06-321: Chemical Engineering Thermodynamics
Final Report

Sean Pereira (sapereir), Anusha Agarwala (anushaa), Violet Dong (xdong1)
and Dhruva Byrapatna (dbyrapat)

Fall 2018
Submitted: December 10, 2018

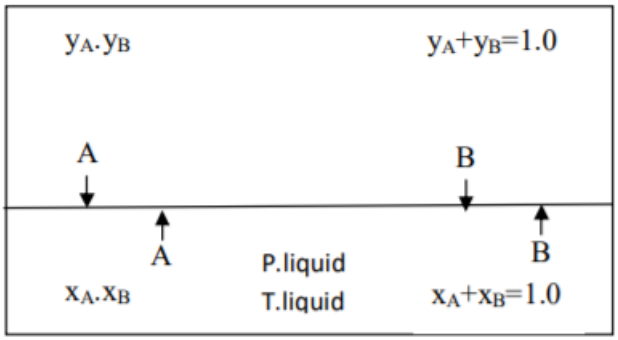
Abstract

We carried out this project to develop a better method of understanding and predicting vapor losses from underground gasoline storage tanks. Predicting these losses is of utmost importance to gasoline distributing companies in order to minimize both financial losses and the environmental impact of their stations. An analysis of current loss detection methods demonstrates that while these monitoring systems are somewhat effective, inadequacies in their modeling methods prevent companies from gaining an accurate understanding of what is happening in their storage tanks. We developed an algorithm based off of thermodynamic concepts to model the vapor and liquid levels inside an underground storage tank. Our algorithm was based off of the Rachford-Rice equation, which governs vapor liquid equilibrium in multiple component systems. We calculated the vapor and liquid volumes, as well as the composition of each, using this equation for a set of sample initial conditions. We then calculated these values for the data-set we were given and compared them to the measured values to ascertain the validity of the model. We determined that the model we developed was somewhat accurate in predicting the quantities of interest. While the model predicts the volume levels and composition with relative accuracy, it is poor at predicting the vapor levels in tanks. While the amount of vapor lost can be determined by the remaining volume, our model cannot accurately predict the composition of vapor, making it difficult to determine which parts of the gasoline mixture are being lost to the environment. While we have developed a relatively accurate base model to predict underground storage tank levels and losses, we can improve on this model further in order to increase the accuracy of vapor loss prediction. Incorporating activity coefficients to account for the non-ideal nature of the gasoline components will increase the accuracy of the model and make our algorithm more useful in predicting vapor losses. Accounting for supply and demand patterns will allow us to take into account environmental factors and give recommendations regarding storage levels in underground storage tanks, enabling us to develop a model in which users can understand the behavior of their underground storage tanks and gain insight in order to make decisions regarding their storage levels.

Introduction

Monitoring losses in underground storage tanks at gas dispensing stations is crucial for companies to both prevent environmental damage and ensure petroleum levels and distribution are properly recorded. There are many components to the problem at hand. The main consideration which gas dispensing stations and energy companies have to take into account are environmental and regulatory concerns. Without an accurate monitoring system for petroleum liquid and vapor levels, it is difficult for gas dispensing stations to gauge whether or not they are adhering to EPA regulations regarding acceptable levels of losses. Because of the damaging impact leaking underground storage tanks can have on groundwater purity, accurate monitoring is crucial. It is a vital to understand the leak losses or liquid losses and evaporation loses or vapor losses. Additionally, being unable to accurately understand petroleum losses has environmental and financial implications. In addition, with levels of leakage varying between dispensing stations, being unable to accurately monitor petroleum levels makes it very difficult to schedule petroleum delivery amounts and manage their overall supply chain. Because of these many issues, improving the current state of technology is essential for the future operations of gas dispensing stations. After understanding the losses, the companies can be advised on new changes that can assist in fixing the losses.

The composition of vapour and liquid equilibrium phases is very important for calculations involving distillation, extraction and absorption processes which find useful application in the chemical process industry, petroleum and refining industries. Complete vapour-liquid equilibrium data for ternary systems are rare in the literature and quaternary data are practically non-existent. VLE data are generally estimated using thermodynamic models based on the well-known fundamental phase equilibrium criterion methods used now.



In this report, we propose a framework to detect the presence of leakages in fuel tanks through the implications of a regression model based solely on thermodynamic models for vapor liquid equilibrium that correlate to volume of the tank.

Literature Review and Results

There are several different methods used to detect leaks in underground storage tanks. Some of these methods include inventory control, nonvolumetric leak detection method, in-tank continuous leak monitoring, leak effects monitoring and tank integrity testing.

Inventory control is a simple detection method, but is not always the most effective leak detection method because it requires that practitioners follow a set of procedures closely. This method involves that practitioners take continuous note of all gas that is dispensed from the tank, and continuous note of all inventory every single day. Because inventory can have a discrepancy if there are liquids other than petroleum in the tank, the practitioners must also check for water in the tanks. Lastly, the gauge caps and fill caps must be in supreme condition to ensure that there are no leaks in liquid transfer. There are ways to replace these practitioners with machinery through an automatic monitoring system.

An automatic monitoring system that decreases the risk of human error in monitoring is called an in-tank continuous leak detection. This method recruits the use of volume in the tank to measure changes

in product level and convert to a volume lost by recruiting information about the dimensions of the tank. While this is a less labor-intensive and easy way to determine the volume of petroleum in the tank that is lost, the actual validity of the data obtained from the equipment is not confirmed, and therefore cannot be used as the sole method for determining leaks.

A separate, but still automatic monitoring system for leak detection uses variables other than volume to understand whether or not there is a leak in the tank. An example of a variable that would lead someone to believe that there is a leak in the UST is the presence of a tracer gas such as helium. While this method is good to identify that there is a leak, it does not quantify the amount of petroleum that is leaking. This also has implications on the UST itself in the case that the petroleum is contaminated by the tracer gas, the leaks are intensified, and the testing time is too long.

The last leak detection method requires the practitioner to examine the area surrounding the tank for any evidence of petroleum. While this seems like a simple and obvious method to looking for a leak, can be unreliable and does not allow for the amount of petroleum leaking out to be quantified.

It is clear that while effective, there are also shortcomings to a lot of the current methods used to understand losses due to leaks in current monitoring systems. Through this report, we aim to develop and test a different method to understanding this information.

One of the challenges in quantitatively and analytically evaluating the leakage of a gas tank, which is the premise of this paper, is imposed by the complexity of gasoline. Gasoline is a multicomponent mixture and it is difficult to determine the partition of components in either gas or liquid phase as Clausius-Clapeyron vapor liquid equilibrium curve is not applicable. However, fundamental equations for vapor liquid equilibrium (VLE) can still be applied to given data.

One of the equations frequently used in VLE calculation is Raoult's Law. In an ideal mixture, $y_i P = x_i P^{sat}$, where y_i is the vapor phase mole fraction, P is the total pressure, x_i is the liquid phase mole fraction, and P^{sat} is the saturation vapor pressure of pure substance i. Given that gas tanks are most likely be in an non-ideal environment, activity coefficient, γ_i should be added to Raoult's Law. The result of it modified Raoult's Law, $y_i P = \gamma_i x_i P^{sat}$. Another correction parameter used in calculation of multicomponent mixture is fugacity. Fugacity is the corrected partial pressure under non-ideal environment and it can be written as $F_i^v(T,P) = \Phi_i(T,P) y_i P$, where $\Phi_i(T,P)$ is the fugacity coefficient. Fugacity of vapor is equal to fugacity of liquid under VLE, so $F_i^v = F_i^l$. P^{sat} mentioned in the above equations can be calculated by Antoine equation, $P^{sat} = e^{(A - \frac{B}{C+T})}$.

Another useful equation is flash calculation. When the pressure is equal or above bubblepoint point of the liquid, liquid evaporates to form a two phase equilibrium. Flash calculation computes the mole fraction of components in vapor and liquid phrases from the initial single phase mole fraction. The equation is written as $\sum_i \frac{z_i K_i}{1+V(Ki-1)} = 1$. This project assumes that the feed stream is 100% liquid with equal partitions, so flash calculation is able to calculate the mole fraction of each component in vapor and liquid phase. Bubblepoint of the system can be determined by $P_{bubble} = \sum_i x_i P_i^{sat}$.

Currently, one method of predicting the composition of the tank and measuring the gas leak is through machine learning, specifically neural network system. Neural network system utilizes pre-existing data sets to model the relationship between variables in layers. It models the path between the output and input through regression by using a large amount of data points. Once the training is done on the majority of the data set, a small amount of the original points are then used to validate the model by comparing the error between experimental results and predicted results. The process is iterated until the most appropriated model is selected. Our team's approach is somewhat similar to the one discussed above. Specific information of the gasoline in the tank such as composition is determined through thermodynamic equations mentioned previously. The information is then trained to formulate a relationship among themselves to determine the gas leak. The following section of this report will discuss, in detail, the technical aspect of this process.

Methods

Today, most facilities operate reactively to problems in tank levels. This often leads to spills, emergency shutdowns, expensive remediation costs, regulatory issues, costly repairs and fines. Tank level forecasting helps manage and abate these and other problems.

Forecasts are created by harnessing the power of real-time and historical data readily available from sensors, meters and records, which helps to:

1. Prevent tank spillage and emergency shutdowns
2. Discover hardware malfunction or failure
3. Schedule maintenance, shutdowns, and logistics
4. Optimize operations and facility efficiency
5. Detect pipeline leaks and slugging
6. Reduce costs, fines, and downtime

The tank level forecasting process starts at the well input. Oil is measured as it comes into the facility via meters and is sent to tanks. Levels are monitored and recorded in tanks during the refining process and then oil, gas, and water output are recorded via sensors, meters, and records. Forecasts are then made using data from the data provided from Veeder with the help of Professor Sahinidis. The folder contains data from one- and two-tank systems:

1. single-tank.zip
2. dual-tank.zip

Each zip file contains 10 .csv files, which range in size from 37M to 127M. Each file contains 1-2 months of data, no large gaps missing data, and at least some non-trivial amount of activity. I have provided our code below as a way of justification of the results later on in the report.

A machine learning algorithm usually takes clean (and often tabular) data, and learns some pattern in the data, to make predictions on new data. However, when ML is used in real-world applications, the raw information that you get from the real-world is often not ready to be fed into the ML algorithm.

So this entire framework from converting raw data to data usable by ML algorithm, training an ML algorithm, and finally using the output of the ML algorithm to perform actions in the real-world is the **pipeline**. It is called a pipeline because it is analogous to physical pipelines just as a liquid passes through one pipe, entering the next, sequentially, our data goes through one stage, entering into the next, sequentially.

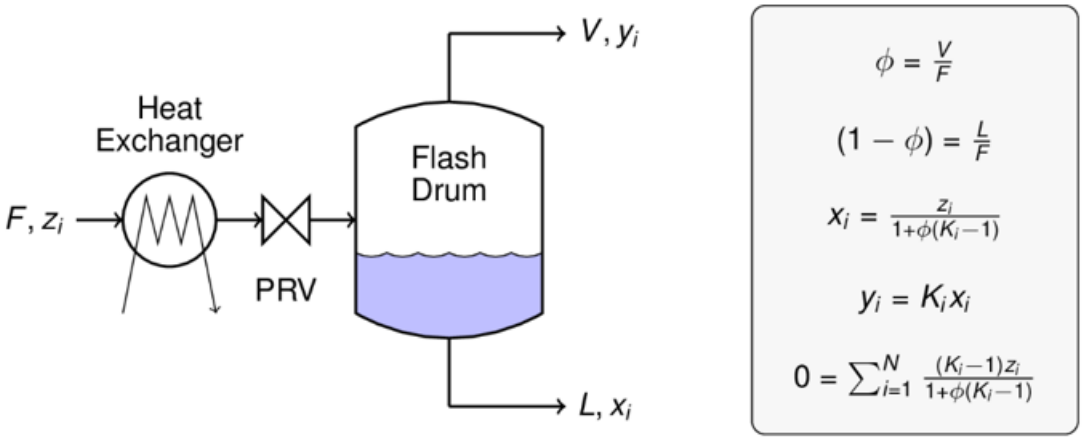
Now, you train an ML model, and you get an accuracy of 80%. Is that good? You cant answer that question unless you compare your accuracy to something else. That something else is the baseline. Sometimes, you pick a simple baseline. So for the example above, a simple baseline could be to just tag each word with its most common part-of-speech. Or you can use existing popular algorithms for that task as the baseline. The choice of the baseline depends on your objective. The goal is to either beat the **baseline**, if the goal of your work is to improve accuracy, or get results comparable to the baseline while improving some other aspect of the algorithm (like training time, prediction time, memory usage, etc.)

The majority of practical machine learning uses supervised learning. Supervised learning is where you have input variables (x) and an output variable (Y) and you use an algorithm to learn the mapping function from the input to the output $Y = f(X)$. The goal is to approximate the mapping function so well that when you have new input data (x) that you can predict the output variables (Y) for that data. Techniques of Supervised Machine Learning algorithms include linear and logistic regression, Supervised learning requires that the data used to train the algorithm is already labeled with correct answers provided by Veeder.

Our project will compute values of Vapor Liquid Equilibrium using ideas such as Raoult's Law, Bubble Point Pressure, then apply Flash calculations to find the vapor levels and liquid levels in the tank $Lx_i + Vy_i = F$. We apply these $f(LiquidRatio) = TankHeight$. Additionally, Like I said before the goal is to approximate the mapping function so well that when you have new input data (x) that you can predict the output variables (Y) for that data. We will compute this on the training set and compare this to the Test Set. In addition, the other model will compute $f(VaporRatio) = VaporVolume$

Results / Our Algorithm

The quantities, definitions, and equations are summarized in the following figure.



To sketch the derivation, we begin with the overall constraint on the liquid phase and vapor phase mole fractions $x_1 + x_2 + \cdots + x_N = 1$ and $y_1 + y_2 + \cdots + y_N = 1$. Subtracting the first from the second we find

$$\sum_{n=1}^{N}(y_n - x_n) = 0$$

This doesn't look like much, but it turns out to be the essential trick in the development.
Next we need expressions for y_n and x_n to substitute into terms in the sum. We get these by solving the component material balance and equilibrium equations for y_n and x_n. For each species we write a material balance

$$Lx_n + Vy_n = Fz_n$$

Dividing through by the feedrate we get a parameter $\phi = \frac{V}{L}$ denoting the fraction of the feedstream that leaves the flash unit in the vapor stream, the remaining fraction $1 - \phi$ leaving in the liquid stream. With this notation the material balance becomes

$$(1 - \phi)x_n + \phi y_n = z_n$$

for each species.
The second equation is

$$y_n = K_n x_n$$

where the 'K-factor' for an ideal solution is given by Raoult's law

$$K_n = \frac{P_n^{sat}(T)}{P}$$

The K-factor depends on the operating pressure and temperature of the flash unit. Solving the material balance and equilibrium equations gives

$$x_n = \frac{z_n}{1 + \phi(K_n - 1)}$$

$$y_n = \frac{K_n z_n}{1 + \phi(K_n - 1)}$$

so that the difference $y_n - x_n$ is given by

$$y_n - x_n = \frac{(K_n - 1)z_n}{1 + \phi(K_n - 1)}$$

Substitution leads to the Rachford-Rice equation

$$\sum_{n=1}^{N} \frac{(K_n - 1)z_n}{1 + \phi(K_n - 1)} = 0$$

This equation can be used to solve a variety of vapor-liquid equilibrium problems as outline in the following table.

Problem Type	z_i's	T	P	φ	Action
Bubble Point	✓	unknown	✓	0	Set $x_i = z_i$. Solve for T and y_i's
Bubble Point	✓	✓	unknown	0	Set $x_i = z_i$. Solve for P and y_i's
Dew Point	✓	unknown	✓	1	Set $y_i = z_i$. Solve for T and x_i's
Dew Point	✓	✓	unknown	1	Set $y_i = z_i$. Solve for P and x_i's
Isothermal Flash	✓	✓	✓	unknown	Solve for φ, x_i's, and y_i's

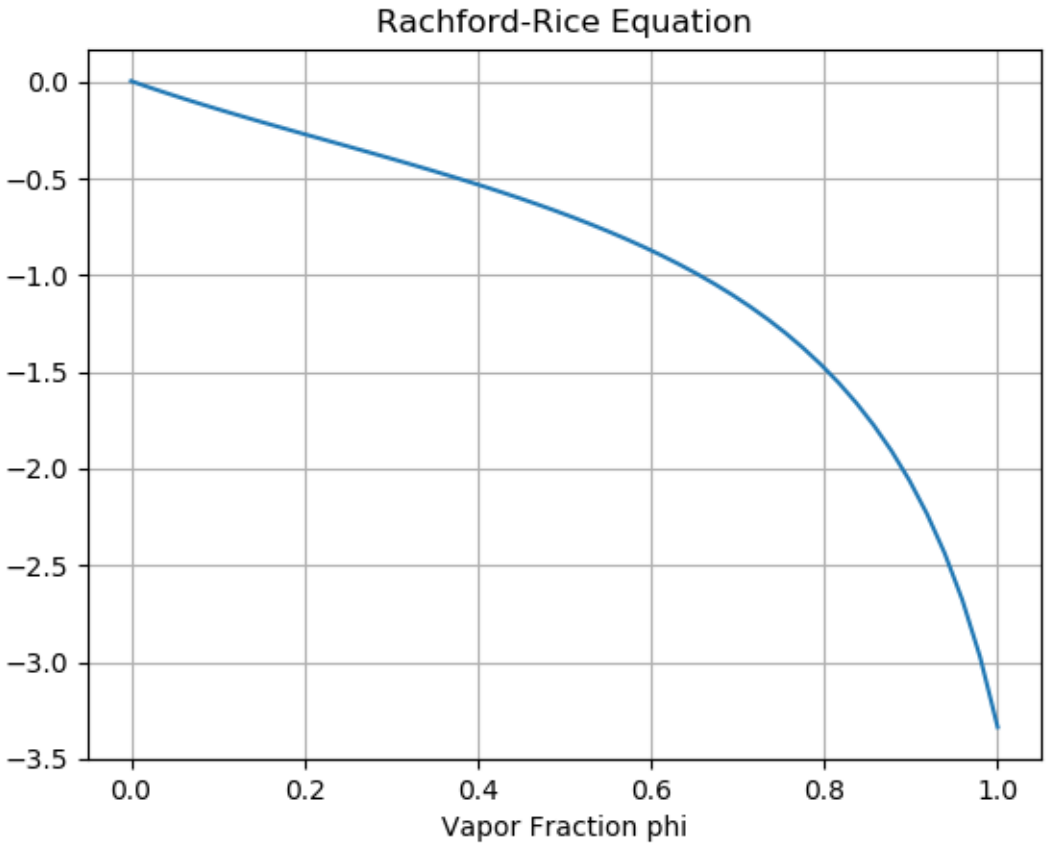
Code

```
#Thermo Project
import numpy as np
import math
import csv
import matplotlib.pyplot as plt
from sklearn import linear_model
from scipy.optimize import fsolve
from scipy.optimize import brentq

Psat = dict()
Psat['3-ethyltoluene'] = lambda T: math.exp(13.7819 - 2726.81/(217.572 + T))
#3-ethyltoluene: https://webbook.nist.gov/cgi/cbook.cgi?ID=C620144&Units=SI&Mask=7FF
Psat['isooctane'] = lambda T: math.exp(13.6703 - 2896.31/(220.767 + T))
#isooctane: https://webbook.nist.gov/cgi/cbook.cgi?ID=C540841&Mask=FFF
Psat['ethanol'] = lambda T: math.exp(16.8958 - 3795.17 /(230.918 + T))
#ethanol: https://webbook.nist.gov/cgi/cbook.cgi?ID=C64175&Mask=4&Type=ANTOINE&Plot=on#ANTOINE
Psat['butane'] = lambda T: math.exp(13.6608 - 2154.70/(238.789 + T))
#butane: https://webbook.nist.gov/cgi/cbook.cgi?ID=C106978&Mask=4&Type=ANTOINE&Plot=on

z = dict()
```

```
species = ['ethanol', 'isooctane', '3-ethyltoluene', 'butane']
T = 55
#x = z
z['ethanol'] = 0.1
z['isooctane'] = 0.3
z['3-ethyltoluene'] = 0.3
z['butane'] = 1 - sum(z.values())

def Bubble_Point(species, T):
    return sum([z[s]*(Psat[s](T)) for s in species])

def ybub(species,x):
    return {s: z[s]*Psat[s](T)/Bubble_Point(species, T) for s in species}

P = 191 #~Bubble_Point(species,T)

K = {n : lambda P,T,n=n: Psat[n](T)/P for n in Psat}

print("Pressure    {:6.2f} [mmHg]".format(P))
print("Temperature {:6.2f} [deg C]".format(T))
print("K-factors:")
for n in K:
    print("  {:s} {:7.3f}".format(n,K[n](P,T)))

def RR(phi):
    return sum([(K[n](P,T)-1)*z[n]/(1 + phi*(K[n](P,T)-1)) for n in K.keys()])

phi = np.linspace(0,1)
plt.plot(phi,[RR(phi) for phi in phi])
plt.xlabel('Vapor Fraction phi')
plt.title('Rachford-Rice Equation')
plt.grid()
plt.show()

phi = brentq(RR,0,1)

print("Vapor Fraction {:6.4f}".format(phi))
print("Liquid Fraction {:6.4f}".format(1-phi))

x = {n: z[n]/(1 + phi*(K[n](P,T)-1)) for n in z}
y = {n: K[n](P,T)*z[n]/(1 + phi*(K[n](P,T)-1)) for n in z}

print("Component  z[n]    x[n]    y[n]")

liquid_ratio = phi
vapor_ratio = 1-phi

for n in z.keys():
    print("{:10s} {:6.4f} {:6.4f} {:6.4f}".format(n,z[n],x[n],y[n]))

def filter_list_liquid(x):
    totalSales = x[3]
    FuelHeight = x[28]
    Temperature_Tank = x[29]
    liquid_Volume = x[26]
    return [totalSales, Temperature_Tank, liquid_Volume, liquid_ratio, FuelHeight]

def filter_list_vapor(x):
    totalSales = x[3]
    Vapor_Temp = x[31]
```

```
    vapor_volume = x[35]
    return [totalSales, Vapor_Temp, vapor_ratio, vapor_volume]

with open('Thermo_Data_Training.csv') as csvfile:
    data = csv.reader(csvfile)
    result_t_liquid = []
    result_t_vapor = []
    row_i = 0
    for row in data:
        if row_i == 0:
            row_i += 1
            continue
        result_t_liquid.append(filter_list_liquid(list(row)))
        result_t_vapor.append(filter_list_vapor(list(row)))
        row_i += 1
    arr_liquid_t_x = np.array(result_t_liquid, dtype='f')
    arr_vapor_t_x = np.array(result_t_vapor, dtype='f')

with open('Thermo_Data_Test.csv') as csvfile:
    data = csv.reader(csvfile)
    result_liquid = []
    result_vapor = []
    row_i = 0
    for row in data:
        if row_i == 0:
            row_i += 1
            continue
        result_liquid.append(filter_list_liquid(list(row)))
        result_vapor.append(filter_list_vapor(list(row)))
        row_i += 1
    arr_liquid_x = np.array(result_liquid, dtype='f')
    arr_vapor_x = np.array(result_vapor, dtype='f')

arr_l_t_x = arr_liquid_t_x[:, :-1]
arr_l_t_y = arr_liquid_t_x[:, -1]
arr_v_t_x = arr_vapor_t_x[:, :-1]
arr_v_t_y = arr_vapor_t_x[:, -1]
arr_l_x = arr_liquid_x[:, :-1]
arr_l_y = arr_liquid_x[:, -1]
arr_v_x = arr_vapor_x[:, :-1]
arr_v_y = arr_vapor_x[:, -1]

liquid_model = linear_model.LinearRegression(normalize=True)
liquid_model.fit(arr_l_t_x, arr_l_t_y)
print "Liquid Model Score:", abs(liquid_model.score(arr_l_x, arr_l_y))

vapor_model = linear_model.LinearRegression(normalize=True)
vapor_model.fit(arr_v_t_x,arr_v_t_y)
print "Vapor Model Score:", abs(vapor_model.score(arr_v_x, arr_v_y))
```

```
C:\Users\Sean Pereira\Desktop\thermo>python thermo.py
Pressure     191.00 [mmHg]
Temperature   55.00 [deg C]
K-factors:
   3-ethyltoluene    0.229
   isooctane    0.124
   ethanol    0.196
   butane    2.928
Vapor Fraction  0.0025
Liquid Fraction 0.9975
Component    z[n]    x[n]    y[n]
3-ethyltoluene 0.3000  0.3006  0.0688
isooctane  0.3000  0.3007  0.0374
ethanol    0.1000  0.1002  0.0196
butane     0.3000  0.2985  0.8742
Liquid Model Score: 0.08897805486293153
Vapor Model Score: 0.4577723419328407
```



Assumptions

1. Assumed 10% Ethanol, 30% Isooctane, 30% 3-ethyltoluene, and 30% Butane

2. Assumed the application of the Ideal Gas System for VLE, however could have used the activity coefficients, and fugacity.

3. Assumed values of Antoine's Coefficients

4. Assumed values for the tank geometry, and calculated pressure instead of having proper value.

The pseudocode below explains the general gist of the code:

Algorithm 1 Our algorithm: VLE Regression Solver

Init Liquid_Train $\leftarrow \{\}$
Init Vapor_Train $\leftarrow \{\}$
Init Liquid_Test $\leftarrow \{\}$
Init Vapor_Test $\leftarrow \{\}$

for ***0*** to ***n*** **do** *(where* ***n*** *is the number of data rows)*
    Compute: $\forall i \in n(1-\phi)x_i + \phi y_i = z_i$ for given V_T, T_T
    if *Liquid Calculation* **then**
        Liquid_Train $\leftarrow \{\} + f(S, L_T, L_V, L_R) = F_H$
        Liquid_Test $\leftarrow \{\} + f(S, L_T, L_V, L_R)$
    if *Vapor Calculation* **then**
        Vapor_Train $\leftarrow \{\} + f(S, V_T, V_R) = V_V$
        Vapor_Test $\leftarrow \{\} + f(S, L_T, L_V, L_R)$
    goto *top.*

$liquid\_model.fit(Liquid\_Train)$
$vapor\_model.fit(Vapor\_Train)$
return $liquid\_model.score(Liquid\_Test)$, $vapor\_model.score(Vapor\_Test)$

Results Summary

Our Liquid Regression Model to predict the Tank Level Height based on the Vapor-Liquid Equilibrium combined with the Veeder Data had a score of ≈ 0.08897 and our Our Vapor Regression Model to predict the Vapor Volume based on the Vapor-Liquid Equilibrium Model combined with the Veeder Data had a score of ≈ 0.45777. This can attributed to the many underlying assumptions we made about the vapor liquid model. The liquid score demonstrates that our model performed closer to the expected value without the basis of the input features. Additionally, the vapor score demonstrates the ability for our model to perform, but doesn't serve as an accurate one for future use. We think this isn't attributed to the lack of the ability of the Regression model, but the incorrect implementation of our Vapor-Liquid Equilibrium Model. This fix will be discussed in our conclusions.

Conclusion and Future Recommendations

Overall, this project was a successful learning experience. We were able to predict vapor and liquid levels in underground storage tanks. We performed a multiple linear regression on the data we obtained for vapor and liquid levels in a tank. The data for liquid levels had a much higher R^2 value than the data for vapor levels, indicating that the model did a better job of predicting tank liquid levels. It is possible to simply predict the vapor level in the tank from the liquid levels. However, as the statistical correlation for both vapor and liquid was fairly low, future work must first concentrate on improving the accuracy of the model in order to make better predictions.

Future work should first focus on gathering more data, most likely by talking to industry experts and company representatives, in order to eliminate some of our assumptions. We faced issues determining the correct value of the Antoine Coefficients, as online sources gave conflicting data. In addition, we assumed

the composition of the gasoline mixture in the tank based off of generalized information. Knowing the exact Antoine's Coefficients which apply to the tank conditions as well as the exact composition of the gasoline from which data was collected will help us improve the accuracy of the model by giving an accurate representation of the mixture's properties. In addition, we did not incorporate activity coefficients into our pressure calculations because we found conflicting values in our literature search; we instead assumed ideal gas behavior for the VLE system. Determining the correct activity coefficients would enable us to obtain more accurate pressures, and therefore a more accurate volume prediction. Moving forward, we will attempt to use the UNIQUAC Functional-group Activity Coefficients (UNIFAC) model in order to calculate the activity coefficient for each species in solution. This method has been employed by other research groups to a good deal of success, and would likely dramatically improve our results.

After we improve the accuracy of our results, we will focus on incorporating other factors so that our model can make recommendations on storage levels in order to satisfy client needs. We will be able to make the model more specific, calculating vapor loss levels for individual gas stations based on their specific locations, tank geometries, composition of their gasoline, and so on. Figuring out how these factors impact vapor loss levels at specific gas stations will enable us to create a model which can incorporate any station's unique characteristics in order to predict their vapor losses. In addition, the amount of gasoline companies need to store in UST's varies over time as demand changes seasonally; periods of higher demand necessitate higher storage levels in order to satisfy consumer needs and prevent stock-out issues. By incorporating predicted demand levels into the model, we will be able to incorporate the conditions a gasoline station faces in order to give users a recommended storage level which will balance loss minimization with ensuring that gas stations have enough gasoline on hand to meet demand. Finally, improving the model accuracy to better predict the vapor composition will enable us to take financial and environmental considerations into account. Not all parts of a gasoline mixture have the same financial value or the same environmental impact. Certain hydrocarbons and additives are more crucial and will have a more harmful environmental impact if released into the environment. A better representation of vapor composition will enable us to minimize the financial and environmental impact of vapor losses, possibly finding an optimal storage level with a higher amount of total vapor release but minimal losses of components which are financially and environmentally impactful. We have developed a base model for vapor losses which we can improve on. Building off of this model will enable us to more accurately represent vapor losses and develop a useful tool for potential clients to determine optimal storage levels in underground storage tanks.

References

[1] Spokane Regional Clean Air Agency. *Air Quality Requirements for Gasoline Dispensing*

[2] *Learn About Underground Storage Tanks (USTs).* EPA, Environmental Protection Agency, 11 Apr. 2017.

[3] Geyer, Wayne. *UST History.*

[4] Leak Detection: Statistical Inventory Reconciliation (SIR). North Dakota UST Operator Training Program.

[5] *Automatic Tank Gauging (ATG).* North Dakota UST Operator Training Program.

[6] *Daniyan, I.A., et al. Correlation of Vapour-Liquid Equilibrium Data Using Neural Network for Hydrocarbon Ternary System (Ethane-n-Pentane-n-Hexane). Department of Chemical Engineering, Department of Computer Science, Afe Babalola University, 2013.*

[7] Electronic Line Leak Detection. Electronic Fuel Line Leak Detection Veeder-Root.

[8] Gas Stations. Home, *www.oilspillprevention.org/oil-spill-sources/gas-stations.*

[9] *Harley, Robert A., et al. Relating Liquid Fuel and Headspace Vapor Composition for California Reformulated Gasoline Samples Containing Ethanol. Environmental Science Technology, vol. 34, no. 19, 2000, pp. 40884094., doi:10.1021/es0009875.*

[10] *Jckantor. Jckantor/CBE20255. GitHub, github.com/jckantor/*

[11] *LanduseKN, director. Convective Losses from Storage Tanks, Retail Gas Station. YouTube, YouTube, 30 Oct. 2012, www.youtube.com/watch?v=ithif3FFIjU.*

[12] U.S. Prime Supplier Sales Volumes of Petroleum Products. Factors Affecting Gasoline Prices - Energy Explained, Your Guide To Understanding Energy - Energy Information Administration.

[13] OPW VaporSaver Stage II Vapor Recovery System. YouTube, 19 Nov. 2010.

[14] Machine Learning with Python https://towardsdatascience.com/machine-learning-with-python-easy-and-robust-method-to-fit-nonlinear-data-19e8a1ddbd49?fbclid=IwAR3mrPkzMeR3GWAZ6lLZTemBSnPVf-AlgPtwiANBBz2xS2yN74bEobGTrmE

[15] *Microsoft Azure Oil Forecasting: https://github.com/Azure/tank-level-forecasting-guide*

[16] *Rowson, Jackie. What Chemicals Are in Motor Oil? It Still Runs, 12 June 2017, itstillruns.com/chemicals-motor-oil-7530158.html.*

[17] Underground Storage Tanks (USTs). EPA, Environmental Protection Agency, 4 Dec. 2018, www.epa.gov/ust.

[18] *Wrenick, Scott, et al. HEAT TRANSFER PROPERTIES OF ENGINE OILS. U.S. Army Tank-Automotive and Armaments Command, Warren, MI.*

[19] *What is machine learning? https://www.quora.com/What-is-a-pipeline-and-baseline-in-machine-learning-algorithms*