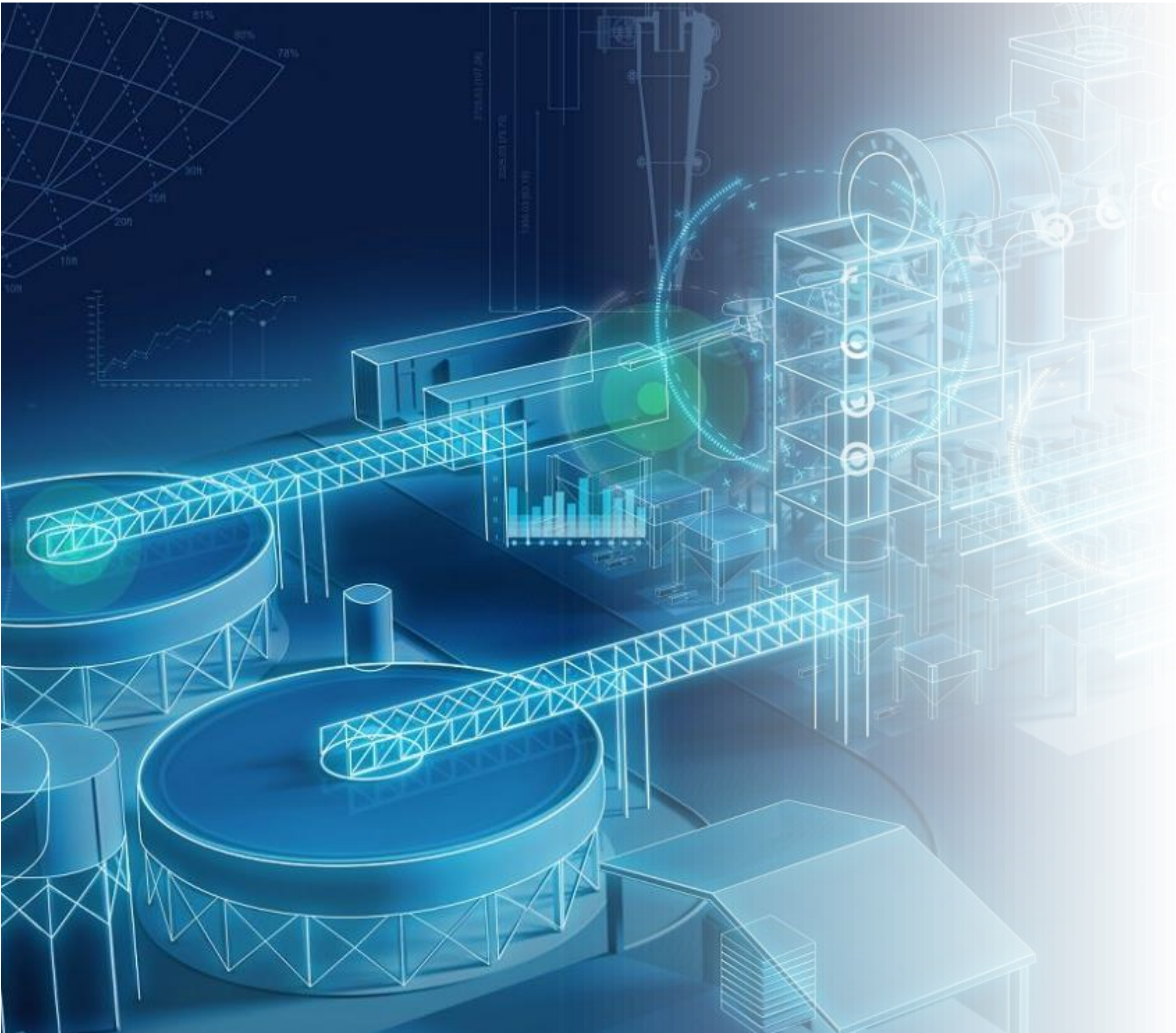


MACHINE LEARNING APPLICATION for MINERAL PROCESSING - 101

Just for the Beginning….

Aydin AKTAR

Mineral Processing Engineer

aktaraydin@gmail.com

Plants controlled by Artificial Intelligence (AI) are no far away. In this presentation, you will find the Machine learning application and concentrate predictions made with the developed model on data from a Flotation Plant.

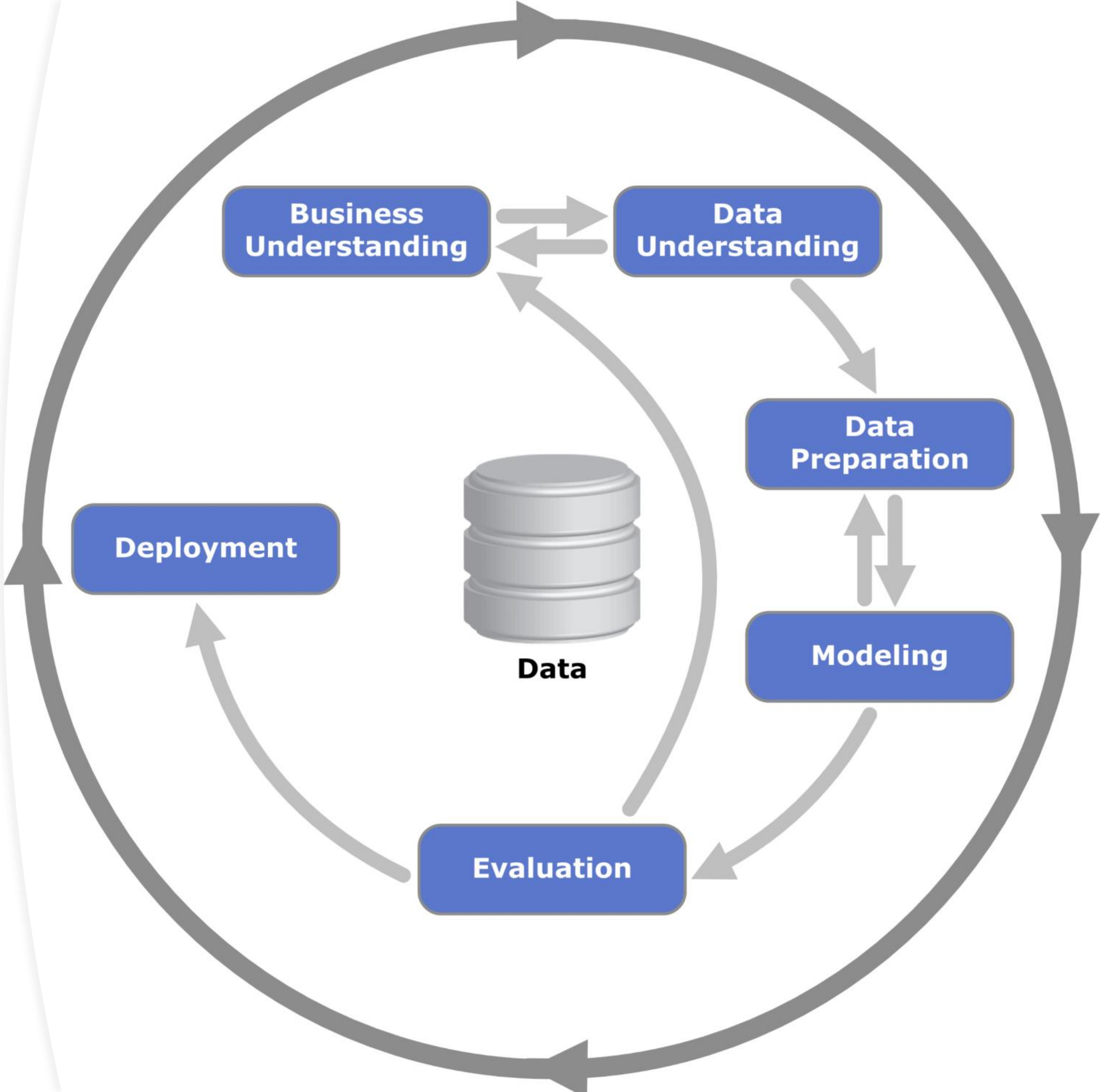
You can reach the detailed report and codes (Jupyter Notebook) links

https://github.com/aktaraydin/ML_for_MineralProcessing

CRISP-DM

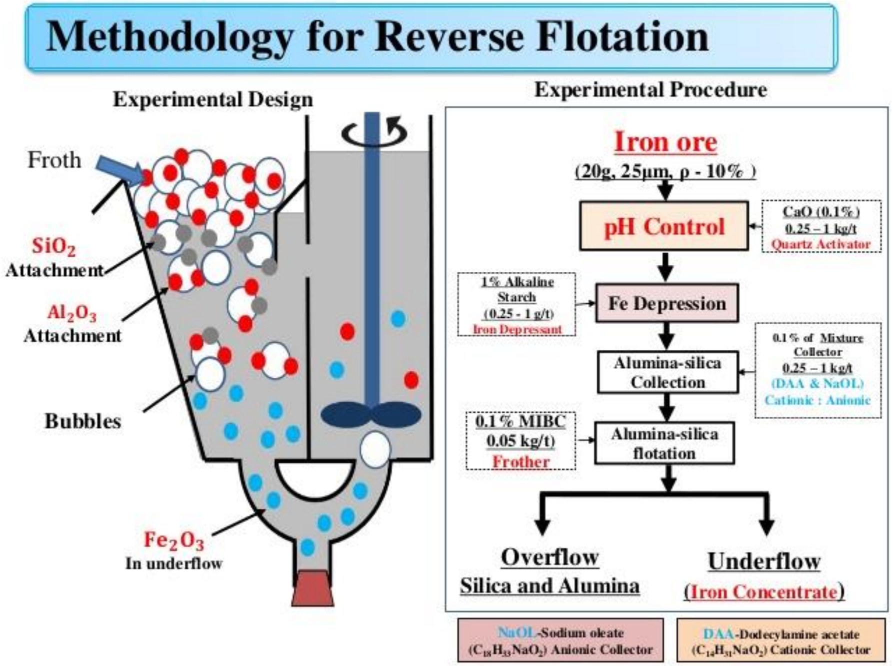
Cross-industry standard process for data mining, known as CRISP-DM, is an open standard process model that describes common approaches used by data mining experts. It is the most widely used analytics model.

I developed the project over the CRISP-DM cycle



CRISP-DM STEP-1: BUSINESS UNDERSTANDING

We cannot use the Mineral or Element in the ore extracted from the earth directly in industry. We
have to carry out an enrichment operation beforehand. The data we obtained was taken from a
Flotation Plant and Flotation method is Cationic Reverse Flotation for separation iron containing ore from silica.



CRISP-DM STEP-2: DATA UNDERSTANDING

Examine Data as Data Scientist

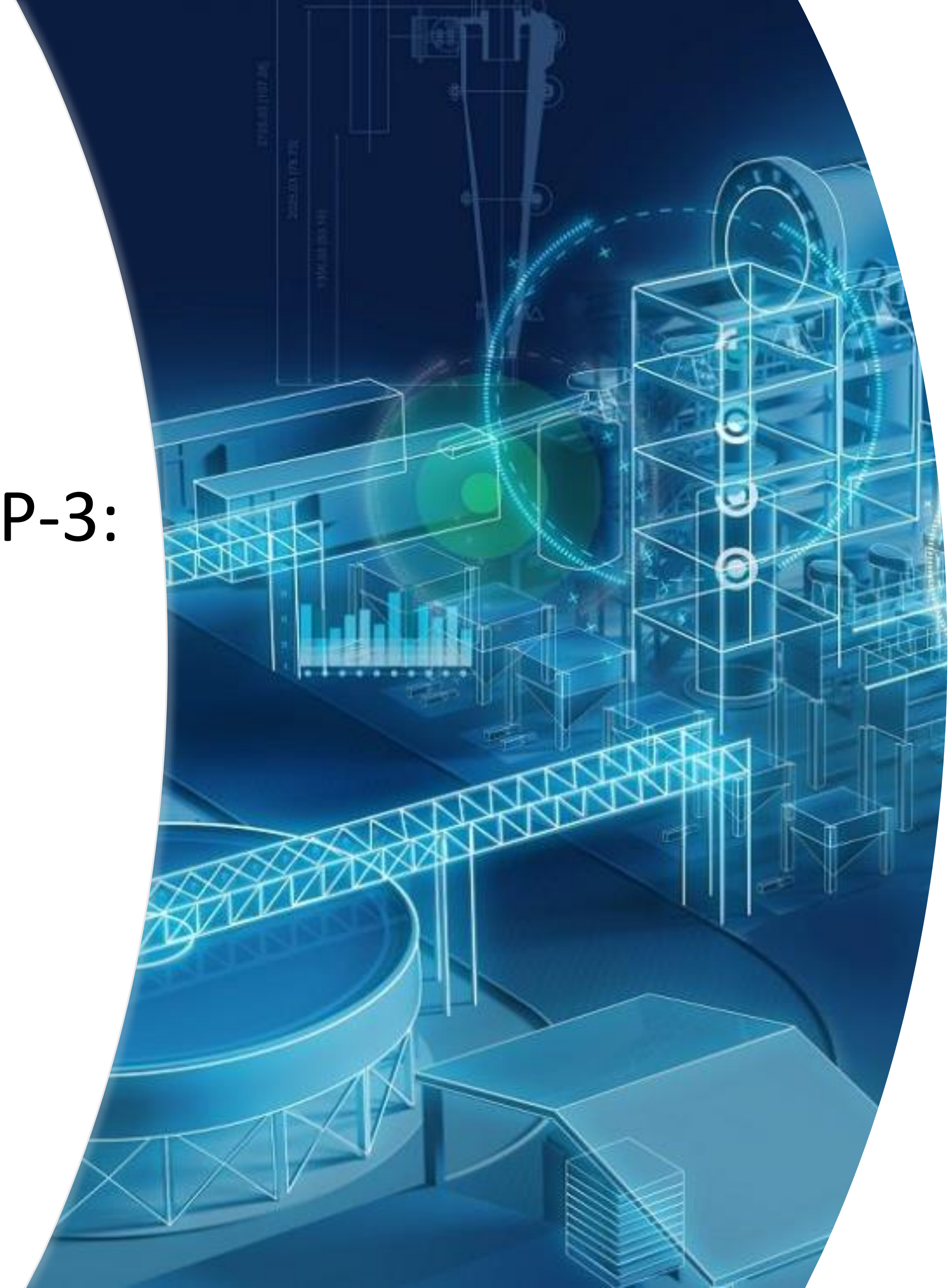
- Data consists of 24 columns and 737453 rows, covers the dates between March 2017 and September 2017 and some of the data were recorded at 10-second intervals and some at 1-hour intervals.
- There is no “null” values in data.
- Column values types are float except date.

CRISP-DM STEP-2: DATA UNDERSTANDING

Examine Data as Mineral Processing Engineer:

- There are missing data from plant for analysis
- Plant must be operating with established automation system. It is not possible collect this data with this frequency by human hand.
- Recovery , Liberation Degree, Grinding Performance, Pumps Amperes, Pressures, Flows, Feed Tons, Feed and Tail Grades for Every Columns , Mineralogy… We need this topics and I am sure we can obtain this data from this plant.
- Because of the missing data (I mentioned above) In my opinion this data useless for real life. We will continue the analysis as we have no other choice.

CRISP-DM STEP-3: DATA PREPARATION



- Rows is grouped with Hourly Frequency.
- Data is divided as “silica concentration”, “iron concentration” and “flotation conditions”

CRISP-DM STEP-4: MODELING

- P-Value calculation is done. It was observed that some data did not affect the result, but because of integrity, no columns were dropped from the data.
- Confidence Interval is calculated.

  Confidence interval of Fe Concentrate:

  [65.01585076976251, 65.0843176461516]

  Confidence interval of Silica Concentrate:

  [2.292315650512625,2.36119179395893]

- Linear Relationship: There is no Linear relationship between columns.

CRISP-DM STEP-5: EVALUATION

- Random forest regressor is preferred for this data because of best R-squared score. (0.90)
- Whole data training gave more higher R-squared result from train test split.

```
'''Iron Random Forest Model'''
regressor_Fe = RandomForestRegressor(random_state = 0, n_estimators = 100)
regressor_Fe.fit(flotation_conditions,iron_concentrate)
y_pred_Fe = regressor_Fe.predict(flotation_conditions)

print('R2 Score of Random Forest Regression with Only Iron',r2_score(iron_concentrate,y_pred_Fe))
```

```
R2 Score of Random Forest Regression with Only Iron 0.9086194504298444
```

```
'''Silica Random Forest Model'''
regressor_Si = RandomForestRegressor(random_state = 0, n_estimators = 100)
regressor_Si.fit(flotation_conditions,silica_concentrate)
y_pred_Si = regressor_Si.predict(flotation_conditions)

print('R2 Score of Random Forest Regression with Only Silica',r2_score(silica_concentrate,y_pred_Si))
```

```
R2 Score of Random Forest Regression with Only Silica 0.9090494050116655
```

CRISP-DM STEP-6: DEPLOYMENT
1. Prediction with Random Values

- **Predicted Fe Concentrate = 64.962 %**
- **Predicted Silica Concentrate = 2.461 %**

```
predictions = {'% Iron Feed':50.5,
           '% Silica Feed':13.3,
           'Starch Flow':3500.0,
           'Amina Flow':580.0,
           'Ore Pulp Flow':400.0,
           'Ore Pulp pH':10.11,
           'Ore Pulp Density':1.69,
           'Flotation Column 01 Air Flow':250.0,
           'Flotation Column 02 Air Flow':150.0,
           'Flotation Column 03 Air Flow':270.0,
           'Flotation Column 04 Air Flow':190.0,
           'Flotation Column 05 Air Flow':230.0,
           'Flotation Column 06 Air Flow':200.0,
           'Flotation Column 07 Air Flow':240.0,
           'Flotation Column 01 Level':480,
           'Flotation Column 02 Level':210.0,
           'Flotation Column 03 Level':550.0,
           'Flotation Column 04 Level':620.0,
           'Flotation Column 05 Level':610.0,
           'Flotation Column 06 Level':615.0,
           'Flotation Column 07 Level':616.0,
           }
```

```
predict_values = []
for value in predictions.values():
    predict_values.append(value)

predict_Fe = regressor_Fe.predict([predict_values])
predict_Si = regressor_Si.predict([predict_values])

print('Predicted Fe Concentrate =',predict_Fe,'%')
print('Predicted Silica Concentrate =',predict_Si,'%')
```

```
Predicted Fe Concentrate = [64.96165] %
Predicted Silica Concentrate = [2.46170555] %
```

CRISP-DM STEP-6: DEPLOYMENT
2. Prediction with Mean Values of Main Data

```
predict_from_means = flotation_conditions.describe().mean()
predict_Fe = regressor_Fe.predict([predict_from_means])
predict_Si = regressor_Si.predict([predict_from_means])

print('Predicted Fe Concentrate from Conditions mean  =',predict_Fe,'%')
print('Predicted Silica Concentrate from Conditions mean =',predict_Si,'%')
```

```
Predicted Fe Concentrate from Conditions mean  = [64.83364983] %
Predicted Silica Concentrate from Conditions mean = [2.96153289] %
```

Predicted Fe Concentrate = 64.833 %
Predicted Silica Concentrate = 2.961 %

ML Result

- We obtained the values we determined in the Confidence Intervals. The results are also very close to
  the average tenor values.

- **After all, we have a working ML application.**

CONCLUSION

Do not restrict the estimates that can be made to concentrated grades only. Questions set the entire limit in machine learning. All the fun starts in this section.

For example:

- Chemical consumption is also a problem. What is the optimum amount of chemicals I should feed by experimenting with plant conditions?

- What can I do to optimize water consumption?

- How should I optimize the plant for the target concentration and/or tail grades?

- Ore mineralogy has changed. I have laboratory data. What are my parameters for optimum processing? Do you think the tale of actual facility data will differ from laboratory results?
- Can I optimize the facility machinery-equipment maintenance tracking according to the ore?
- Can I predict plant machinery-equipment failures and optimize warehouse stocks and workforce? (Predictive Maintenance)

Whole answers are YES you can.

With the analysis made over the same data set, we can write thousands more questions and get results.

Our touchstone data on the road to IoT (internet of things) and how we store them. As long as we take care of our data and enrich them, the future is not far away. The more data you get, the more accurate results are inevitable.

Artificial intelligence, which analyses the ore coming from the mine and gives us the facility parameters as ready and calculate equipment maintenance possibilities, is now very close.

See you on the next project.

Aydin AKTAR

ps1: This presentation design was made by artificial intelligence.

ps2: Artificial intelligence has been assisted for the Turkish-English translations of this report.

Good Luck!