Capstone Project 1 Summary

Julia Hu

July 2020

1. Introduction

- Well logs are important for subsurface characterization
- However it is difficult to obtain sonic logs directly
- The goal of the “SPWLA’s 1st Petrophysical Data-Driven Analytics Contest” is to develop data-driven models to generate synthetic compressional and shear travel-time logs (DTC and DTS, respectively) in Well #2.

2. Training Dataset

- All the values equals to -999 are marked as missing values.
- - CAL - Caliper, unit in Inch,
- - CNC - Neutron, unit in dec
- - GR - Gamma Ray, unit in API
- - HRD - Deep Resisitivity, unit in Ohm per meter,
- - HRM - Medium Resistivity, unit in Ohm per meter,
- - PE - Photo-electric Factor, unit in Barn,
- - ZDEN - Density, unit in Gram per cubit meter,
- - DTC - Compressional Travel-time, unit in nanosecond per foot,
- - DTS - Shear Travel-time, unit in nanosecond per foot,
- The test data has all features that you used in the train dataset, except the two sonic curves DTC and DTS.

3. Data Exploration

	CAL	CNC	GR	HRD	HRM	PE	ZDEN	DTC	DTS
count	20525.000000	20525.000000	20525.000000	20525.000000	20525.000000	20525.000000	20525.000000	20525.000000	20525.000000
mean	8.426679	0.274416	49.889253	2.598719	5.835466	3.833792	2.410734	88.312221	182.051067
std	1.845912	3.062495	54.811017	3.465665	422.449589	4.375818	0.181713	23.542419	84.670122
min	5.930400	0.014500	1.038900	0.123600	0.134100	-0.023200	0.680600	49.970500	80.580400
25%	6.629100	0.120300	16.036800	0.810000	0.797300	0.049800	2.236100	70.423100	127.148800
50%	8.578100	0.187700	37.498000	1.814900	1.829300	3.287800	2.466500	79.695400	142.678500
75%	8.671900	0.329000	61.140700	3.337400	3.463300	7.061300	2.563700	102.482800	192.757800
max	21.064200	365.885000	1470.253400	206.718200	60467.761700	28.106400	3.259700	155.980300	487.438400

Figure 1: Description of the dataset, about its statistic values.

4. Pair Plot

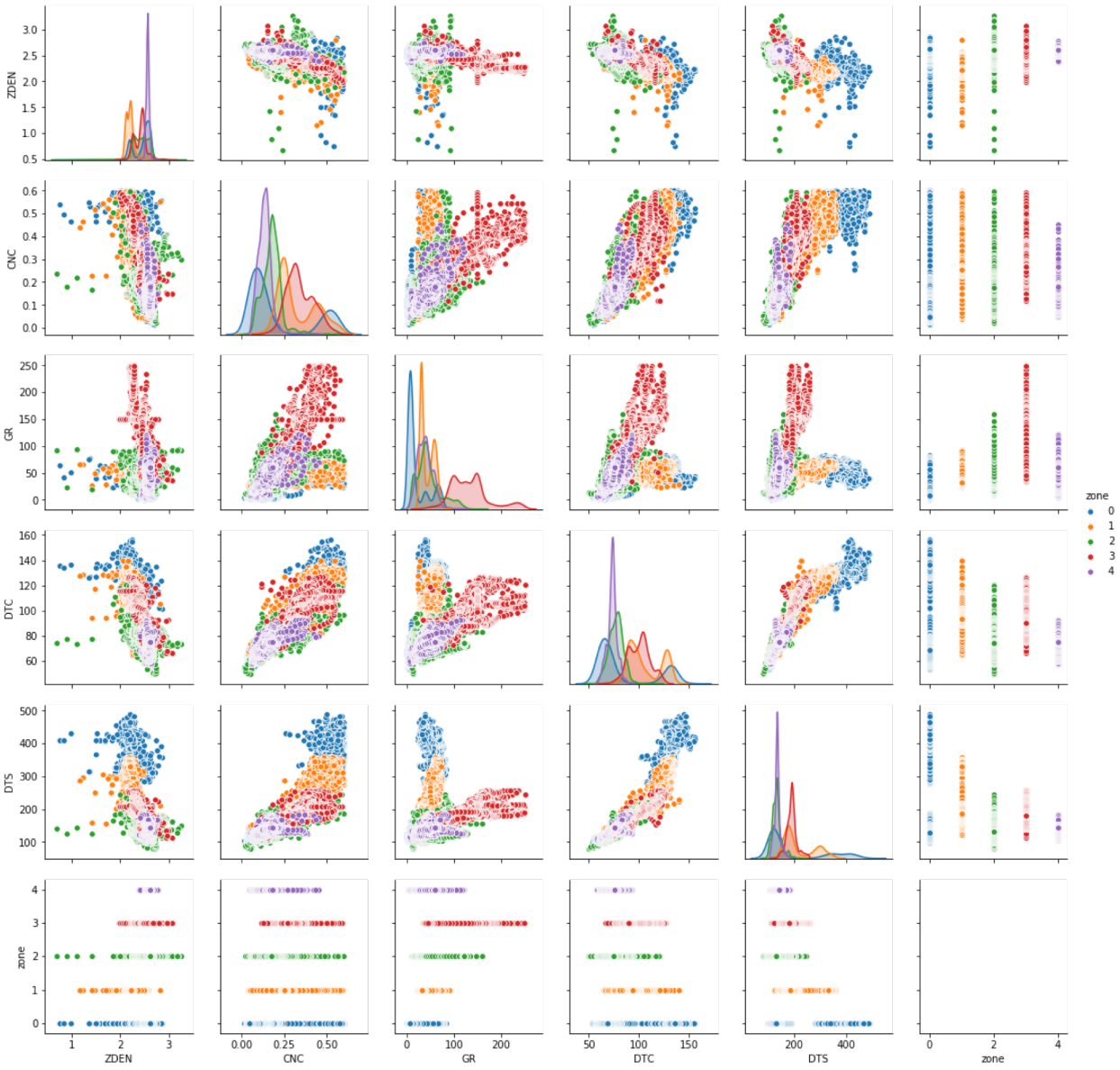


Figure 2: Pair plot comparisons of the density log and DTC log.

5. Sediment Based Zone Separation

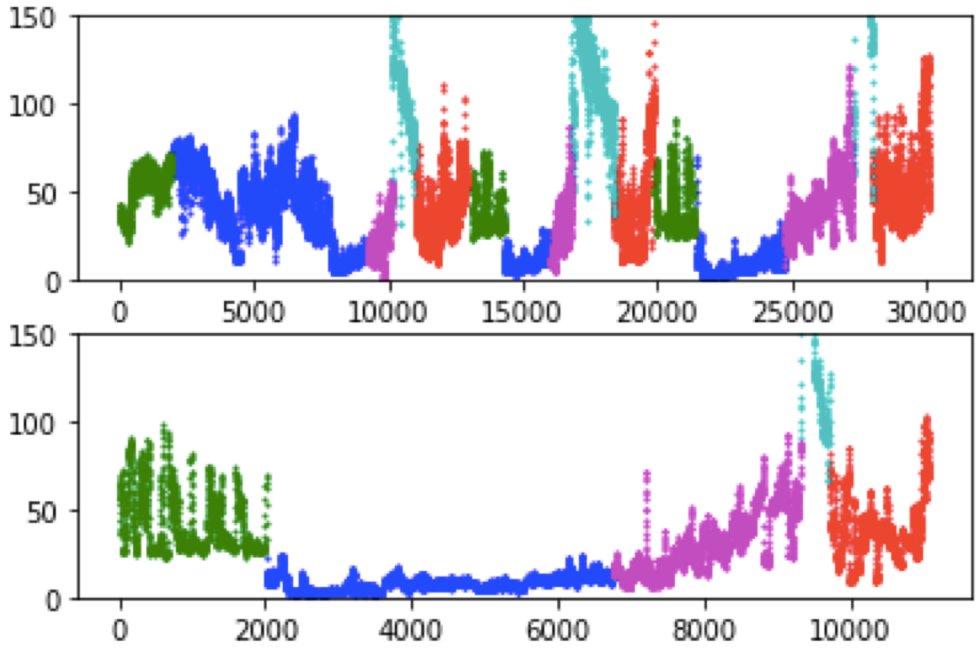


Figure 3: Top figure shows training data rezoning. Bottom figure shows test data rezoning distribution.

6. Fill in Missing Data

	Missing Values
CAL	510
CNC	735
GR	254
HRD	385
HRM	385
PE	579
ZDEN	681
DTS	4865
DTC	4054

Figure 4: Value counts of missing values in logs.

7. Plot of Missing Values in Well Logs

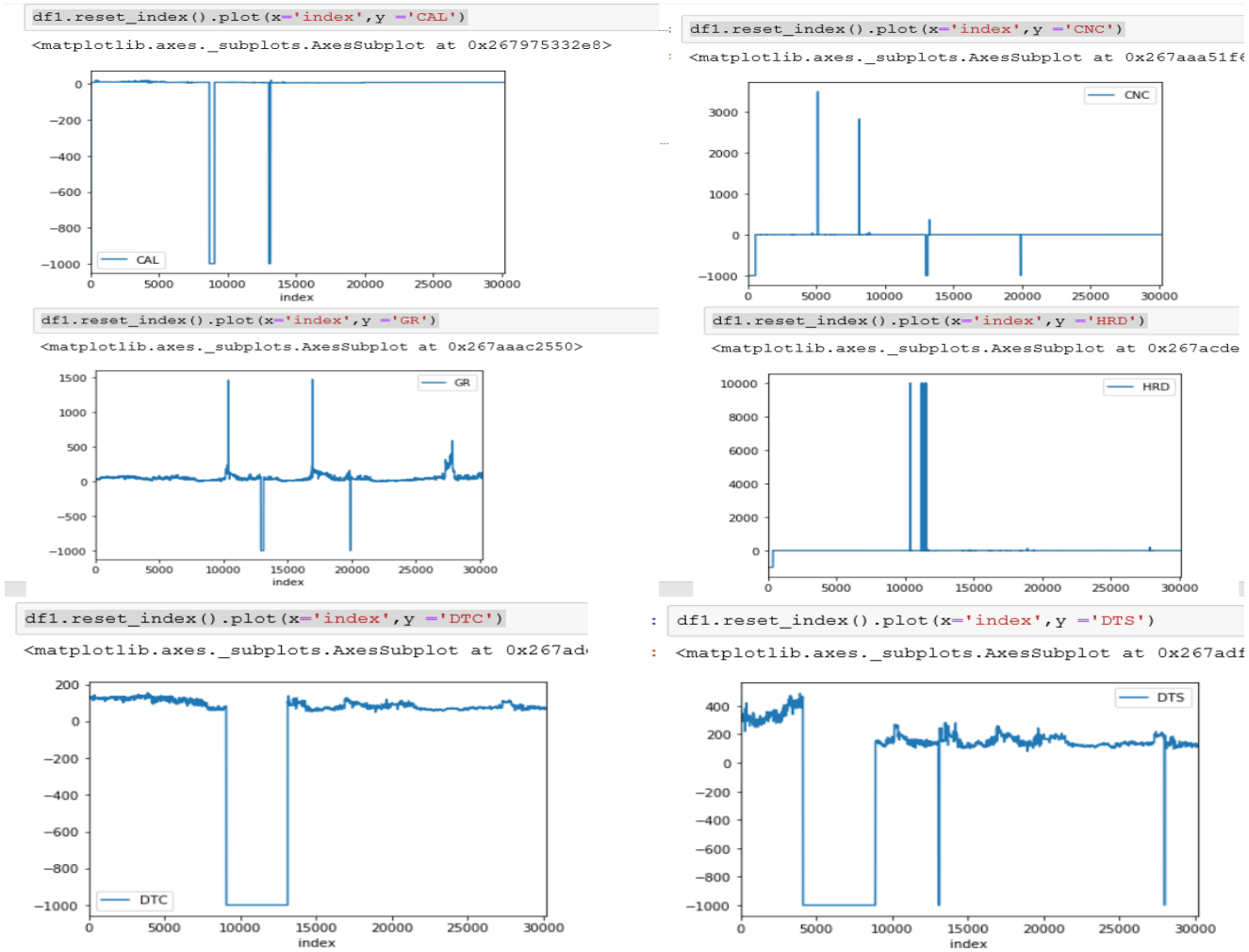


Figure 5: Visualization of missing values in logs.

8. First fill missing values by using Peudo-labeling method

- First, ML modeling based approach is used to fill in missing values. This approach uses target without missing values to develop a ML model, such as XGBoost. Then this model is used to fill in missing values of the target.
- However the filled in missing values do not improve the validation accuracy. There is no big difference after filled DTC missing values. From the RMSE values measured on validation data, the result is similar to that with missing values (2.8).

8. Visualization of Well Log after Filled Missing Values

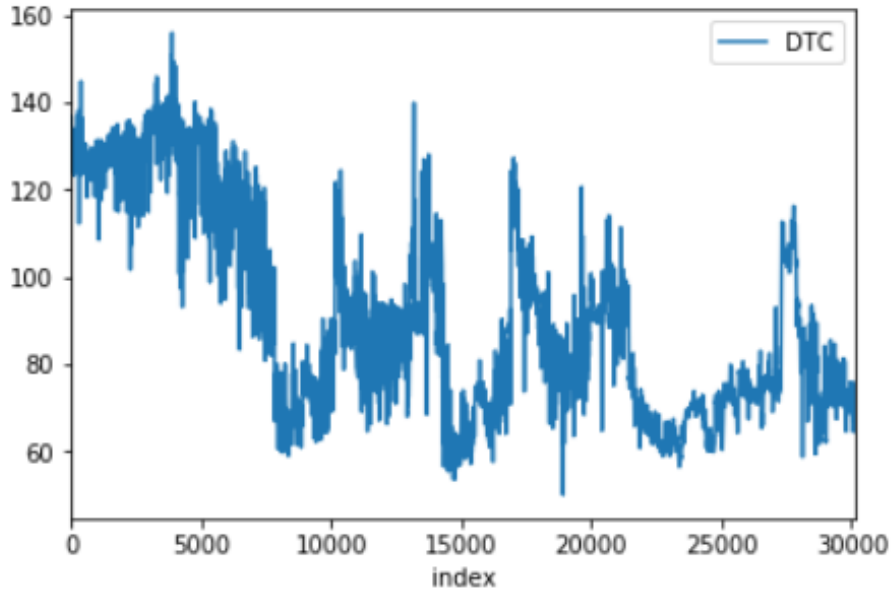


Figure 5: Figure of DTC log with filled in missing values.

DTC RMSE without missing values fill in		DTC RMSE with missing values fill in	
Predicted from features only	2.73	Predicted from features only	2.66

DTS RMSE without missing values fill in		DTS RMSE with missing values fill in	
Predicted from features only	7.17	Predicted from features only	7.66

Figure 6: RMSE improvement for DTC and DTS.

9. Accuracy of Pseudo-label method for filling missing units

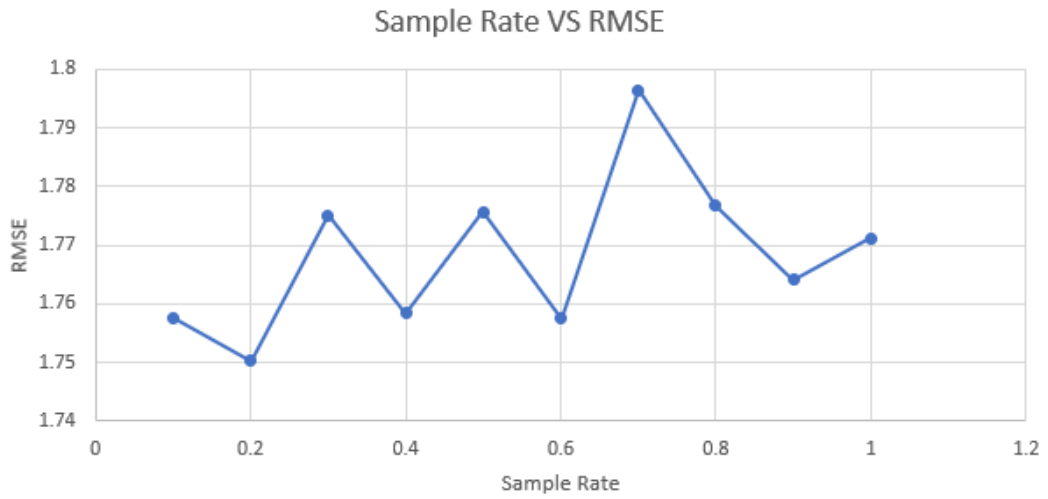


Figure 7: The accuracy against varying percentage of psudolabeled data.

Feature Engineering Method

- Two features were engineered to improve accuracy: (1) Apply logarithm. (2) Use Gradient method to accentuate difference.

```
df1['HRD_log']=np.log(df1.HRD.values)
df2['HRD_log']=np.log(df2.HRD.values)
df1['HRM_log']=np.log(df1.HRM.values)
df2['HRM_log']=np.log(df2.HRM.values)

# Calculate the gradients of input logs
# Interpolated well logs should have similar gradients
for i in df1.keys():
    df1[i+'grad']=np.gradient(df1[i].values)
for i in df2.keys():
    df2[i+'grad']=np.gradient(df2[i].values)
#Observe the patterns of logs
```

Figure 8: Feature engineering to improve model.

10. ML model Building

- After preprocessing a simple fully connected ANN network, with very few nodes, was applied. The first layer has 24 nodes, second layer 8 nodes, and then connected to a dense layer before dropout.

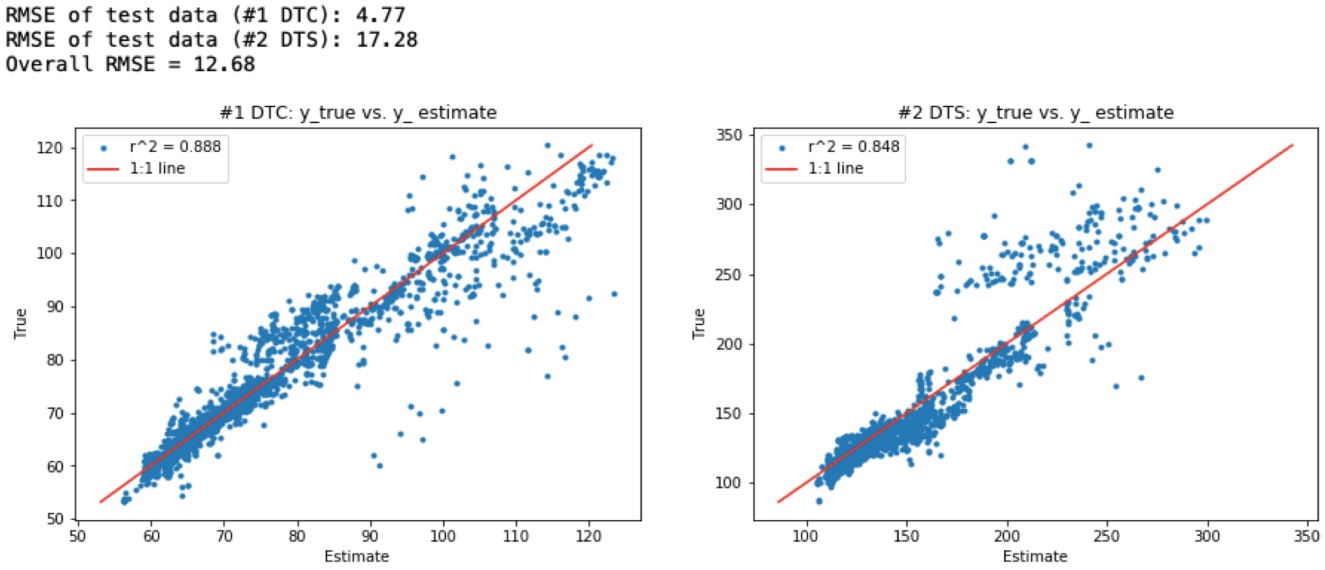


Figure 9: Comparison of predicted data and test data shows good agreement.

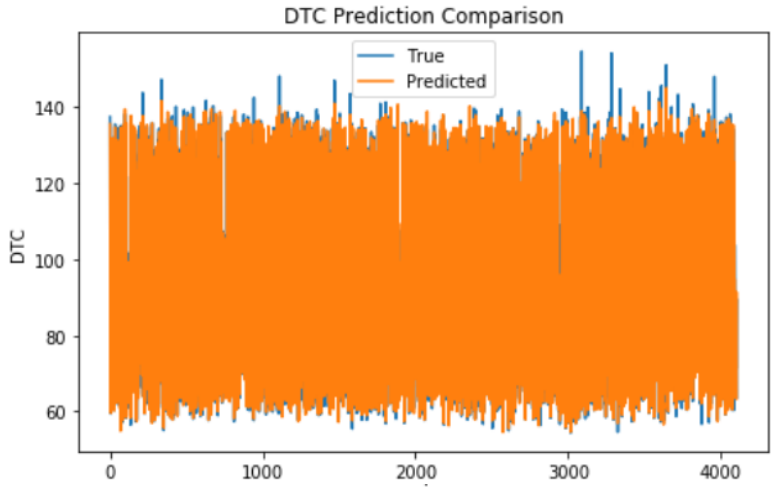
11. ML model comparison

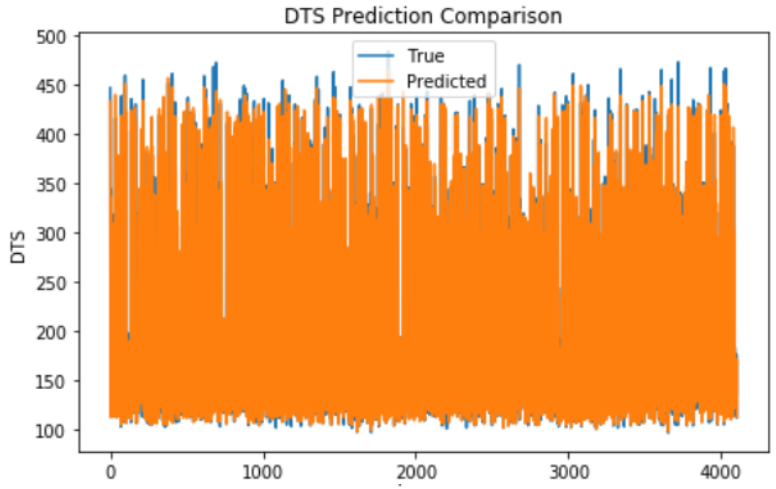
- XGBoost was also tested in this project, and the following table shows hyperparameter tuned with XGBoost for performance improvement.

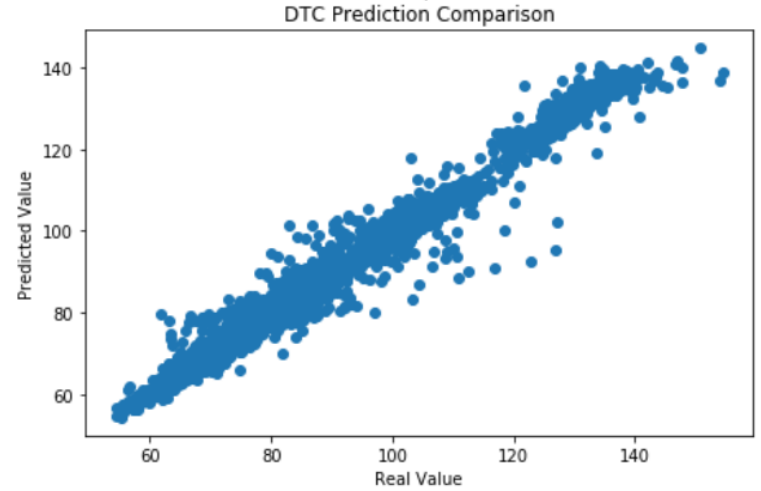
	score	max_depth	min_child_weight	learning_rate	n_estimators	colsample_bytree	subsample	gamma
7	0.989576	17	2	0.05	180	0.9	0.9	0.6
6	0.989536	17	2	0.05	160	0.9	0.9	0.6
5	0.989448	17	2	0.05	140	0.9	0.9	0.6
4	0.989368	17	2	0.05	130	0.9	0.9	0.6
3	0.989231	17	2	0.05	120	0.9	0.9	0.6
2	0.988953	17	2	0.05	110	0.9	0.9	0.6
1	0.988422	17	2	0.05	100	0.9	0.9	0.6
0	0.984736	17	2	0.05	80	0.9	0.9	0.6

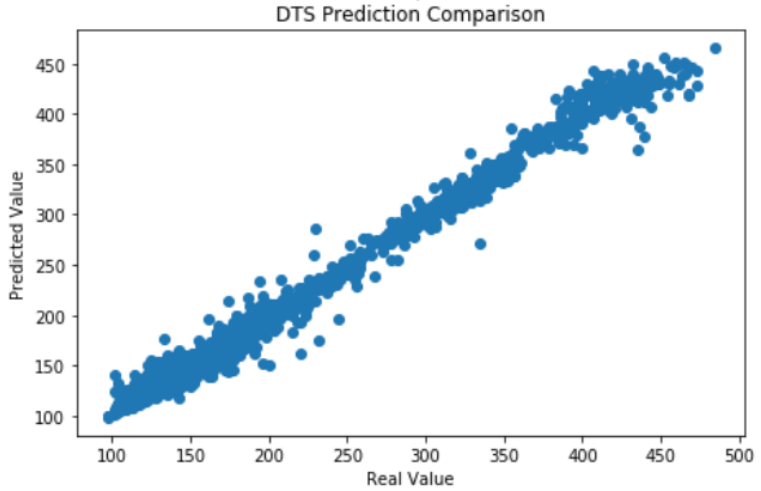
12. ML Best Algorithm R2 = 0.99

```
R2: 0.9901
Root Mean Square Error is: 5.43997
```









13. LSTM Alone

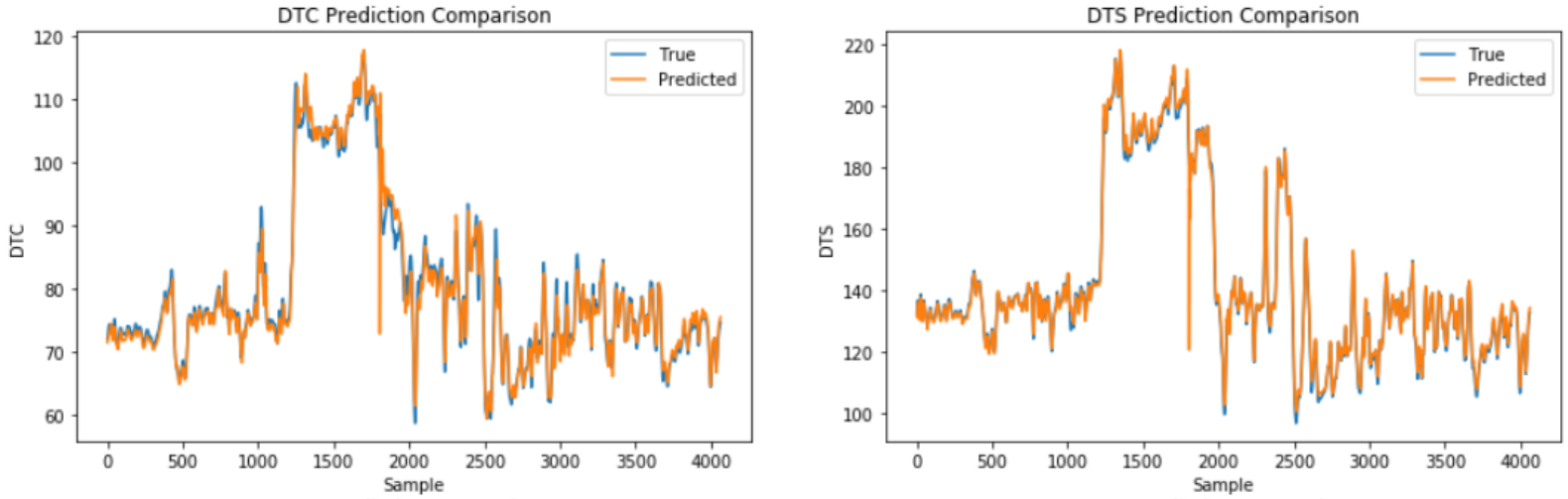
- No change on Min-Max scaler
- Used re-shape to prepare data for Multi-variate LSTM
- Two layer LSTM, with each layer 256 neurons, history window size 40, epoch 100, Relu activation, MSE loss.
- Initial result is promising, RMSE=2.84006, R2=0.9679.

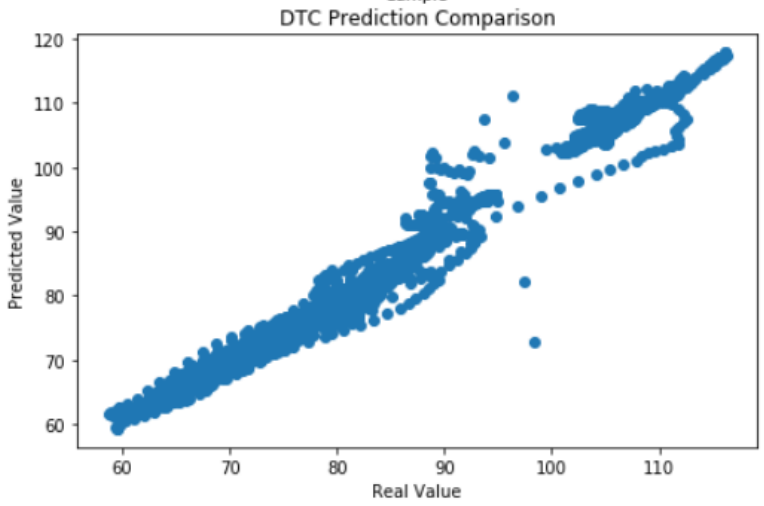
CNN + LSTM

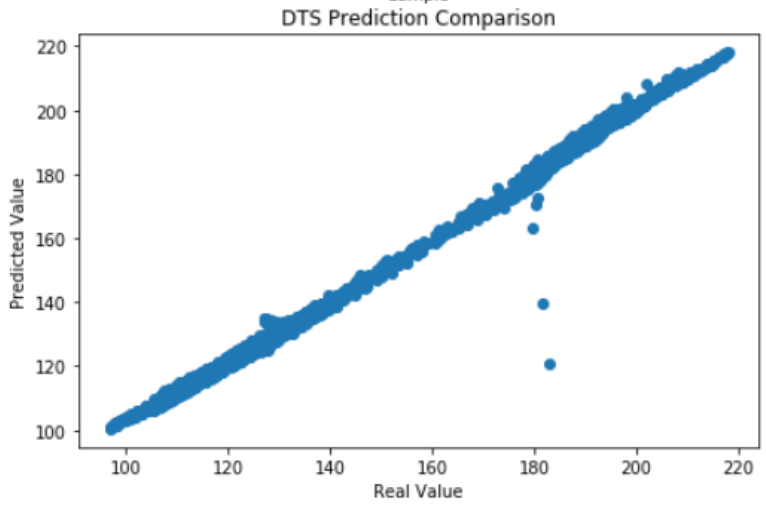
```
Model: "model_5"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_6 (InputLayer)         (None, 40, 9)             0
_________________________________________________________________
conv1d_9 (Conv1D)            (None, 40, 128)           5888
_________________________________________________________________
conv1d_10 (Conv1D)           (None, 40, 128)           82048
_________________________________________________________________
max_pooling1d_5 (MaxPooling1D) (None, 14, 128)         0
_________________________________________________________________
conv1d_11 (Conv1D)           (None, 14, 256)           164096
_________________________________________________________________
conv1d_12 (Conv1D)           (None, 14, 256)           327936
_________________________________________________________________
max_pooling1d_6 (MaxPooling1D) (None, 5, 256)          0
_________________________________________________________________
lstm_9 (LSTM)                (None, 5, 256)            525312
_________________________________________________________________
lstm_10 (LSTM)               (None, 5, 256)            525312
_________________________________________________________________
time_distributed_5 (TimeDistrib (None, 5, 128)         32896
_________________________________________________________________
dropout_5 (Dropout)          (None, 5, 128)            0
_________________________________________________________________
flatten_9 (Flatten)          (None, 640)               0
_________________________________________________________________
flatten_10 (Flatten)         (None, 640)               0
_________________________________________________________________
dense_14 (Dense)             (None, 1)                 641
_________________________________________________________________
dense_15 (Dense)             (None, 1)                 641
=================================================================
Total params: 1,664,770
Trainable params: 1,664,770
Non-trainable params: 0
_________________________________________________________________
None
Train on 16380 samples, validate on 4065 samples
```

15. CNN+LSTM Result R2 = 0.988

```
R2: 0.9882
Root Mean Square Error is: 1.76934
```







16 Summary

- Well log data science model is developed
- Different algorithm, XGBoost, LSTM and CNN+LSTM are used for prediction.
- XGBoost has the best accuracy with R2 = 0.99.
- CNN +LSTM is already impressive, with R2= 0.98

Future Plan

- In the future, more data from different wells will be tested. The developed will be tested for its accuracy against more data collected.