 6250 psi for Helium

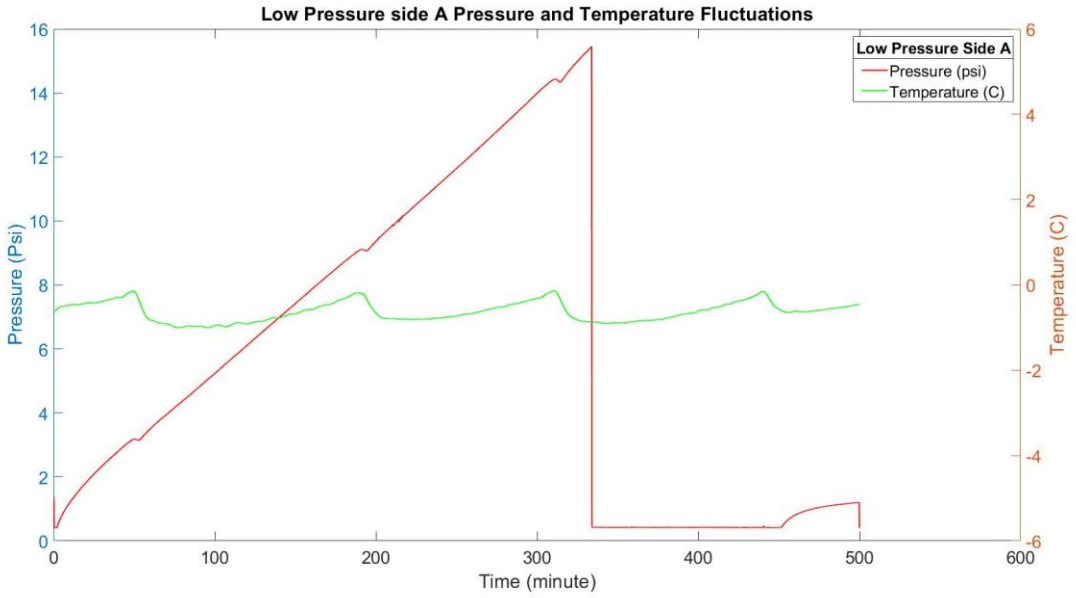


Figure 47 - plastic seals at 0˚C and 6250 psi for Argon

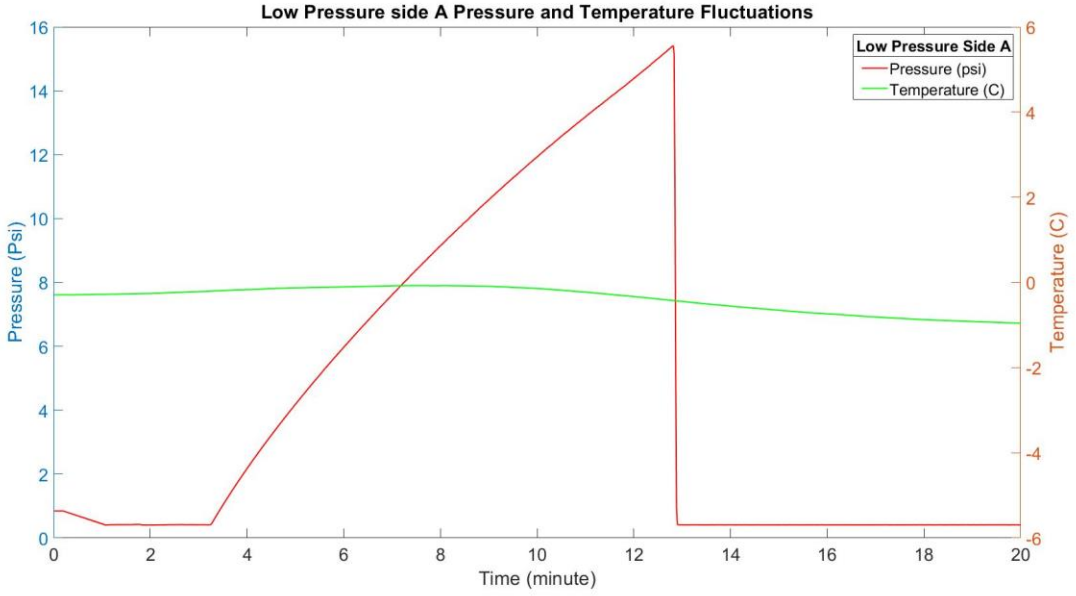


Figure 48 - plastic seals at 0˚C and 6250 psi for Helium

Appendix C – Individual Plots for Random Forest Modeling Data

X-axis units: Time (sec)

Y-axis units: Pressure (psi)

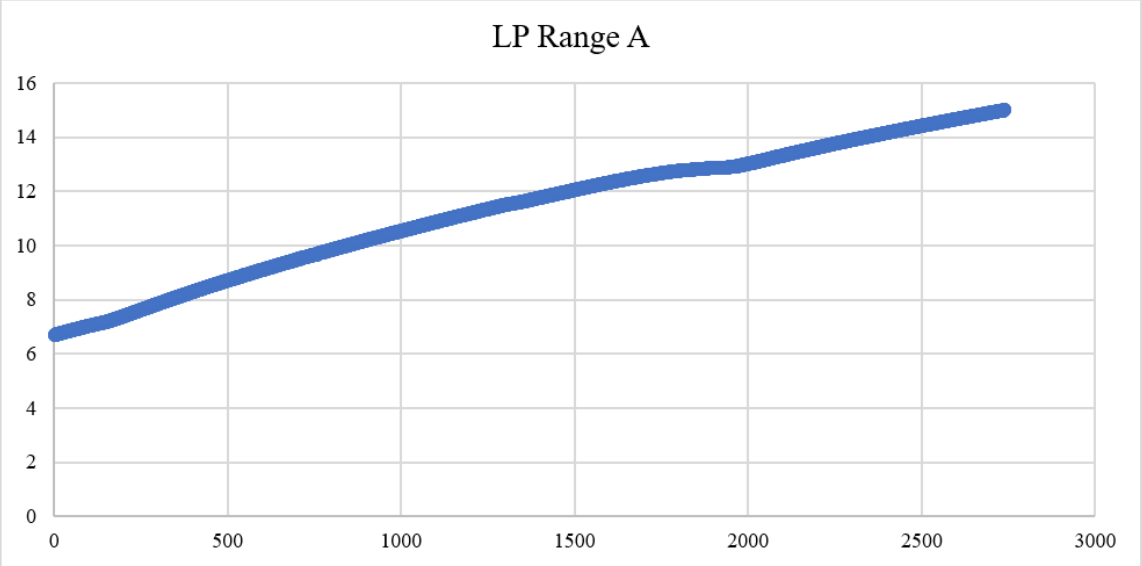


Figure 49 - 6250psi Run 1 -46˚C Ar

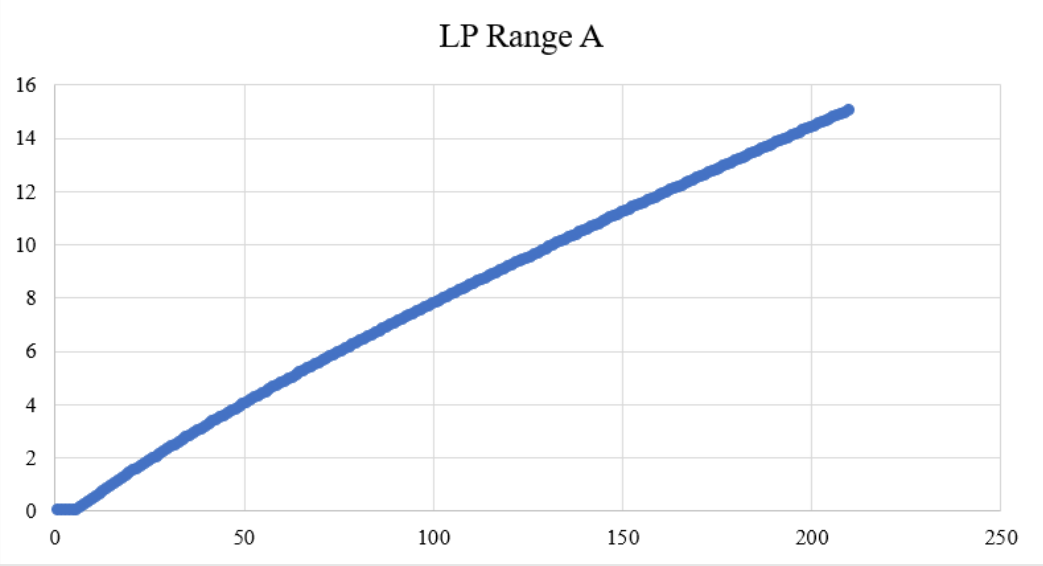


Figure 50 - 2250psi Run 2 -46˚C He

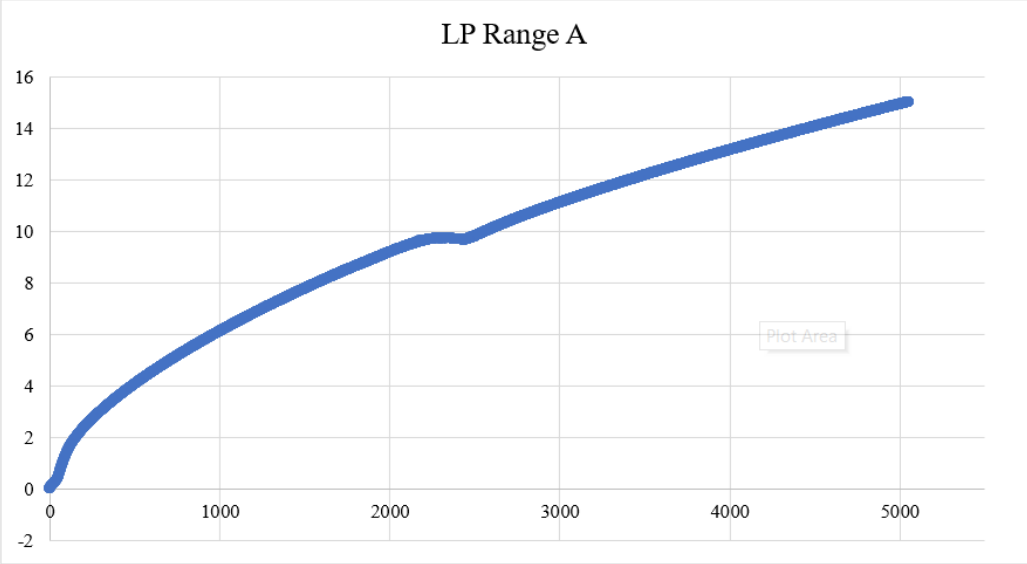


Figure 51 - 2250psi Run 1 -29˚C Ar

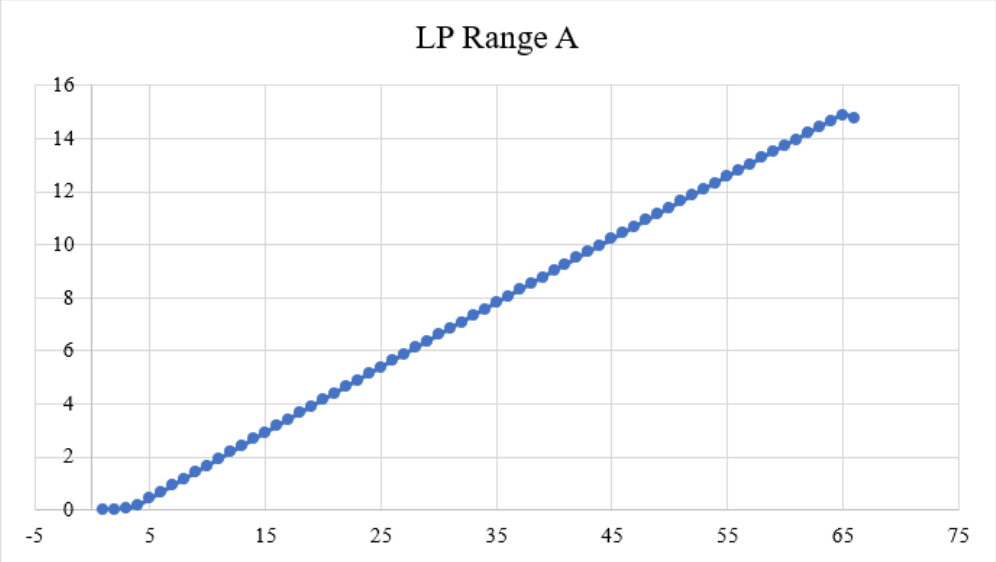


Figure 52 - 600psi Run 1 -29˚C He

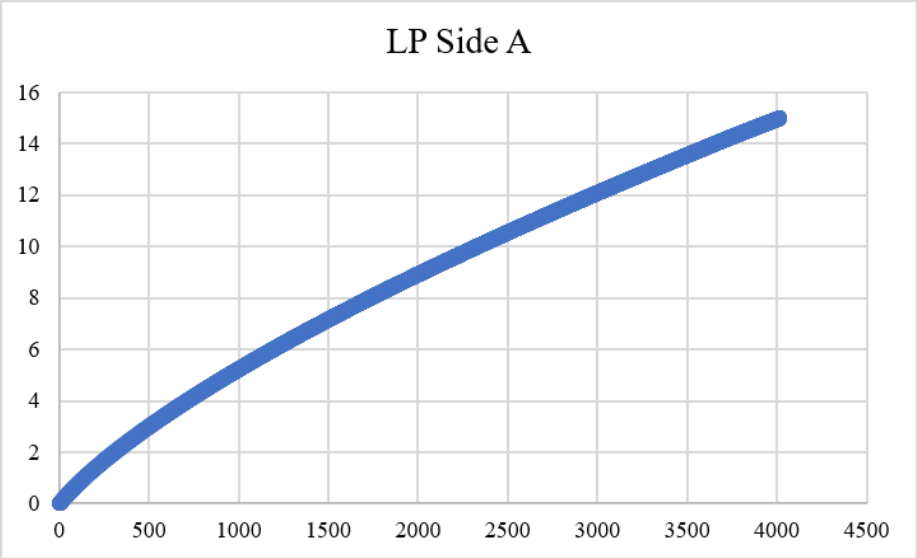


Figure 53 - 600psi Run 2 0˚C Ar

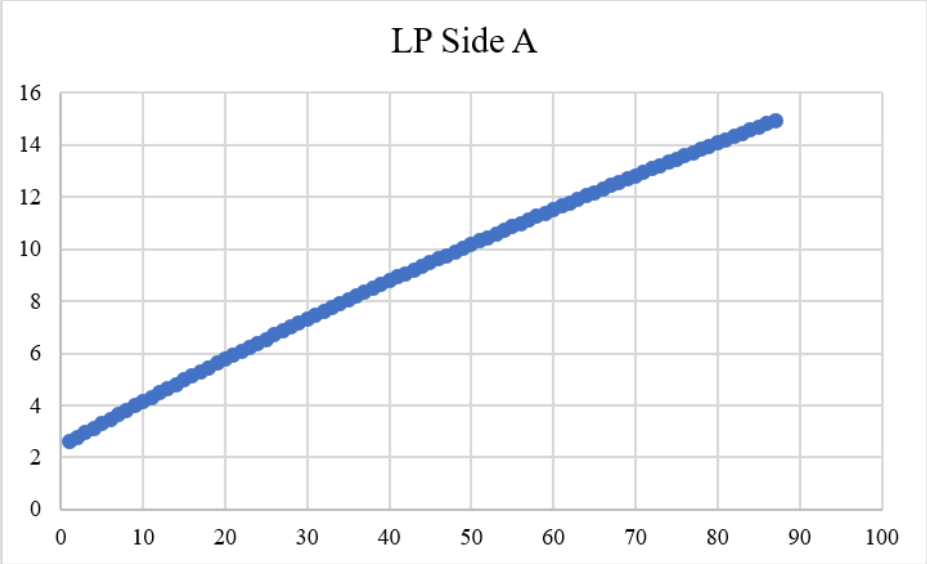


Figure 54 - 10ksi Run 5 0˚C He

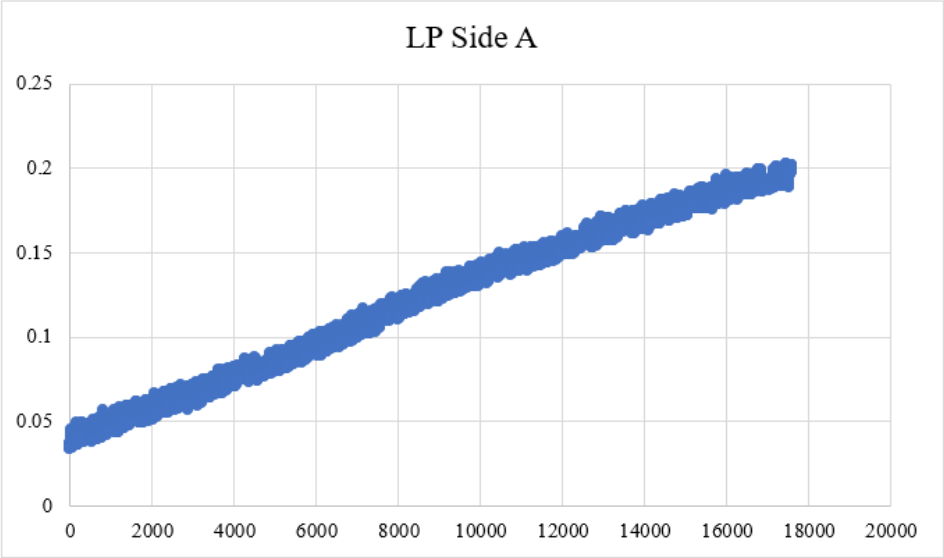


Figure 55 - 3750psi Run 3 25˚C Ar

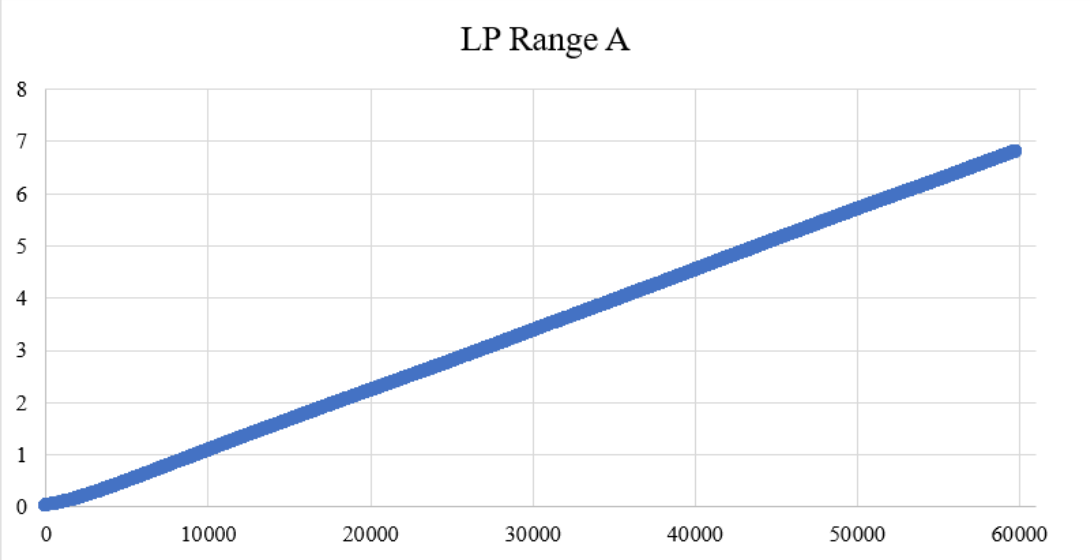


Figure 56 - 10ksi Run 1 25˚C He

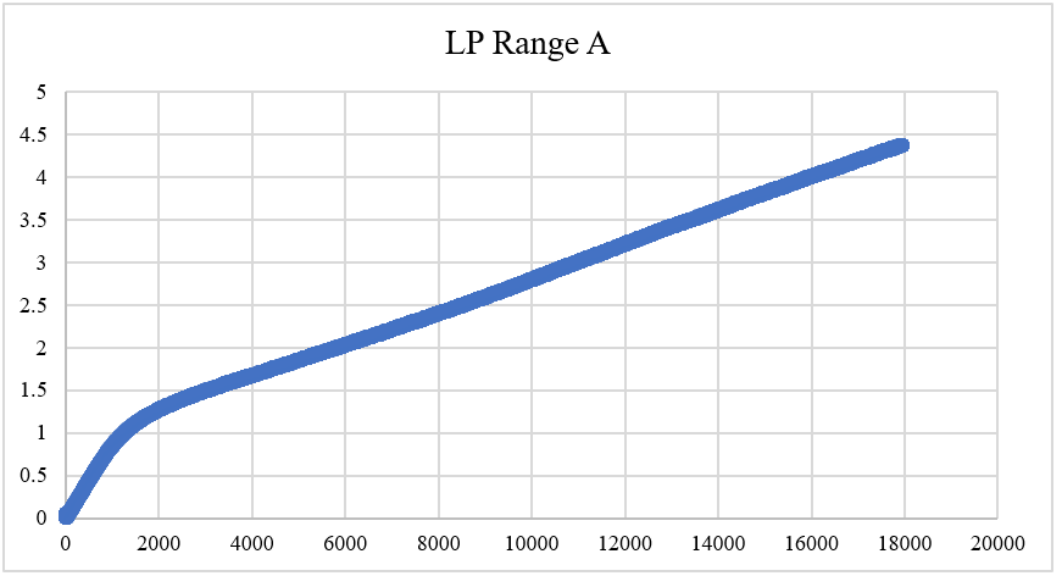


Figure 57 - 600psi Run 1 121˚C Ar

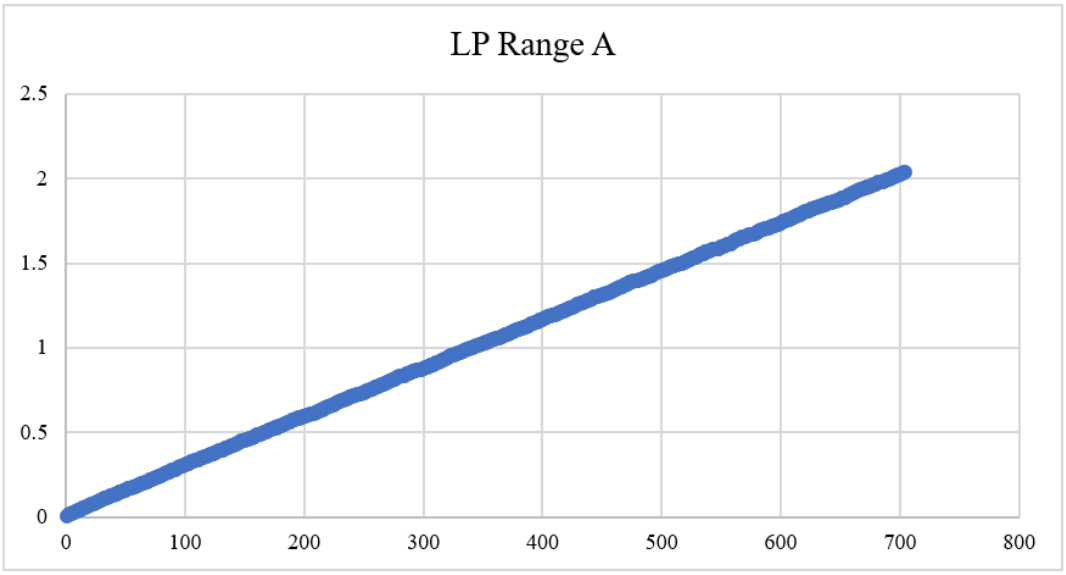


Figure 58 - 10ksi Run 1 121˚C He

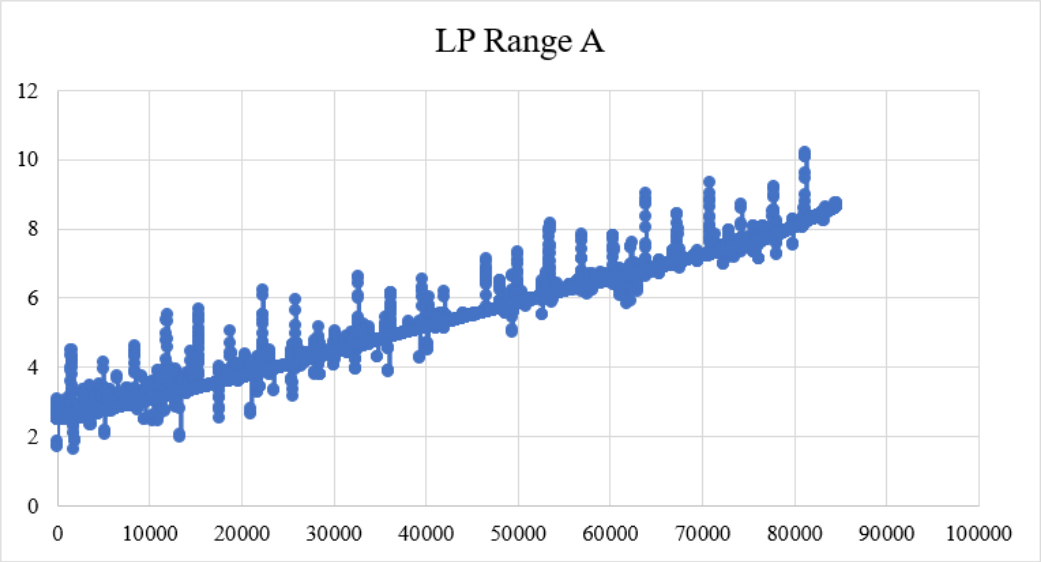


Figure 59 - 10ksi Run 1 204˚C Ar

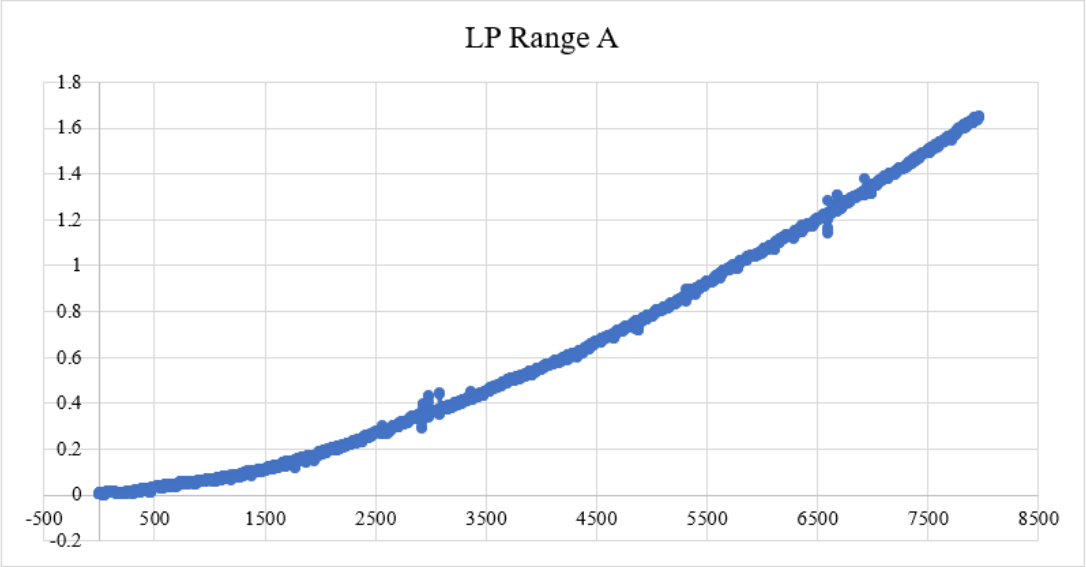


Figure 60 - 10ksi Run 1 204˚C He

Appendix D – Seal Barrier

Figure 61 shows the picture a typical spring energized plastic barrier seal and Figure 62 shows a typical V-ring stack. These ring geometries were used to build different configurations for the experiments and obtain back-to-back leakage results.

Figure 61 - Picture of a spring energized plastic seal ring

Figure 62 - Schematic of a generic plastic V-stack

Appendix E – Glossary of relevant terms

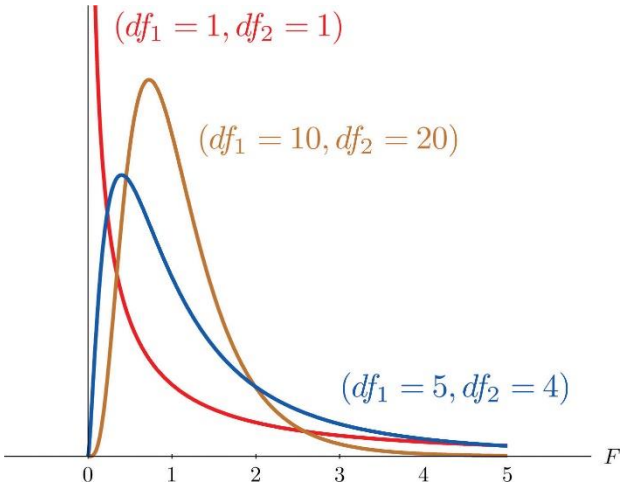
σ – sigma; standard deviation; measure of variation from the median of a data set; helps to measure variation and dispersion of a data set

Å – Ånström; a unit of length equal to 10^{-10}m

Analysis of Variance (ANOVA) – a statistical analysis tool that splits an observed aggregate variability found inside a data set into two parts: systematic (influential on the data) factors and random (error) factors (investopedia.com/terms/a/anova.asp)

df – degrees of freedom; the number of data points that can be assigned to a particular distribution

F-distribution – Each point on the distribution requires a pair of parameters in the form of degrees of freedom



(Saylor Academy, 2012)

F-statistic – a test statistic on the F-distribution; a ratio of explainable and unexplainable variance

Hyperparameter – a parameter whose value controls the machine learning process and accuracies

Interquartile range (IQR) – a measure of where the bulk of the values lie; measurement between the first and third quartile (NIST/SEMATECH e-Handbook of Statistical Methods, 2012)

Mass flow – the movement of fluids down a temperature or pressure gradient

Volumetric flow – movement of volume unit

mbar – millibar; 1 mbar = 1×10^{-3}bar; bar is a metric unit of pressure

Median – middle number in an ascending sorted data set

MS – mean squares; are the sum of squares divided by their respective degrees of freedom; provides an understanding of population variance

Normality probability plot – a graphical technique to assess whether a data set is approximately normally distributed; data are plotted against a theoretical normal distribution to observe behavior

ppmv – parts per million volume; air pollutant concentration

SS – sum of squares; the squared sum of each data point's variation from the mean; allows the computation of variance displayed in the ANOVA Table

Shapiro-Wilk test – a normality test for data sets; produces a W-test statistic using the given data set

Significance F-value – the F-statistic divided by the respective degrees of freedom

Test statistic – a numerical description of the outcome that is calculated using sample data

Quartiles – the division of a data set into four intervals based on the values of each point and how they compare to the rest of the data set; divisions are based on the spread of the data

W-test statistic – using a prescribed data set; $W = \frac{\left(\sum_{i=1}^{n} a_i x_{(i)}\right)^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$

Appendix F – VBA Code

```
Sub count()
Dim ws As Worksheet
Dim dynamicCount1 As Long
Dim dynamicCount2 As Long
Dim dynamicCount As Long
Dim staticCount As Long
Dim pasteCount As Long
Dim neededRange As Range
Dim tempRange As Range
Dim overallRange As Range
For Each ws In Worksheets
    count_LP = ws.Cells(Rows.count, "A").End(xlUp).Row
    ws.Range("N1:P100").ClearContents
    ws.Range("N1") = "LP A Temps"
    ws.Range("O1") = "LP A Slopes"
    ws.Range("P1") = "Overall LP A Slope"
    splitCount1 = Application.WorksheetFunction.RoundDown(count_LP / 15, 0) 'change
the value diving count_LP to control how many points we want
    dynamicCount = splitCount1 + 1
    staticCount = 2
    pasteCount = 2
    ws.Range("t1") = count_LP
```

```
    ws.Range("t2") = splitCount1
    ws.Range("t4") = dynamicCount
    ws.Range("t5") = staticCount
    Set overallRange = ws.Range(ws.Cells(2, 3), ws.Cells(count_LP, 3))
    overallSlope = Application.WorksheetFunction.LinEst(overallRange)
    ws.Range("P2") = overallSlope
    Do While Not IsEmpty(ws.Range("A" & dynamicCount))
        Set neededRange = ws.Range(ws.Cells(staticCount, 3), ws.Cells(dynamicCount, 3))
        ratioSlope = Application.WorksheetFunction.LinEst(neededRange)
        ws.Range("O" & pasteCount) = ratioSlope
        'Set tempRange = ws.Range(ws.Cells(staticCount, 5), ws.Cells(dynamicCount, 5))
        'averageTemp = Application.WorksheetFunction.Average(tempRange)
        'ws.Range("N" & pasteCount) = averageTemp
        dynamicCount = dynamicCount + splitCount1
        staticCount = staticCount + splitCount1
        pasteCount = pasteCount + 1
    Loop
Next ws
End Sub
```

Appendix G – R Code

1. R Code for first modelList

```
install.packages(c('caret', 'skimr', 'RANN', 'randomForest', 'fastAdaboost', 'gbm', 'xgboost', 'caretEnsemble', 'C50', 'ea
rth'))

library(dplyr)
library(ggplot2)
library(PerformanceAnalytics)
library(ggthemes)
library(corrplot)
library(car)
library(psych)
library(caret)
library(caretEnsemble)
library(doParallel)

#created this variable so that I could keep changing the data set with my new ones and not have to copy and paste a
million times to parRF the data set
current_data <- finalData1

#looking at the data and making sure there are not any gaps I need to deal with
dim(current_data)
summary(current_data$Ratio)
anyNA(current_data)

#creating a simple linear model so that I can do a Variance Inflation Factor check
#VIF values over 10 suggest multicollinearity is present and we should get rid of the features causing it
simple_lm <- lm(Ratio ~ ., data = current_data)
vif(simple_lm)

#shuffle the data so that we can ensure randomization
set.seed(123) #ensures reproducability
data_rand <- current_data[sample(1:nrow(current_data)), ]
dim(data_rand)

#creating predictors
X = data_rand[, -3]
Y = data_rand[, 3]
#Y<- as.factor(Y)
#checking to make sure they are good
str(X)
str(Y)
#now to split these into train and test sets
#using 80% to train and the remaining 20% to test
set.seed(123)
part.index <- createDataPartition(data_rand$Ratio,
                                  p = 0.8,
                                  list = FALSE)
#creating the inputs and outputs for the training and testing
X_train <- X[part.index, ]
X_test <- X[-part.index, ]
Y_train <- Y[part.index]
Y_test <- Y[-part.index]
```

```
#checking to make sure they are good
str(X_train)
str(X_test)
str(Y_train)
str(Y_test)
registerDoParallel(4)
getDoParWorkers()

#cross validation
set.seed(123)
my_control <- trainControl(method = 'cv',
                           number = 5,
                           savePredictions = 'final',
                           allowParallel = TRUE,
                           index = createFolds(Y_train,5))

#TRAINING THE MODEL USING OUR X&Y TRAIN SETS
#testing multiple model methods using methodList
set.seed(222)
model_list <- caretList(X_train,
                        Y_train,
                        trControl = my_control,
                        methodList = c('treebag', 'svmRadial', 'rf', 'parRF', 'xgbLinear', 'knn', 'lm'),
                        tuneList = NULL,
                        continue_on_fail = FALSE,
                        preProcess = c('center','scale'))
#I like to see the models to make sure nothing suspect happened in the making of them
model_list$svmRadial
model_list$treebag
model_list$rf
model_list$parRF
model_list$xgbLinear
model_list$knn
model_list$lm

#now I want to see which model produced the smallest RMSE
options(digits = 3)
model_results <- data.frame(
  treebag = min(model_list$treebag$results$RMSE),
  SVM = min(model_list$svmRadial$results$RMSE),
  RF = min(model_list$rf$results$RMSE),
  parRF = min(model_list$parRF$results$RMSE),
  XGBL = min(model_list$xgbLinear$results$RMSE),
  KNN = min(model_list$knn$results$RMSE),
  LM = min(model_list$lm$results$RMSE)
)
print(model_results)
#let's resample and plot because visuals are always nice when explaining things to people
resamples <- resamples(model_list)
dotplot(resamples, metric = 'RMSE')

#here we can see if any of the models are highly correlated with another
modelCor(resamples)

#training an ensemble of the models, which is going to perform a linear combination with all of them
set.seed(222)
```

```
ensemble_1 <- caretEnsemble(model_list,
                            metric = 'RMSE',
                            trControl = my_control)
summary(ensemble_1)
plot(ensemble_1)

#doing another ensemble with caretStack
set.seed(222)
ensemble_2 <- caretStack(model_list,
                         method = 'glmnet',
                         metric = 'RMSE',
                         trControl = my_control)
print(ensemble_2)

#TIME TO EVALUATE PERFORMANCE OF THE MODELS OVER UNSEEN DATA, which is in our test data s
et
#first predict the test set with each model then compute RMSE
# PREDICTIONS
pred_treebag <- predict.train(model_list$treebag, newdata = X_test)
pred_svm <- predict.train(model_list$svmRadial, newdata = X_test)
pred_rf <- predict.train(model_list$rf, newdata = X_test)
pred_parRF <- predict.train(model_list$parRF, newdata = X_test)
pred_xgbL <- predict.train(model_list$xgbLinear, newdata = X_test)
pred_knn <- predict.train(model_list$knn, newdata = X_test)
pred_lm <- predict.train(model_list$lm, newdata = X_test)
predict_ens1 <- predict(ensemble_1, newdata = X_test)
predict_ens2 <- predict(ensemble_2, newdata = X_test)
# RMSE
pred_RMSE <- data.frame(ensemble_1 = RMSE(predict_ens1, Y_test),
                        ensemble_2 = RMSE(predict_ens2, Y_test),
                        treebag = RMSE(pred_treebag, Y_test),
                        SVM = RMSE(pred_svm, Y_test),
                        RF = RMSE(pred_rf, Y_test),
                        parRF = RMSE(pred_parRF, Y_test),
                        XGBL = RMSE(pred_xgbL, Y_test),
                        KNN = RMSE(pred_knn, Y_test),
                        LM = RMSE(pred_lm, Y_test)
                        )
print(pred_RMSE)

#show the correlation score for each test
pred_cor <- data.frame(ensemble_1 = cor(predict_ens1, Y_test),
                       ensemble_2 = cor(predict_ens2, Y_test),
                       treebag = cor(pred_treebag, Y_test),
                       SVM = cor(pred_svm, Y_test),
                       RF = cor(pred_rf, Y_test),
                       parRF = cor(pred_parRF, Y_test),
                       XGBL = cor(pred_xgbL, Y_test),
                       KNN = cor(pred_knn, Y_test),
                       LM = cor(pred_lm, Y_test)
                       )
print(pred_cor)

#TUNING
#grid <- expand.grid(
#                    sigma = c(0.01),
```

```
#                 mtry = c(2),
#                 nrounds=c(100),
#                 lambda=c(2),
#                 alpha = c(0.1),
#                 eta=c(5)
#)
xgb_grid_1 <- expand.grid(
  nrounds= 1500,
  eta=c(0.01,0.001,0.0001),
  lambda = 2,
  alpha =0.0
)


set.seed(222)
model_list <- caretList(X_train,
                        Y_train,
                        trControl = my_control,
                        methodList = c('treebag', 'svmRadial', 'rf', 'parRF', 'xgbLinear','knn', 'lm'),
                        tuneList = list
                          item1 =caretModelSpec(method = 'rf', tuneGrid=data.frame(.mtry=2)),
                          item2  = caretModelSpec(method = 'xgbLinear', tuneGrid = xgb_grid_1)
                          ),
                        continue_on_fail = FALSE,
                        preProcess = c('center','scale')
                        )
model_list$svmRadial
model_list$treebag
model_list$rf
model_list$parRF
model_list$xgbLinear
model_list$knn
model_list$lm

#now I want to see which model produced best R squared
options(digits = 3)
model_results <- data.frame(
  treebag = min(model_list$treebag$results$RMSE),
  SVM = min(model_list$svmRadial$results$RMSE),
  RF = min(model_list$rf$results$RMSE),
  parRF = min(model_list$parRF$results$RMSE),
  XGBL = min(model_list$xgbLinear$results$RMSE),
  KNN = min(model_list$knn$results$RMSE),
  LM = min(model_list$lm$results$RMSE)
)
print(model_results)
#let's resample and plot
resamples <- resamples(model_list)
dotplot(resamples, metric = 'RMSE')

#here we can see if any of the models are highly correlated with another
modelCor(resamples)

#training an ensemble of the models, which is going to perform a linear combination with all of them
set.seed(222)
ensemble_1 <- caretEnsemble(model_list,
```

```
                    metric = 'RMSE',
                    trControl = my_control)
summary(ensemble_1)
plot(ensemble_1)

#doing another ensemble with caretStack
set.seed(222)
ensemble_2 <- caretStack(model_list,
                    method = 'glmnet',
                    metric = 'RMSE',
                    trControl = my_control)
print(ensemble_2)

#TIME TO EVALUATE PERFORMANCE OF THE MODELS OVER UNSEEN DATA, which is in our test data s
et
#first predict the test set with each model then compute RMSE
# PREDICTIONS
pred_treebag <- predict.train(model_list$treebag, newdata = X_test)
pred_svm <- predict.train(model_list$svmRadial, newdata = X_test)
pred_rf <- predict.train(model_list$rf, newdata = X_test)
pred_parRF <- predict.train(model_list$parRF, newdata = X_test)
pred_xgbL <- predict.train(model_list$xgbLinear, newdata = X_test)
pred_knn <- predict.train(model_list$knn, newdata = X_test)
pred_lm <- predict.train(model_list$lm, newdata = X_test)
predict_ens1 <- predict(ensemble_1, newdata = X_test)
predict_ens2 <- predict(ensemble_2, newdata = X_test)
# RMSE
pred_RMSE2 <- data.frame(ensemble_1 = RMSE(predict_ens1, Y_test),
                    ensemble_2 = RMSE(predict_ens2, Y_test),
                    treebag = RMSE(pred_treebag, Y_test),
                    SVM = RMSE(pred_svm, Y_test),
                    RF = RMSE(pred_rf, Y_test),
                    parRF = RMSE(pred_parRF, Y_test),
                    XGBL = RMSE(pred_xgbL, Y_test),
                    KNN = RMSE(pred_knn, Y_test),
                    LM = RMSE(pred_lm, Y_test)
)
print(pred_RMSE2)
```

2. R Code for Creating Random Forest and Testing ISO Conditions

```
#specifically rf code
library(dplyr)
library(randomForest)
library(caret)
library(e1071)

current_data <- finalData1

set.seed(100)

trainRowNumbers <- createDataPartition(current_data$Ratio, p=0.8, list=FALSE)

train_data <- current_data[trainRowNumbers,]
```

```
test_data <- current_data[-trainRowNumbers,]


x = train_data[, 1:2]
y= train_data$Ratio

a = test_data[, 1:2]
b= test_data$Ratio

HERE IS A CARETLIST TO DEMONSTRATE WHY I WANT TO USE RANDOMFOREST
#cross validation
set.seed(123)
my_control <- trainControl(method = 'cv',
number = 5,
                           savePredictions = 'final',
                           allowParallel = TRUE,
                           index = createFolds(Y_train,5))

#TRAINING THE MODEL USING OUR X&Y TRAIN SETS
#testing multiple model methods using methodList
set.seed(222)
model_list <- caretList(x,
                        y,
                        trControl = my_control,
                        methodList = c('rf', 'lm', 'ctree'),
                        tuneList = NULL,
                        continue_on_fail = FALSE,
                        preProcess = c('center','scale')
                        )
#I like to see the models to make sure nothing suspect happened in the making of them
model_list$rf
model_list$lm
model_list$ctree


#rf is the winner~~~~~~~~~~~~~~~~~~~~~~~~~~

# Create the forest.
output.forest <- randomForest(Ratio ~ Temp + Pressure,
                              data = current_data,
                              ntree = 10,
                              mtry = c(1:10),
                              maxnodes = NULL,
                              importance = TRUE)
# View the forest results.
print(output.forest)

# Define the control
trControl <- trainControl(method = "cv",
                          number = 250,
                          search = "grid")


set.seed(1234)
# Run the model
rf_default <- train(train_data[, 1:2],
```

```
                train_data$Ratio,
                data = train_data,
                method = "rf",
                metric = "RMSE",
                trControl = trControl,
                preProcess = c('center','scale'),
                tuneLength = 3)
# Print the results
print(rf_default)

#prediction <- predict(rf_default, newdata= a)
#print(prediction)

varImp(rf_default)

###################################
#let's do better
set.seed(1234)
tuneGrid <- expand.grid(.mtry = c(1: 3))
rf_mtry <- train(train_data[, 1:2],
                train_data$Ratio,
                data = train_data,
                method = "rf",
                metric = "RMSE",
                tuneGrid = tuneGrid,
                trControl = trControl,
                importance = TRUE,
                preProcess = c('center','scale'),
                tuneLength = 3)
print(rf_mtry)

#can store it and use it when needed to tune other parameters
max(rf_mtry$results$Rsquared)

#best value of mtry here
best_mtry <- rf_mtry$bestTune$mtry
best_mtry

#let’s find maxnodes
store_maxnode <- list()
tuneGrid <- expand.grid(.mtry = best_mtry)
for (maxnodes in c(1:20)) {
  set.seed(1234)
  rf_maxnode <- train(train_data[, 1:2],
                      train_data$Ratio,
                      method = "rf",
                      metric = "RMSE",
                      tuneGrid = tuneGrid,
                      trControl = trControl,
                      importance = TRUE,
                      nodesize = 16,
                      maxnodes = maxnodes,
                      ntree = 300,
                      preProcess = c('center','scale'),
                      tuneLength = 3)
  current_iteration <- toString(maxnodes)
```

```
  store_maxnode[[current_iteration]] <- rf_maxnode
}
results_node <- resamples(store_maxnode)
summary(results_node)


#search best ntrees using best maxnode
store_maxtrees <- list()
tuneGrid <- expand.grid(.mtry = best_mtry)
for (ntree in c(5, 50, 100, 150, 200, 250, 300, 350, 400, 450, 500, 550, 600, 800, 1000, 2000, 5000)) {
  set.seed(5678)
  rf_maxtrees <- train(Ratio~.,
                       data = train_data,
                       method = "rf",
                       metric = "RMSE",
                       tuneGrid = tuneGrid,
                       trControl = trControl,
                       importance = TRUE,
                       nodesize = 16,
                       maxnodes = 8,
                       ntree = ntree,
                       preProcess = c('center','scale'),
                       tuneLength = 3
                       )
  key <- toString(ntree)
  store_maxtrees[[key]] <- rf_maxtrees
}
results_tree <- resamples(store_maxtrees)
summary(results_tree)

#search best nodesize using best maxnode
store_nodesize <- list()
tuneGrid <- expand.grid(.mtry = best_mtry)
for (nodesize in c(1:20)) {
  set.seed(5678)
  rf_nodesize <- train(Ratio~.,
                       data = train_data,
                       method = "rf",
                       metric = "RMSE",
                       tuneGrid = tuneGrid,
                       trControl = trControl,
                       importance = TRUE,
                       nodesize = nodesize,
                       maxnodes = 8,
                       ntree = 100,
                       preProcess = c('center','scale'),
                       tuneLength = 3
                       )
  key <- toString(nodesize)
  store_nodesize[[key]] <- rf_nodesize
}
results_nsize <- resamples(store_nodesize)
summary(results_nsize)


# Define the new control
```

```
trControl2 <- trainControl(method = "cv",
                           number = 250,
                           search = "grid")

#final train model babyyy
set.seed(1234)
tuneGrid <- expand.grid(.mtry = 2)#it is not liking best mtry
fit_rf <- train(train_data[, 1:2],
                train_data$Ratio,
                method = "rf",
                metric = "RMSE",
                tuneGrid = tuneGrid,
                trControl = trControl2,
                importance = FALSE,
                #nodesize = 8,
                maxnodes = 24,
                ntree = 2000,
                preProcess = c('center','scale'),
                tuneLength = 10
                )
#summary(fit_rf)
print(fit_rf)
varImp(fit_rf)

#look at the test data
prediction <- predict(fit_rf, newdata= a)
print(prediction)

postResample(pred = prediction, obs = b)

#THE ISO PREDICTIONS!
ISO_prediction <- predict(fit_rf, newdata = ISOComparisons)
print(ISO_prediction)

#THE OVERALL RATIO PREDICTIONS FOR PLOTTING PURPOSES
overallTrend_prediction <- predict(fit_rf, newdata = overallTrend)
print(overallTrend_prediction)
write.Table(overallTrend_prediction, file = "overallPrediction-ratio-values1.csv",
            row.names = F,
            sep = ","
)
```

3. R Code Output After Cross-Validation Manipulation and Testing ISO Conditions Results

```
>
> current_data <- finalData1
>
> set.seed(100)
>
> trainRowNumbers <- createDataPartition(current_data$Ratio, p=0.8, list=FALSE)
>
> train_data <- current_data[trainRowNumbers,]
>
> test_data <- current_data[-trainRowNumbers,]
>
>
```

```
> x = train_data[, 1:2]
> y= train_data$Ratio
>
> a = test_data[, 1:2]
> b= test_data$Ratio
> #HERE IS A CARETLIST TO DEMONSTRATE WHY I WANT TO USE RANDOMFOREST
> #cross validation
> set.seed(123)
> my_control <- trainControl(method = 'cv',
+                            number = 5,
+                            savePredictions = 'final',
+                            allowParallel = TRUE,
+                            index = createFolds(Y_train,5))
>
> #TRAINING THE MODEL USING OUR X&Y TRAIN SETS
> #testing multiple model methods using methodList
> set.seed(222)
> model_list <- caretList(x,
+                         y,
+                         trControl = my_control,
+                         methodList = c('rf', 'lm', 'ctree'),
+                         tuneList = NULL,
+                         continue_on_fail = FALSE,
+                         preProcess = c('center','scale')
+                         )
note: only 1 unique complexity parameters in default grid. Truncating the grid to 1 .

> #I like to see the models to make sure nothing suspect happened in the making of them
> model_list$rf
Random Forest

804 samples
  2 predictor

Pre-processing: centered (2), scaled (2)
Resampling: Cross-Validated (5 fold)
Summary of sample sizes: 161, 160, 162, 161, 160
Resampling results:

  RMSE      Rsquared  MAE
  6.108728  0.399136  3.915907

Tuning parameter 'mtry' was held constant at a value of 2
> model_list$lm
Linear Regression

804 samples
  2 predictor

Pre-processing: centered (2), scaled (2)
Resampling: Cross-Validated (5 fold)
Summary of sample sizes: 161, 160, 162, 161, 160
Resampling results:

  RMSE      Rsquared   MAE
  6.998939  0.2047864  5.122917
```

```
Tuning parameter 'intercept' was held constant at a value of TRUE
> model_list$ctree
Conditional Inference Tree

804 samples
  2 predictor

Pre-processing: centered (2), scaled (2)
Resampling: Cross-Validated (5 fold)
Summary of sample sizes: 161, 160, 162, 161, 160
Resampling results across tuning parameters:

  mincriterion  RMSE      Rsquared   MAE
  0.01          6.598019  0.3026478  4.446630
  0.50          6.683906  0.2837249  4.606848
  0.99          7.195303  0.1691319  5.258810

RMSE was used to select the optimal model using the smallest value.
The final value used for the model was mincriterion = 0.01.
>
>
> #rf is the winner~~~~~~~~~~~~~~~~~~~~~~~~~
> # Define the control
> trControl <- trainControl(method = "cv",
+                           number = 20,
+                           search = "grid")
>
>
> set.seed(1234)
> # Run the model
> rf_default <- train(train_data[, 1:2],
+                     train_data$Ratio,
+                     data = train_data,
+                     method = "rf",
+                     metric = "RMSE",
+                     trControl = trControl,
+                     preProcess = c('center','scale'),
+                     tuneLength = 3)
note: only 1 unique complexity parameters in default grid. Truncating the grid to 1 .

> # Print the results
> print(rf_default)

Random Forest

804 samples
  2 predictor

Pre-processing: centered (2), scaled (2)
Resampling: Cross-Validated (20 fold)
Summary of sample sizes: 763, 763, 764, 764, 764, 764, ...
Resampling results:

  RMSE      Rsquared   MAE
  5.693178  0.4700003  3.643701
```

```
Tuning parameter 'mtry' was held constant at a value of 2

> # Define the new control
> trControl2 <- trainControl(method = "cv",
+                         number = 250,
+                         search = "grid")
>
> #final train model
> set.seed(1234)
> tuneGrid <- expand.grid(.mtry = 2)
> fit_rf <- train(train_data[, 1:2],
+                 train_data$Ratio,
+                 method = "rf",
+                 metric = "RMSE",
+                 tuneGrid = tuneGrid,
+                 trControl = trControl2,
+                 importance = FALSE,
+                 #nodesize = 8,
+                 maxnodes = 24,
+                 ntree = 2000,
+                 preProcess = c('center','scale'),
+                 tuneLength = 10
+                 )
> print(fit_rf)
Random Forest

804 samples
  2 predictor

Pre-processing: centered (2), scaled (2)
Resampling: Cross-Validated (250 fold)
Summary of sample sizes: 801, 801, 800, 801, 800, 800, ...
Resampling results:

  RMSE      Rsquared   MAE
  4.627624  0.7219174  3.625007

Tuning parameter 'mtry' was held constant at a value of 2
> varImp(fit_rf)
rf variable importance

         Overall
Pressure     100
Temp           0
> #look at the test data
> prediction <- predict(fit_rf, newdata= a)
> print(prediction)
       3       13       14       16       27       46       49       50       58       60       63       77
 5.476508  5.476508  5.476508  3.256996  3.256996 13.809098 13.809098 13.809098 13.809098  2.325385  2.3253
85  2.325385
      83       87       88       90      101      105      109      110      125      128      129      134
 2.325385  2.325385  2.325385  9.434709  9.434709  9.434709  9.434709  9.434709 18.041882 18.041882 18.04188
2 18.041882
     135      141      142      145      146      149      151      167      170      172      176      181
```

```
18.041882 18.041882 18.041882 18.041882 18.041882 14.936981  1.272582  1.272582  1.272582  1.272582  1.272
582 12.737691
     187      191      196      200      201      220      229      244      249      253      260      263
12.737691 12.737691 12.737691 12.737691 12.737691 20.851698 32.762029  8.085833  8.085833  8.085833  8.085
833  8.085833
     276      277      281      284      285      286      308      311      313      324      326      327
 8.085833  8.085833  8.085833  8.085833  8.085833  8.085833  6.409885  6.409885  6.409885  6.409885  6.409885
6.409885
     332      339      356      362      370      373      375      381      385      387      389      392
 6.409885  6.409885  5.613533  5.613533  5.613533  5.613533  5.613533  5.613533  5.613533  5.613533  5.613533
5.613533
     394      396      400      404      409      412      414      417      426      432      433      436
 5.613533  5.613533  5.613533  5.613533  5.613533  5.613533  5.613533  5.613533  5.613533  5.613533  5.613533
5.613533
     437      445      452      458      469      473      474      477      481      482      486      490
 5.613533 10.608918 10.608918 10.608918 10.608918 10.608918 10.608918 10.608918 10.608918 10.608918  8.65
7347  8.657347
     492      496      499      503      504      508      509      511      519      520      525      526
 8.657347  8.657347  8.657347  8.657347  8.657347  8.657347  8.657347  8.657347  8.657347  8.657347  8.657347
8.657347
     527      528      529      537      544      559      560      561      564      569      573      578
 8.657347  8.657347  8.657347  8.657347  8.657347  8.657347  8.657347  8.657347  8.657347  8.657347  8.621056
8.621056
     579      587      588      593      600      605      606      629      636      642      654      661
 8.621056  8.621056  8.621056  8.621056  8.621056  8.621056  8.621056  8.621056  8.621056  8.621056  9.080287
9.080287
     672      683      685      687      689      691      698      706      710      711      715      718
 9.080287  9.080287  9.080287  9.080287  9.080287  9.080287  9.080287  9.080287  9.080287  9.080287  9.080287
9.080287
     724      725      738      739      740      745      747      749      757      758      760      763
 9.080287  9.080287  9.080287  9.080287  9.080287 16.678851 16.678851 16.678851 16.678851 16.678851 16.678
851 16.678851
     776      787      790      791      794      796      806      826      832      836      839      841
16.678851 16.678851 16.678851 16.678851 16.678851 16.678851 16.678851  2.450250  2.450250  2.450250  2.712
128  2.712128
     843      849      852      862      867      871      873      874      887      889      890      893
 2.712128  2.712128  2.712128  2.712128  4.154567  4.154567  4.154567  4.154567  4.154567  4.154567  4.154567
4.154567
     898      905      915      916      926      927      929      933      948      949      955      962
 4.154567  4.154567 11.502261 11.502261 11.502261 11.502261 11.502261 11.502261  4.607275  4.607275  4.607
275  4.607275
     965      970      973      977      984      991      997      999
 4.607275  4.607275  4.607275  4.607275  4.607275  2.893933  2.893933  2.893933
>
> postResample(pred = prediction, obs = b)
    RMSE Rsquared      MAE
6.004487 0.404344 3.942619
> View(ISOComparisons)
> interactionCheck <- read.csv("C:/Users/Asus ROG Beast/Desktop/Schlumberger/DATA ANALYSIS THESIS/Fin
al Consolidation/interactionCheck.csv")
>   View(interactionCheck)
> #ISO Predictions
> ISO_prediction <- predict(fit_rf, newdata = ISOComparisons)
> print(ISO_prediction)
       1        2        3
 4.062301  2.450250 11.502261
```