Comparative Assessment of Random Forest Algorithm, Probabilistic Neural Networks, K-Nearest Neighbor Algorithms, and Decision Tree Algorithms for Predicting Water Usage and Product Wastage In Drilling Operations

By: Sendhil Sridhar

Abstract:

Drilling prospects in the United States are heavily dependent on the economic cycle. Optimizing the process is very desirable for energy producers to maximize profits overall and/or minimize wastage of valuable products - whether it be oil or gas. The goal of this work is to analyze well site data from different formations in order to develop models that predict how much water will be used in the drilling process (operating costs for the drilling company), and predict how much flaring/venting will occur per well. In this paper, the performance of four data-driven models with different structures including a Artificial Neural Network (ANN), K-Nearest Neighbor Regression (KNN), Decision Tree Regression (DTR), and a Random Forest Regression (RFR) are calculated to project future usage of water in drilling and amount of product either vented/flared. The returned scores from the models are determined using Monte-Carlo Cross Validation Method (MCCVM). Results show that performance of the Artificial Neural Network is the best as it manages to compute an R^2 score for all the different formations, however the model returns poor R^2 scores across all the formations analyzed. Random Forest Regression serves to be the most practical algorithm as it returns consistently high R^2 scores and only was unable to compute one R^2 score due to the low amount of data available for that calculation. The results of this work can be used in future regulations in monitoring the amount of water used in drilling or how much gas is vented and or flared. The results can also be applied to future drilling processes by companies drilling for oil/gas in any of the formations that were studied, allowing them to optimize their drilling process.

Introduction:

Understanding the costs involved in drilling is important before the process is actually carried out. In a standard drilling process for a conventional well in an arbitrary formation, tools like degassers, shakers, sand pumps etc. are required for a successful procedure. The average well cost from exploration all the way to completion in 2016 was somewhere between $4.9 million and $8.3 million, however depending on an individual well, the costs could vary significantly.

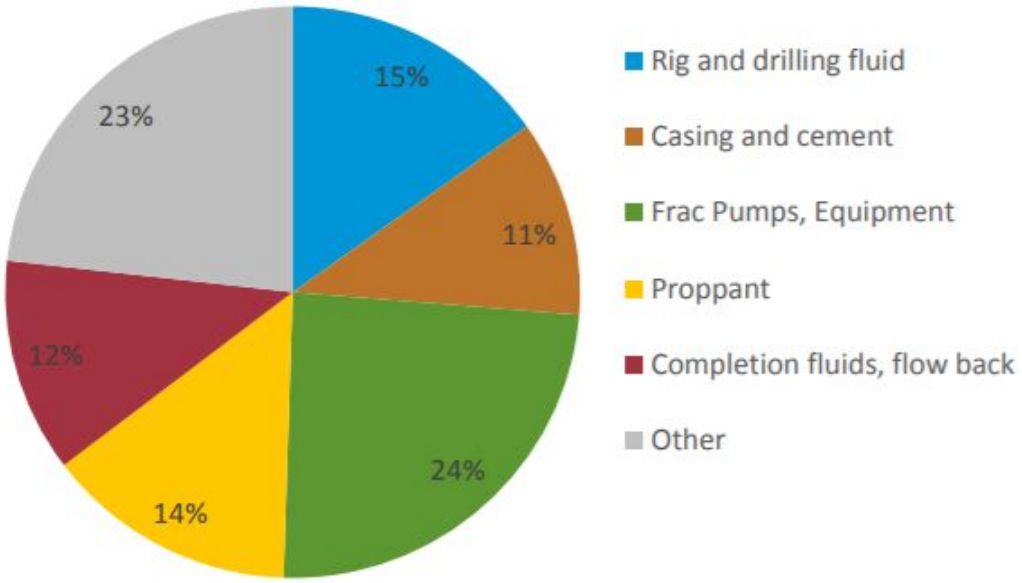

Fig 1: Pie Chart breaking down the costs in drilling a well. Source: EIA

It was found that for an average well, the majority of the cost came from fluids - primarily water based fluids - as completion fluids, rig fluids, drilling fluids, and casing/cement were found to account for about 38% of the total cost (EIA, 2016).

The Energy Department analyzed the amount of flaring/venting that took place during the drilling process. It was estimated that in 2017 that Texas and North Dakota alone accounted for nearly 200 billion cubic feet of volume of gas that was flared/vented, and that the total volume flared by the USA in 2017 was about 300 billion cubic feet, and that they expect the number to increase in the future; prediction made in mind before COVID-19 (Energy Department, 2017). A more recent investigation found that in North Dakota's Bakken formation alone, approximately $100 million dollars of revenue is lost due to flaring, and that number is expected to increase as the volume of natural gas drilled in the Bakken formation is expected to double by 2025. The average closing price of natural gas in 2018 was $3.15/MMBtu. Using that cost, this means the daily, about 300,000 mmcf of natural gas was being flared every day (Scheyder, 2018).

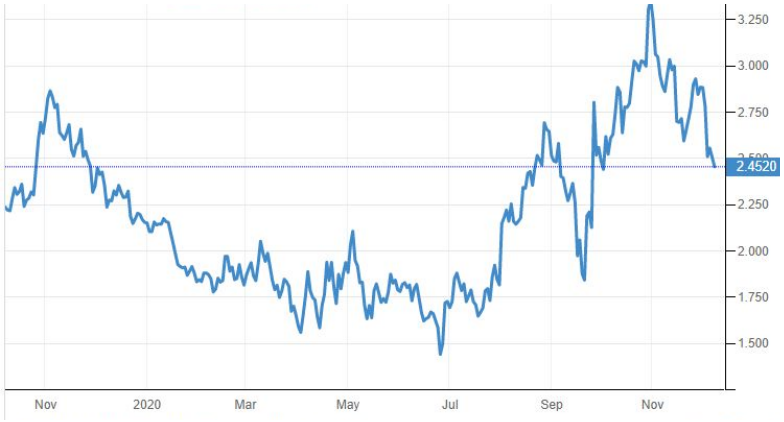

Fig 2: Average price of Natural Gas in 2021 is estimated to be about $2.452/mcf. Source: Trading Economics

New pipelines and storage will inevitably be developed in the future which will likely minimize the amount of product that is flared. It is impossible to predict the cost of natural gas five years into the future. As a result, projections into 2021 seem to be the most reliable cost estimate in the Using the Bakken projections, in 2025 about 1,800,000 mmcf would be produced daily. Assuming a direct relationship between volume of flared product and the future project amount of flared gas would be 600,000 mmcf daily. This would mean $147 million of revenue (based on 2021 natural gas prices) - calculated using projected costs - would be lost monthly from natural gas produced in the Bakken field, which is a significant amount of money.

In order to avoid such steep costs - whether it comes from water usage or product wastage - it would be prudent to take a look at past data to make more educated guesses about the future.

Rationale:

The data used in the work is collected from 22 of the most commonly drilled formations - including formations like Bakken and Tyler - across the United States. The wells are primarily conventional wells with oil being the primary product drilled for in many of the wells and natural gas being a byproduct. The data collected spans a time frame from 2009-2013. Using Regression algorithms, analyzing past data from different formations to predict future usage in water or how much gas will be flared and or vented in the future when drilling naturally picks in those formations again, with the goal of minimizing water and natural gas wastage with more targeted drilling. Models will be developed with quantities like oil and gas produced and be used to predict the amount of water used, and volume of gas flared and or vented.

Method and Design:

Since there are no classes being analyzed in this work, the necessary algorithms to analyze the data are regressor algorithms. Along with that, this is an unsupervised learning problem as the input data contains no labels. The algorithms that were used were K-Nearest Neighbors, Random Forest, and Decision Tree. The goal was to find a solution with the most accurate results but most practical computation time. To compare the results, the R^2 scores of the various models were computed. Using the R^2 scores, a metric was developed to function as a pseudo-accuracy. This metric was based on the average R^2 scores returned and the number of non-computed R^2 scores.

<u>K-Nearest Neighbor:</u>

K-Nearest Neighbor algorithm (KNN) is a non-parametric method, meaning the algorithm assumes the data being analyzed has no implicit distribution that it follows. It is also a lazy learning algorithm, and only generalizes training data once a query is made by the system. This method utilized a weighted nearest neighbour classifier with the ith nearest neighbor assigned weight w_{ni}, with the total weight summing to 1 as shown below.

$$\sum_{i=1}^{n} w_{ni} = 1$$

For calculating the distance function, an Euclidean distance function was implemented with the formula below:

$$d(x, y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

The advantages with KNN is that it's easy to implement. Being a lazy learning algorithm, there is no training period and only makes predictions real time. However, this property causes KNN to fail when larger datasets are present. The cost involved in calculating the distance is expensive and only done real time rather than the training period (which KNN does not have), causing the performance to drop. To optimize the model for this work, the number of n_neighbors were modified to see which one had the best result.

Random Forest:

Random Forest algorithm (RFR) is an ensemble learning method, meaning it combines other trees into one large 'forest' of trees which has a stronger predictive power than a singular tree alone. For training the model, bagging was used. A random sample with replacement was used to train each individual tree and create an overall model on the test data using the average prediction of all the trees.

$$\hat{f} = \frac{1}{B}\sum_{b=1}^{B} f_b(x')$$

Where *B* represents the total number of samples, *x'* is the test data, f_b represents a trained tree, and $\widehat{f}$ is the average prediction from all trees.

The advantage with RFR is that a strong computing power is given for any dataset. The necessary number of trees are always created for any dataset given a specification (a max depth, leaf purity, criterion etc.). RFR presents an issue where it will sometimes overfit the data. In the process of creating a multitude of trees, the forest can train itself to be 100% accurate on training data and when used on test data, it performs poorly. In the models, 2 different criterions - Mean Absolute Error (MAE) and Mean Squared Error (MSE) - were computed to optimize the model. MAE measures the absolute average distance between training and test data, while MSE computes the squared average data between training and test data.

Decision Tree:

The Decision Tree algorithm (DTR) is somewhat similar to the Random Forest algorithm. Decision Trees are in essence a sub-unit of Random Forests, and so many of the properties found in the Decision Trees are the same as Random Forest. In a similar fashion to

RFR, DTR calculates an overall average prediction instead of f_b representing a trained tree, it represents a node within the DTR.

A property of DTR's is that it is a greedy algorithm. This means DTR's take the best choice at each step rather than choosing the most optimal pattern overall. This allows for straightforward visualization of the path taken while forming the tree, and often produces a fast and generally robust model. However, being greedy is the downfall for the DTR. Especially in large datasets, the resulting tree that's formed is extremely deep due to its greedy nature and can result in overfitting. As a result the tree requires pruning and a max depth limit in order to avoid overfitting. In a method similar to RFR, the criterion was modified with MAE and MSE functions, to generate potential optimized models.

Artificial Neural Network:

The Artificial Neural Network (ANN)is a construct made up of neurons, and can recognize patterns from training data to make a final model which predicts based on test data. The function of the ANN in this work was to act as a control model, and thus would not be optimized. It consisted of 1 layer with an arbitrary number of nodes. The ANN had a weight tensor and drew sample from a normal distribution centered around 0 following the equation:

$$\sigma = (2/f)^{1/2}$$

where sigma represents the standard deviation, and f represents the number of inputs to the weight tensor.

For this work, the data that was used was data consisting of the quantities API, formation locations, RPT date, days the wells was producing, barrels of oil produced, barrels of oil conditions, barrels of water used, cubic feet of natural gas produced, cubic feet of natural gas sold, cubic feet of product (not necessarily natural gas) flared, cubic feet of product vented, and treatment date. The data consisted of approximately 300,000 rows of wellsite data from 22 different formations. The quantities used for the general dependent variable 'y' were barrels of water, cubic feet of product flared, and cubic feet of product vented. The quantities used for the general independent variable 'X' were days the well was producing, barrels of oil produced, barrels of oil conditioned, cubic feet of natural gas produced, and cubic feet of natural gas sold. The reason these quantities were chosen was due to the direct impact/influence they have on one of the three quantities that were being analyzed. Formation location was used to further classify the data to give results for each specific formation. The other quantities (API, RPT date, treatment date) were 'dropped' during the pre-processing of the data as they had no significant relationship with 'y'.

For all the models, a general procedure was employed. First the data was checked to see if there were any Nan or null scores. Thankfully, there were no such scores. Then the data was transformed into the 'X' and 'y' variables used in the work as described earlier. Then the dataset was shuffled and split into training data and test data, with the training data size being 80% of the total dataset and the test data size taking the other 20%. Using those general variables - no

formation sorting yet - the Regressor algorithms were run to return a 'score' which returns the mean weighted R^2 score of the model on the dataset. Using the Monte-Carlo Cross Validation Method, the test-train split of 'X' and 'y' and computation of the R^2 score was calculated 50 more times and the average R^2 was computed as the final R^2 score of the model.

Using the K-Nearest Neighbors algorithm, the general regression model returned a R^2 score of .81 and the Random Forest algorithm returned a R^2 score of .94. Due to the multi-out put of the model (three quantities as the 'y' variable), the ANN was unable to create a general R^2 score that could be compared against the other models, as it would output three R^2 scores across the three variables rather than one R^2 score. The Decision Tree algorithm did not return a general regression R^2 score. It looks like in the code (coded in Python), the algorithm is recognizing a class DecisionTreeRegressor() however is unable to return an object from it (Refer to code for more info). This seems to be a feature of the Decision Tree class itself. As a result, the Decision Tree models had no baseline model and was graphed with a score of 0. The results are shown in the graph below.

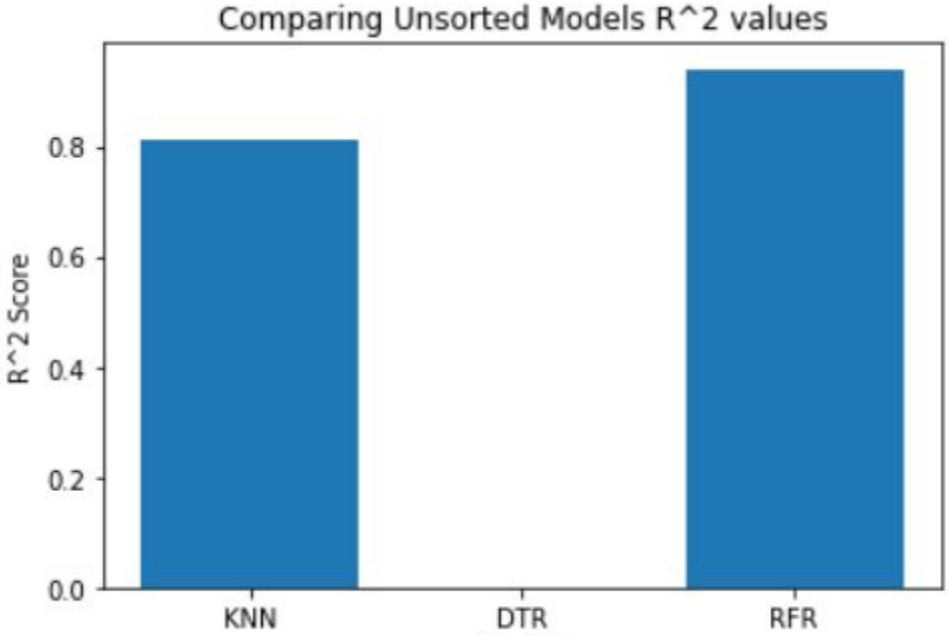

Fig 3: Result showing the scores of the R^2 score. As explained earlier, the DecisionTreeRegressor() failed to produce a model, and has a value of 0.

To generate more specific models, the different formations that were found in the 'Formation location' quantity was used to sort the data into the 22 different datasets based on the 21 unique formations that the data was collected from. There the models were developed using the same algorithm as before and the R^2 scores were calculated using the same Monte-Carlo Cross Validation Method as described before. The individual algorithms were optimized by changing the hyperparameters present within the model. For the K-Nearest Neighbors algorithm it was changing the 'n_neighbors' hyperparameter, for the Decision Tree algorithm it was

modifying the criterion used, and for the Random Forest algorithm it was modified similarly to the Decision Tree Algorithm by altering the criterion. Along with this, a baseline Artificial Neural Network (ANN) was created to function as a control against the other models to see which one returned the most valid model. For the graphs, an R^2 score of 2 indicates that the R^2 score was not computed. This was a common pattern for datasets 4 and 21 which corresponds to the Bakken Formation and UnknownXML respectively. For the Bakken formation dataset, there were runtime issues with some of the models causing the IDE to crash due to an excess in RAM usage, leading to no R^2 score. For the UnknownXML dataset, the lack of data points (only three present) led to an non-computable R^2 score. Since Multi Output Regression was done on all the models except for the ANN, the R^2 scores remain the same across the three graphs except for the ANN. The results of the models R^2 scores are below.

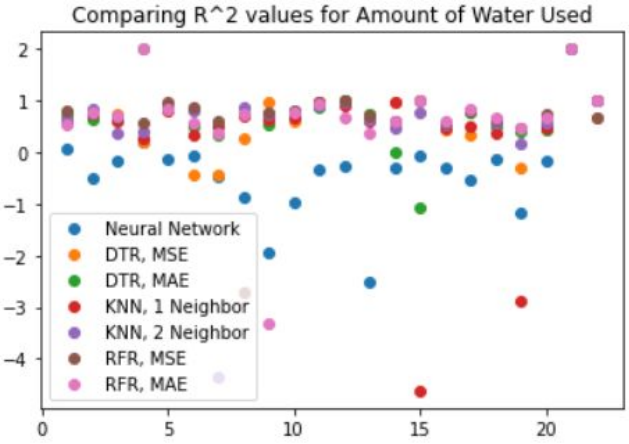

Fig 3: Plotting the R^2 scores across the 22 different formations measuring water usage

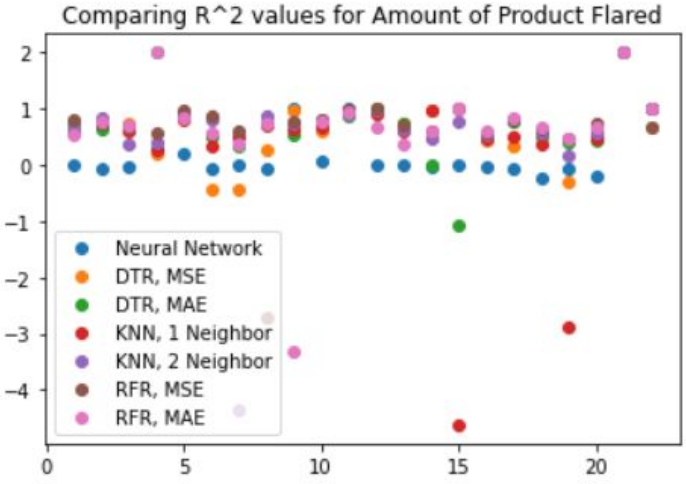

Fig 4: Plotting the R^2 scores across the 22 different formations measuring product flared

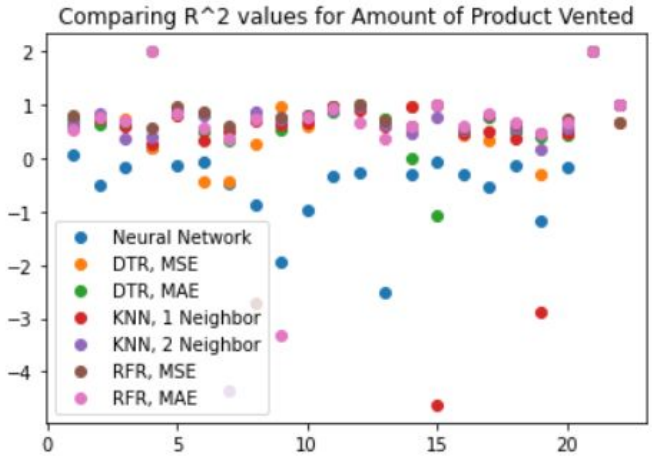

Fig 5: Plotting the R^2 scores across the 22 different formations measuring product vented

Along with that, the number of fail cases - cases where the R^2 scores are not computed - are graphed below.

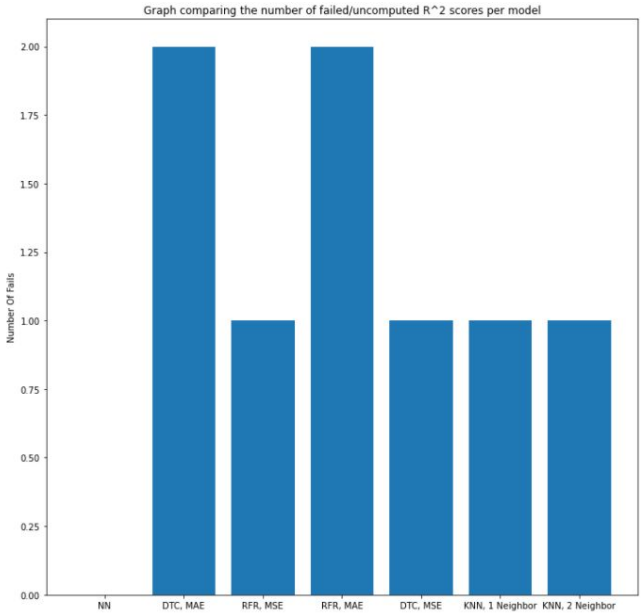

Fig 6: Graph comparing the number of fail cases across the 7 models

Conclusion

The goal of this work was to develop a model that could accurately predict how much water would be used in the drilling process and how much potential product would be lost due to flaring/venting. This paper compared 4 different algorithms: ANN, KNN, RFR, DTR. The mean weighted R^2 scores were calculated using a Monte-Carlo Cross Validation method, with the average R^2 score being computed over 50 runs. Using the pseudo-accuracy metric detailed earlier in the method and design section, the most accurate model would be the ANN. The ANN managed to produce an R^2 score for all of the 22 points once the data was sorted. However, this must be taken with a grain of salt. The average R^2 scores over the three outputs of the ANN are

-0.35, 0.09, 0.66. The poor R^2 scores seems to suggest that a different algorithm may be better to use. This means KNN (both 1 and 2 neighbors), RFR (with MSE criterion , and DTR (with MSE criterion) are the next best algorithms as they only have 1 fail case. The model with the best overall R^2 score is the RFR with MSE criterion with an average R^2 score of .58. The low R^2 scores across the models though likely would be improved with the increased availability of data. While some formations had over 10,000 data points, like the Bakken formation, some of the formations only had 3 data points (unknownXML) and other formations had only 20 data points available. Increasing the data availability should increase the overall robustness of all the models present. A similar workflow was done by Modaresi et al. which investigated the predictive powers of the same algorithms in forecasting monthly streamflow in the Karkheh dam in Iran. It was found that ANN should fashion the most robust model overall while KNN should create a similarly robust but not as strong model as ANN (Modaresi, 2017). A similar result was achieved in this work as ANN created a robust model that had no fail cases, and KNN was consistent in its modeling as well. Overall, taking a look at the data analyzed in this work, KNN and RFR are the two algorithms with the better predictive power. RFR returns slightly higher average R^2 scores compared to KNN (.58 vs .52) but can have computing issues due to the high usage of RAM during RFR training. This leads to significantly longer training times and could be inconvenient with larger datasets. Future usage of either model should depend on the size of the dataset.

While this work serves to be a start, there are still many other factors that are necessary to consider before coming to any significant conclusions. Quantities like mud weight, sand composition, fracking fluid used etc. all have a direct impact on the quantities analyzed in this experiment. Models like Least Squares Support Vector Machines can be developed and optimized to predict the outputs. In future comparison, using a Generalized Regression Neural Network or even optimizing the ANN should be compared to develop more models that could be more accurate than the other regression algorithms analyzed. The methodology used in this work can be used potentially in the oilfield while drilling to have more targeted drilling with more accurate results or for new standards in regulation in monitoring certain quantities like oil/gas production with a clear model of knowing how much will be lost.

References

Fig 1: US Energy Information Administration. (2016, March). *Trends in U.S Oil and Natural Gas Upstream Costs*. US Department of Energy. https://www.eia.gov/analysis/studies/drilling/pdf/upstream.pdf

Fig 2: TRADING ECONOMICS. (2010). *Natural gas | 1990-2020 Data | 2021-2022 Forecast | Price | Quote | Chart | Historical*. https://tradingeconomics.com/commodity/natural-gas

Modaresi, F., & Araghinejad, S. (2014). A Comparative Assessment of Support Vector Machines, Probabilistic Neural Networks, and K-Nearest Neighbor Algorithms for Water Quality Classification. Water Resources Management, 28(12), 4095–4111. https://doi.org/10.1007/s11269-014-0730-z

Scheyder, E. (2013, July 29). *Exclusive: Bakken flaring burns more than $100 million a month*. U.S. https://www.reuters.com/article/us-bakken-flaring/exclusive-bakken-flaring-burns-more-than-100-million-a-month-idUSBRE96S05320130729

Short-Term Energy Outlook - U.S. Energy Information Administration (EIA). (2020, November 10). Energy Information Administration. https://www.eia.gov/outlooks/steo/#:%7E:text=EIA%20expects%20drilling%20activity%20to,million%20b%2Fd%20in%202021.

US Department of Energy. (2019, June). *Natural Gas Flaring and Venting: State and Federal Regulatory Overview, Trends, and Impacts*. https://www.energy.gov/sites/prod/files/2019/08/f65/Natural%20Gas%20Flaring%20and%20Venting%20Report.pdf