Oil and Gas Well Initial Production

Samuel Rogers

**Introduction**

The initial production of hydrocarbons from an oil and gas (O&G) well is likely the largest production seen for the life of the well. It’s also the most variable data point. Prior to actual production, future production of a well can be forecasted using type curves, usually generated using analogous wells tapping into the same reservoir. This means before a well is even drilled, investors can quantify the value of the well based on location. Once drilled, the initial production of oil/gas tells the reservoir engineer where to start their type curve which can form a fair estimate of the economic prosperity of the well. Using surrounding wells as variables in Decline Curve Analysis is an everyday practice in the O&G industry.

**Objective**

This project focuses on predicting, with quantifiable accuracy, the first month’s production for the purpose of fitting to a decline curve. Reservoir engineers can make estimates as to where the well might begin, but these can vary widely. Variables from neighboring wells contribute to this analysis such as reservoir, lateral length, depth, fracture fluid quantities, and various completion techniques. Using these variables, the initial production of the well will be predicted.

**Data Description**

The area of interest for this project was simplified to Grady County in Oklahoma. The scope includes only active, currently producing, horizontal wells. After cleaning the data, this amounted to 626 wells being tested. The variables from all the datasets began at 283 and was dwindled down to 10. From industry knowledge the attributes most important in prediction are operator, reservoir, production type, measured depth, completed interval, fracture treatment, flowing method, lateral length, frac fluid, first month oil and gas production, and proppant. An explanation of these terms can be found in **Table 1**. Data used for this project came from three sources: FracFocus, Oseberg dataStream and Enverus. Oseberg and Enverus are subscription-based data brokerage companies, while FracFocus is open to the public. This data is all technically public domain, but much of it is collected from handwritten documents.

In terms of scalability, each data source arrives in SQL so the project could be scaled to multi-county or state interpretations using SQL to house and manipulate the data. The real-world application of this project is to use FracFocus and Oseberg dataStream to pull permit information before a well has begun production and predict the y-output.

|  |  |
| --- | --- |
| Attribute | Description |
| API | Identifier of well |
| Operator | Company serving as overall manager of drilling the well |
| Reservoir | Subsurface body of rock containing hydrocarbons able to be produced |
| Prod\_Type | Type of hydrocarbons being produced |
| MD | Measured depth is the distance traveled underground (ft) |
| Completed\_Interval | Portion of horizontal well open and producing from the reservoir (ft) |
| Fracture\_Treatment | Chemicals, fluid volume, proppant quantiy used in fracture (unstructured format) |
| Pumping\_Flowing | Production method of oil and gas from reservoir |
| Lateral\_Length | Length of horizontal well (ft) |
| TVD | True Vertical Depth is distance from surface to deepest point (ft) |
| Frac\_Fluid\_W | Fluid injected under high pressure into hydrocarbon bearing rock to increase flow of oil and gas (gallon) |
| F\_m\_BOE | First month oil and gas quantities converted to equivalent energy produced by barrel of crude (barrel) |
| Proppant | Sand used to keep fractures ajar (lbs) |

**Table 1**: Data Terms

**Data Preparation**

1. Joining Datasets

These datasets have many fields in common, the most uniform connection between them being their unique identifier, the API. Before joining, specific variables were selected from each dataset, with preference given to the most complete ones. Datasets were joined starting with datasets with the fewest records first to create the most complete dataset.

1. Removing Inconsistent Records

Oil and gas data is notoriously dirty and incomplete, especially when combining data from numerous operators with different reporting methods. This is the why many variables and rows were excluded, to maintain a clean crisp dataset. One idea was to use K-Nearest neighbor to populate missing values which aren’t the y output IP. It was found that NA values were usually accompanied by several NA values in the same row. This made it near impossible to predict using neighboring attributes, not to mention the lack of accuracy. All NA rows were excluded.

1. Calculating y-output: Initial Production (IP)

Gas and oil can be added together by dividing gas by 6 when the gas is in MCF (million cubic feet). The sum of these will equal BOE (barrel oil equivalent), which equates to the energy produced when burning a barrel of crude oil. Unrefined hydrocarbons burn at different intensities, so this is a rough conversion frequently used by the oil and gas industry. This conversion was used to convert and sum the first month production of oil and gas (y output).

1. Prediction Attributes

Attributes used for prediction were the Operator, Reservoir, Production Type, Measured Depth, Completed Interval, Pumping Method, Lateral Length, True Vertical Depth, Frac Fluid Volume, First Month BOE. Proppant, while a valuable predictive attribute, caused a large decrease in prediction accuracy. It required using ‘sub’ in R to extract proppant quantity from an unstructured text field after the word “POUNDS”, then manually converting 60+ records reporting proppant mass in MM (meaning 1000\*1000) to numeric values. It’s suspected that due to the lack of consistent reporting methods from operators the proppant values were off.

1. Discretize Data

Character data was converted to factor, then the numeric First Month BOE was broken into equally distributed bins and formatted as a factor (**Table 2**)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| First Month BOE (barrels) | | | | |
| Trickle | Low | Medium | High | Super\_Max |
| 2 – 1,350 | 1,350 – 4,670 | 4,670 – 10,000 | 10,000 – 18,100 | 18,100 – 69,400 |

**Table 2**: Discretized First Month BOE separated into bins of equal frequency.

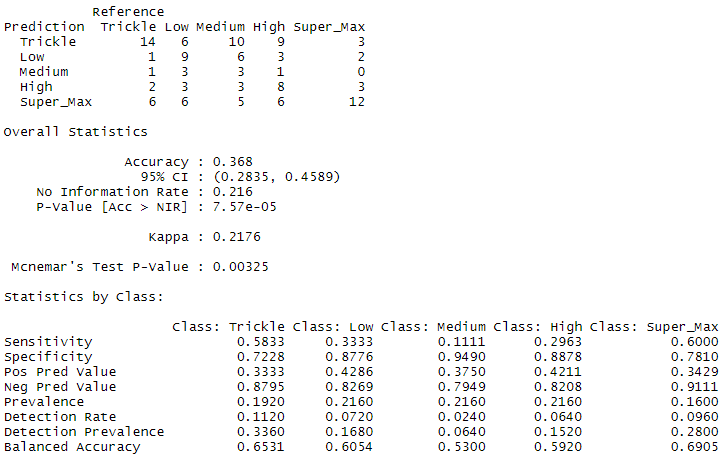
1. Preparing Train and Test Datasets

The Clean\_Grady dataset was preprocessed by scaling then 80% was randomly selected for training data and remaining 20% used for testing data.

**Results**

1. Support Vector Machine: Linear

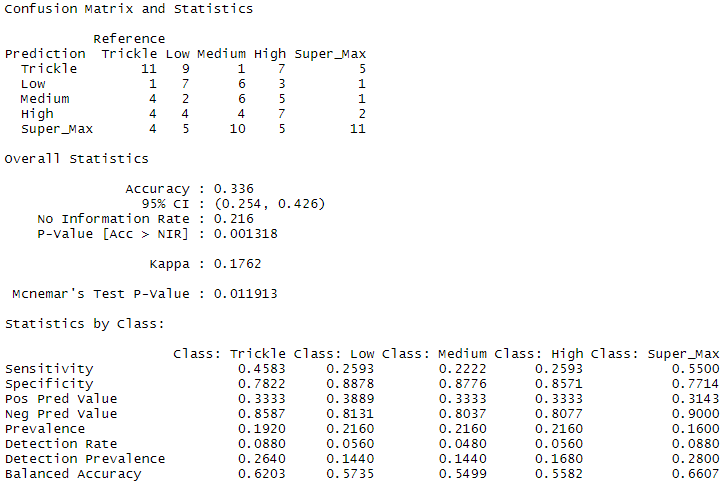
The first SVM used linear kernel with a tuning parameter. The preferred cost was 5 and gamma value was 0.001. The results can be seen in (**Figure 1**) below. 36.8% accuracy is one of the higher accuracies. The most concerning issues in the reference prediction table is the Super\_Max value. It has a wide array of classes it’s incorrectly predicting into; the worst was 6 times is incorrectly chose the Trickle class instead of Super\_Max. Discretions between neighboring is not as concerning due to the nature of the bins, but broad jumps such as this one is concerning.



**Figure 1:** SVM Linear Kernel using Tuning Parameter

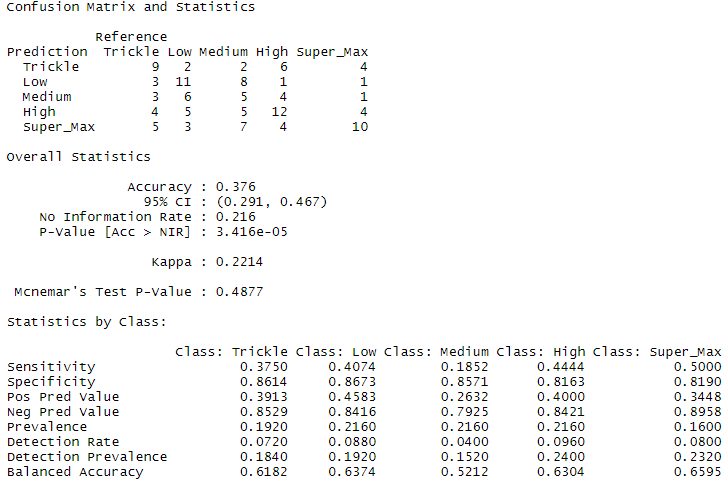
2. SVM: Radial

The radial kernel received a 33.6% accuracy using a cost of 20 and gamma of 0.1. It was able to better predict Super\_Max and Trickle correctly as compared to the linear model. The sensitivity across all class average to 35% with specificity at 84% (**Figure 2**). This indicates it is better at predicting when it’s not the correct class, than predicting the correct class. When tuning was applied to the radial model, the results decreased greatly so it was excluded.



**Figure 2**: SVM Radial using Default

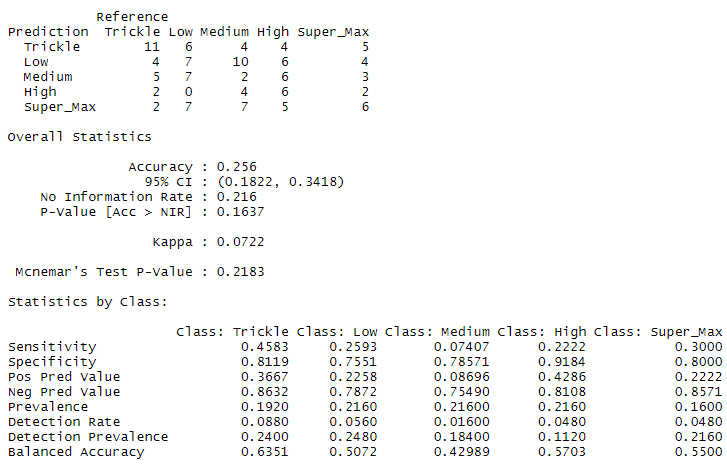
3. Random Forest

The 2nd best results came from using this ensemble prediction method. RF had an accuracy of 37.6% (**Figure 3**) and increased the sensitivity values in the middle classes (Low, Medium & High) as compared to the linear and radial SVMs. This model uses 200 trees and 5 nodes and performed much better than greater quantities of tree and node combinations.

**Figure 3**: Random Forest using 200 trees and 5 node size

4. K-Nearest Neighbor

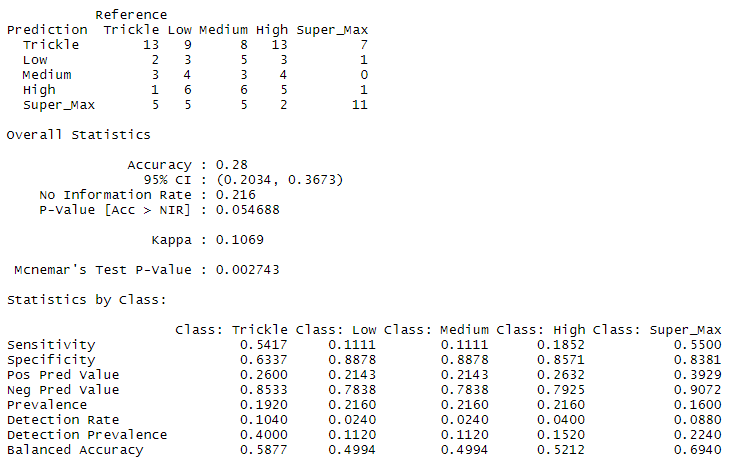
In taking out factor data such as Operator, Reservoir, Prod\_Type and Pumping\_Flowing, this model was able to calculate the predictor variable using new training data. Unfortunately, this was the least accurate model at 25.6% (**Figure 4**). Of the neighbors considered, 20 performed the best.



**Figure 4**: K-Nearest Neighbor with k=20

5. Naïve Bayes

This model was built using the R package e1071. This model had the lowest sensitivity values at 11.11% for both Low and Medium. It’s not surprising that the accuracy is only 28% (**Figure 5**). Seeing as the true positive rate (sensitivity) is more important than the true negative rate (specificity) in this project’s data problem, this NB algorithm was considered the worst predictor.

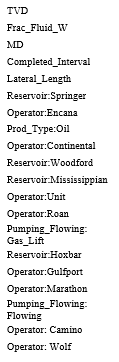
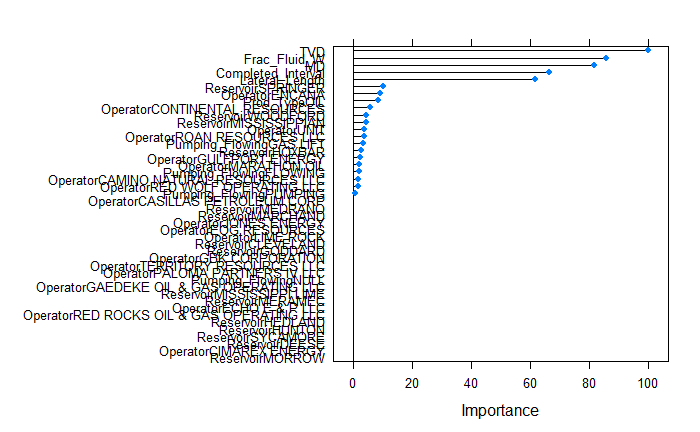


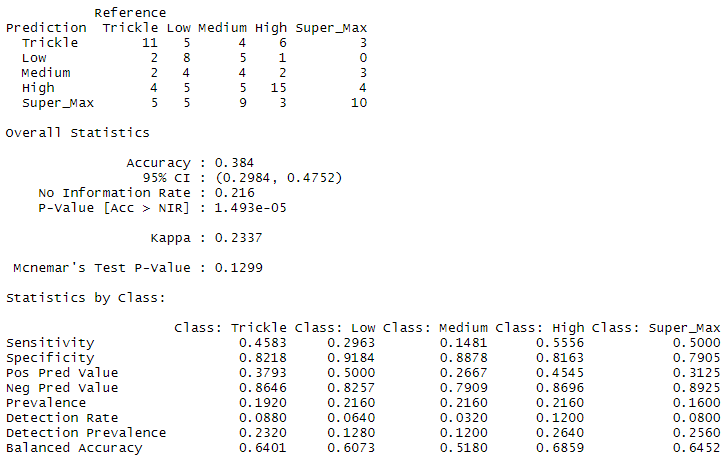
**Figure 5**: Naive Bayes model e1071

6. Gradient Boosted Machine

Using a cross validation of 15 resulted in the greatest accuracy of all the models at 38.4% (**Figure 6**). The most important attributes for this model can be seen in Figure 3 with the top 5 being TVD, Frac\_Fluid\_W, MD, Completed\_Interval, and lastly Lateral\_Length. After these 5, the relative importance drops from 60%-100% down to 0%-15% (**Figure 7**). It’s not able to quantify factors like a specific reservoir or operator so this makes sense that those have such low importance levels.

**Figure 7**: GBM Importance Chart





**Figure 8**: GBM using CV = 15

The results discussed above can be seen summarized in **Table 3**.

**Table 3**: Summary of models with respective parameters and accuracies.

|  |  |  |
| --- | --- | --- |
| Model | Parameters | Accuracy |
| SVM | Linear; c = 1, g = 0.001 | 36.80% |
| SVM | Radial; c = 20, g = 0.1 | 33.60% |
| Random Forest | Tree= 200, node size = 5 | 37.60% |
| KNN | k=20 | 25.60% |
| Naïve Bayes | Default | 28.00% |
| GBM | CV =20 | 38.40% |

**Conclusion**

The ensemble models (Random Forest and GBM) performed the best out of all the methods attempted at 37.6% and 38.4% respectively. The GBM provided valuable insight into what variables were the most important, particularly the TVD being number 1. It was followed quickly behind by the Frac\_Fluid\_w, which means that if Proppant was better reported by the operator, it likely would have been a very valuable predictor, as suspected. The linear and radial kernels for SVMs performed the 2nd best overall in terms of accuracy. The sensitivity performance was the best in the SVMs, which is one of the features believed to be the most important.

KNN and Naïve Bayes on the other hand provided the worst accuracy and lowest sensitivity. KNN reported sensitivity for the Medium class at 7%, while Naïve Bayes had the middle classes reporting between 11% - 18%.

This project predicted on data that is notoriously difficult to predict on, as it’s dirty and requires a lot of industry knowledge and manipulation to work properly. While the overall results were not very high, it is arguably more accurate than current methods employed. Current methods don’t have a quantifiable accuracy except to use the average of the points wells in the neighboring area. Perhaps a better algorithm would include an attribute that included the distance between wells. This is easier said than done but is an area with which to further investigate.