Reduced Order Greenhouse Gas Flaring Estimation

Sharad Bharadwaj, Sumit Mitra
Energy Resources Engineering
Stanford University

Abstract— Global gas flaring is difficult to sense, a tremendous source of wasted revenue, and causes ecological problems. We use satellite sensors to predict gas flares' sizes. We tested regression and classification algorithms, along with anomaly detection using k-means, and found that linear regression and 2-class SVM are almost as good as the full-tilt sensor model produced by the National Oceanic Atmospheric Administration (NOAA).

I. Introduction

Natural gas flaring causes environmental damage and can be a significant source of lost revenue for oil producers. In this paper we apply various machine learning techniques to estimate global greenhouse gas flaring emissions. We will discuss the technical aspects of the machine learning bits further on, but first we give a brief introduction to gas flaring.

Figure 1: Gas flare in North Dakota [1]

In many oil wells, associated natural gas is produced alongside the oil. This gas can occur because of a variety of reasons; the drilled reservoir could contain both oil and gas, the gas could exist as a pressurized liquid in the reservoir and come out of solution while traveling in the wellbore, or there could be chemical processes that cause the gas to bubble out of solution. Whatever the reasons are, the problem is that natural gas at the surface is much harder to capture and safely store than liquid petroleum products are. In some locations, this natural gas can be safely captured and brought to market. However, in other areas the infrastructure necessary to adequately capture and transport the natural gas does not exist. In these cases, because of the safety hazard in having quantities of flammable gas floating around, the associated natural gas is flared [2]. An example of a natural gas flare can be seen in Figure 1.

Unfortunately little data exist to estimate the amount of global natural gas that is flared. Some jurisdictions require companies report their flaring emissions, but the data are often poor quality, not in the public domain, or are simply not reported. Therefore we aim to investigate the use of National Oceanic and Atmospheric Administration (NOAA) satellite data to estimate global greenhouse gas flaring. NOAA estimates CO_2 emissions, but this process requires hundreds of sensors and features; our goal is to replicate the NOAA CO_2 estimation results using a subset of their features [3].

II. Data, Features, and Preprocessing

Our data set consists of geotagged sensor readings from a NOAA satellite which performs infrared imaging of the earth. We have roughly 6 months of daily estimates of CO_2emissions (ground truth) derived from little over hundred different features.

The features included in the full dataset are various sensors aboard the satellite, and some manufactured features such as transmissivity of sensor readings. To perform our analysis, we chose to train on a subset of the total features: the infrared measurements. There are 8 different infrared spectral band sensors onboard the satellite, with each sensor measuring the intensity

of light received from a different band of infrared light. Gas flares burn at a significantly hotter temperature than the background Earth, so they emit light in the near-far infrared, precisely those spectrums which the NOAA infrared sensors pick up [3]. Figure 2 shows an example of one data point, with the NOAA estimate of CO_2 highlighted in peach and the 8 sensor readings highlighted in blue.

Figure 2: Example truncated data point

To preprocess the dataset, we performed a first round of manual cleaning. The ground truth reported CO_2 values (the CO_2 equivalent of the burned natural gas) are real numbers representing kg/s flow rates; there are a significant portion of the data which have physically impossible CO_2 values, such as 10^6 kg/s; these impossible CO_2 values were thrown out, leaving us with a data set containing roughly 300,000 points.

Table 1: PCA for sensor readings

Principal Component	% Variance Explained
1	22.3
2	18.3
3	15.7
4	12.3
5	9.8
6	8.6
7	7.4
8	5.6

As a first pass at further shrinking the range of input data, we ran Principal Components Analysis (PCA) on the set of sensor measurements. We calculated the percent of variance that each principal component explained; Table 1 contains the output of our analysis. It is evident that no single set of principal components explains the vast majority of our data, so we decided not to perform data shrinkage and move forward by using all 8 sensor readings in our analysis. We feel this is appropriate also because for our dataset there are vastly more observations than features, so there wasn't a pressing need to shrink our feature space for analysis purposes.

III. Models

The following models were utilized for this investigation. All training was done on the first third of the data and testing was done on the last two thirds.

A. *Linear Regression*

We chose linear regression to determine how accurately we could predict exact CO_2 emissions values. The 8 sensor features were used with the built in MATLAB® linear regression model that used the Moore-Penrose pseudoinverse $\beta = (X^T X)^{-1} X^T y$ to calculate the regression parameters [4]. We first ran this algorithm with an intercept term (x_0) and noticed a 75% error in the model, which seemed unusually high. After further looking at the type of data that was reported, we realized that most of our CO_2 readings were centered on 0. This meant a more realistic model would include an intercept term of 0. We also looked into running higher order polynomial regression models but decided that because we didn't know the exact structure of the data it would be difficult to ensure we weren't overfitting it and thus decided to only work with linear regression.

B. *SVM*

After running linear regression, we wanted to see if we could reduce the training and test error significantly through classification. We decided to implement a multi class SVM model that would split CO_2 emissions into small, medium and large bins. However, because the small and medium values were too similar, we were not able to linearly separate them for any kernel choice. We then focused on a 2 class model that separated small

and large CO_2 emissions. We used MATLAB®'s svmtest and svmclassify functions, tools that were based on a soft-margin optimization problem:

Equation: Soft-margin SVM optimization formulation

$$\begin{aligned} \min_{\mathbf{w},b,\boldsymbol{\xi}} \quad & \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{l}\xi_i \\ \text{subject to} \quad & y_i(\mathbf{w}^T\phi(\mathbf{x}_i) + b) \geq 1 - \xi_i, \\ & \xi_i \geq 0. \end{aligned}$$

We separated small and large values based on a threshold of .5 kg/s. We performed appropriate transformations to change our training y values into -1 and 1 and we used the default values for the regularization and error tolerance terms.

There were two main issues that we faced during this classification. First, threshold values to separate these bins were not given in literature and thus we did not have an accurate way to separate these values. We decided to conduct a sensitivity analysis on the threshold to handle this problem. We wanted to understand how the error changed when we varied the threshold value from .3 to 1 kg/s in our test set**.** Second, our data was not separable using a linear kernel. Because we didn't completely understand the entire structure of the data set, it was difficult to determine which kernel would best fit our needs. After testing our classification with multiple kernels, we thought the quadratic kernel was the optimal choice because it performed well and was only one order higher than linear, reducing potential overfitting. However, further research should be conducted to determine the most appropriate kernel for this dataset and features.

C. K-means

Despite our initial cleaning of the data, there were still anomalies present based on the histogram depicted in Figure 3. Notice the vast majority of the data clumped in the small CO_2 emissions values. We think some portion of those points corresponding to the long, sparse tail of high CO_2 emissions values are anomalies, so we investigated k-means for anomaly detection.

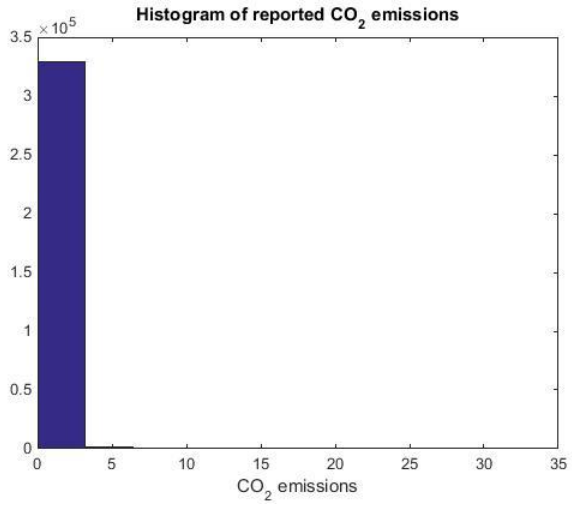

Figure 3: Histogram of reported carbon dioxide

We used K-means to detect these potential anomalies and then reran our line regression to determine whether or not these anomalies significantly affected our results. We didn't rerun SVM because the error was already relatively low and removing a few points wouldn't affect the model significantly for the classification. Six clusters were chosen to represent our data. Since we didn't know how to determine how many potential anomalies there were, we ran a sensitivity analysis on removing clusters that contained 100 to 6000 values and then reran our algorithms. In addition, because K-means only finds a local optima, we ran this 25 times and chose the clusters with the lowest cost function $J(c,\mu) = \sum_{i=1}^{m}\left\|x^{(i)} - \mu_{c^{(i)}}\right\|^2$ to represent our cluster segmentation.

IV. Results

Figure 4 illustrates the sensitivity of the SVM test error to changes in the CO_2 threshold. As expected there is a trend of smaller proportion misclassified for the test set as the threshold increases because there are fewer CO_2 equivalent values that are that large. We used a threshold of .5 in our results, which represents the largest potential misclassification of our SVM, to put an upper bound on the misclassification error with this algorithm.

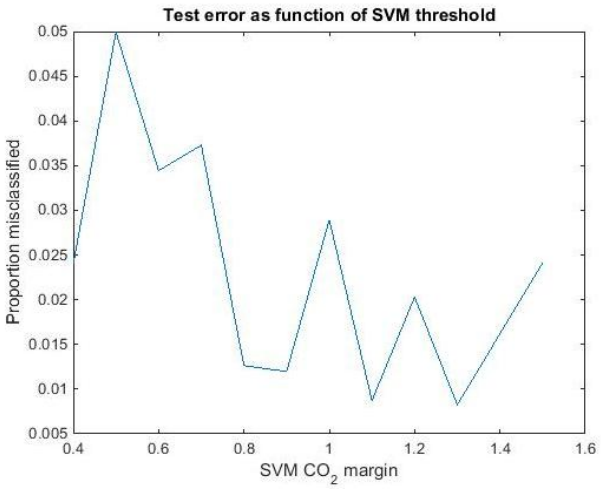

Figure 4: Sensitivity of SVM error to threshold

Table 2 illustrates sample sizes, training, and testing errors for each algorithm. The linear regression model's error is halved when the artificial intercept is removed. SVM has the lowest error at 5%. After running k-means and removing 100 potential anomalies, linear regression had same test error. We were unable to run SVM after k-means anomaly detection because our computer ran out of memory. In the future, we propose running PCA to determine the principal components and using that for SVM classification.

Table 2: Summary results

MODEL	TRAIN SIZE	TEST SIZE	TRAIN ERROR	TEST ERROR
Linear with Intercept	91,355	240,057	65%	68%
Linear without Intercept	91,355	240,057	33%	33%
2-class SVM	91,355	240,057	4%	5%
Multiclass SVM	91,355	240,057	N/A	N/A
Linear k-means	110,440	220,881	32%	32%

Figure 5 shows the sensitivity of linear regression to the number of potential anomalies removed through k-means.

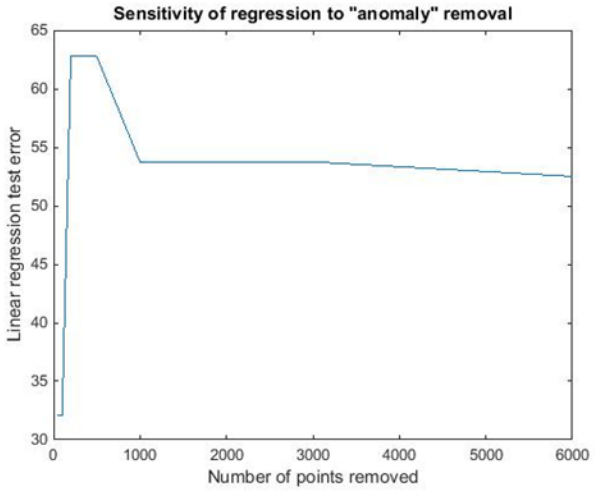

Figure 5: Linear regression anomaly removal sensitivity

Based on the histogram of CO_2 emissions, we expected there to be at most 100-200 additional anomaly points. At this range we noticed that the linear regression error hardly changes which reveals that these points are not significantly impacting our results. There is a sharp increase in the test error when approximately 300 points are removed. This indicates that there is a small subset of our data that significantly affects our results but are not anomalies. As we increase the number of points removed, we see that the error stabilizes. Once we have removed all points outside the largest cluster, removing any more points has little effect as the volume of training data in the large cluster is so high (Table 3).

Table 3: Size of k-means clusters

Cluster	Points in cluster
1	325426
2	4974
3	178
4	545
5	198
6	91

V. Discussion and Conclusion

As we see in Table 2, linear regression without an intercept term has a lower test error than linear regression with an intercept term. We suspect this is due to the physical model underlying our regression. If a linear model is accurate, the CO_2 prediction from an input of 0 across all sensors "should" be 0 - i.e. the model should predict that f(0) = 0. This makes sense; if the infrared sensors are reporting no light in their spectral bands, which means that there is nothing creating heat to be picked up by the sensors, which means there should be no underlying gas flare.

For the SVM, we see a strong classifier with respect to separating between small and large flares, classified as those with flaring intensities larger than 0.5 kg/s. Unfortunately SVM is not successful at classifying between small/medium flares as it seems the dataset is simply too dense in this region and there does not exist a separating hyperplane. Therefore, perhaps regulators can use SVM as a tool to identify the worst polluters in a given region and then perform more sensitive analyses, either by drilling deeper into satellite imagery or using physical sensors on-site, to quantify the emissions more finely.

Given more time, the next steps would be to obtain a dataset of the Bakken oil field that has true CO_2 emissions as reported by oilfield operators. We would train our algorithms with these values instead of the satellite estimates and then test to determine how accurate NOAA satellite values are with respect to ground truth CO_2 emissions worldwide. Next, we would want to introduce a penalty function to sensor readings that took into consideration the cloud covering and see its effect on the overall performance of our algorithms.

References

[1] K. Cenedo, Corbis. National Geographic News, May 22, 2014.

[2] A. O. Bisong. "Effects of Natural Gas Flaring on Climate Change in Nigeria," SPE Nigeria ATCE, August 5-7, 2014.

[3] C. Elvidge et al, "VIIRS Nightfire: Satellite Pyrometry at Night,"Remote Sensing vol. 5(9) pp 4423-4449, September 2013.

[4] J.C.A. Barata, M.S. Hussein, "The Moore-Penrose Pseudoinverse. A Tutorial Review of the Theory," Instituto de Fisica, Universidade de Sao Paulo. arxiv.org:1110.6882v1