DOI: 10.9744/JIRAE.5.1.7-13

Implementation of Data Flow, Predictive Model, and Data Visualization in Corbion Process Analysis System

Sie Ivan Hinman Siehoyono[1,2,a], Li Li[2,b], Thanh Tran[3,c]

[1]Department of Informatics, Petra Christian University, Surabaya, Indonesia
[2]Information and Communication Technology, Fontys University of Applied Science, Eindhoven, The Netherlands
[3]Corbion Group Netherlands B.V., Gorinchem, The Netherlands
[a]ivan_h12@yahoo.com, [b]li.li@fontys.nl, [c]Thanh.Tran@corbion.com

Abstract. Corbion Group Netherlands B.V. is a company specialized in food ingredients such as lactic acid which involves a fermentation process. Thus, a system was created to monitor a real-time fermentation process in order to gain insight and control of the process. This system is called Corbion Process Analysis System, or CorPAS. CorPAS covers many projects in Corbion, and in general, it extracts data from flat files, then transforms and loads the data into Power BI for data visualization. The data extraction still needs to be manually done by the users by opening the database and copying all the data into flat files. CorPAS 2.0. was initiated to improve the previous system and to add a new feature, such as data extraction, transformation, and loading, applying data science predictive model, and data visualization. Automated data extraction was successfully implemented with data transformation and visualization, alongside with implementation of a new feature, data science predictive modeling with Random Forest and Apriori algorithm, along with data visualization. Random Forest model can reach up to 90% of accuracy. The solution was built with R language and data visualization tool such as Power BI.

Keywords: Extract, transform, and load; data visualization; data science predictive model.

1. Introduction

In this digital era, everything is built upon data and has a large impact on everyday lives. “Data is a precious thing and will last longer than the system themselves” said Tim Berners-Lee, the inventor of the World Wide Web **Error! Reference source not found.**. Data can allow an organization to measure how well the organization is doing and can also lead to better decision making. In order to have useful information, data needs to be stored and processed correctly so the data does not provide useless or false information.

Corbion has a system that stores all its fermentation process data in a database, transforms all necessary data, and visualizes the data. These data can help Corbion or the users to monitor the fermentation process and can take action if there is something wrong or not in line with the correct procedures. This system is called Corbion Process Analysis System, or CorPAS. CorPAS 1.0. had been released and used within Corbion. In CorPAS 1.0., the data extraction was still not automated, where the users need to copy the data from database and paste it into an excel file. There were several projects and user requirements that have not been implemented in the system. Thus, CorPAS 2.0. was initiated to cover additional projects and user requirements. Automated data extraction was implemented to reduce the manual labor done by users to copy-paste the data and to reduce human error in the system.

This paper describes the assigned tasks to develop CorPAS 2.0. such as implementation of new data flow, data science predictive modeling, and data visualization, alongside with newly added project for CorPAS called Waste Water Treatment System (WWTS). All of the implementations were applied into WWTS project.

Chapter 2 describes the current system and the proposed works to achieve the assigned tasks or project goals along with the technologies and tools used during the project. The implementation and results are presented in Chapter 3. Chapter 4 briefly summarizes the overall project results.

All table and variable names are obscured due to confidentiality.

2. Research

This section details the curent system and the proposed works to achieve the project goals. The goals comprise creating data extraction, transformation, and loading; implementation of data science predictive model; and data visualization.

2.1. Current Situation of CorPAS

Corbion Process Analysis System or CorPAS is a system that has been developed by Corbion for monitoring real-time fermentation process. The system collects, transforms, and visualizes the data from different data sources for project support within the company. CorPAS 1.0. had already been launched and used by some projects and staff in Corbion. Each of these projects has its own requirements, ranging from the formula/calculation implementation, data visualization, to deploying advanced features like predictive modeling.

CorPAS itself is based on R programming language and Power BI. R is used to make scripts to handle extract, transform, and load (ETL) process from raw data to data that is needed by the project or user. Power BI is used to visualize the data that has been gone through ETL process so the user can get useful information and visualization from the data. Some projects need to use CorPAS and apply a new feature like data science predictive modeling in the system.

The goal is to develop CorPAS 2.0. with additional requirements, such as implementation of a new project in Corbion called Waste Water Treatment System to make use of

CorPAS, data flow with automated data extraction, application of data science predictive modeling, and data visualization in Power BI. Most of the user requirements are about data selection, applying calculation, and data visualization.

2.2. Data Extraction, Transformation, and Loading

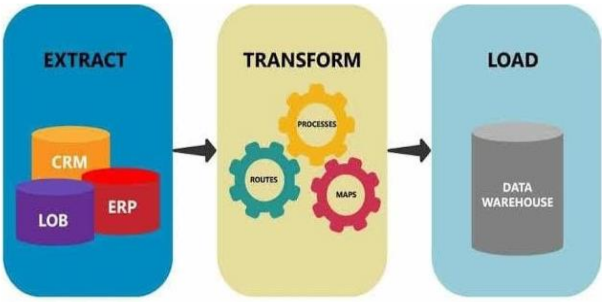


Figure 1. ETL Process

Extract

Extract is the first step of ETL process where the data is retrieved from the desired data sources along with specific rows and columns that are needed. The data sources can be varied, from relational database, excel files, and other types of data storage systems. The data extracted is moved into staging area between the data sources and data target, where data is processed in this area during ETL process [2].

Transform

After getting all the desired data, data transformation is the second step. Data that is extracted from the data source is called raw data and it is unusable for data analysis and result reporting. Cleansing, transforming, and aggregating the data occur in this step. It is crucial to ensure the best data quality is created during this step since the data comes from different kinds of data storage with various formats. Transform is the key step in ETL process where it adds value in the data. The transformed data has all the needed and useful information for critical decision making for the company.

Load

The final step in ETL process is to load data into the data target. The target may be a database, data warehouse, flat file, or business intelligence tool. Data that is loaded in target data is the final data that is used for analysis.

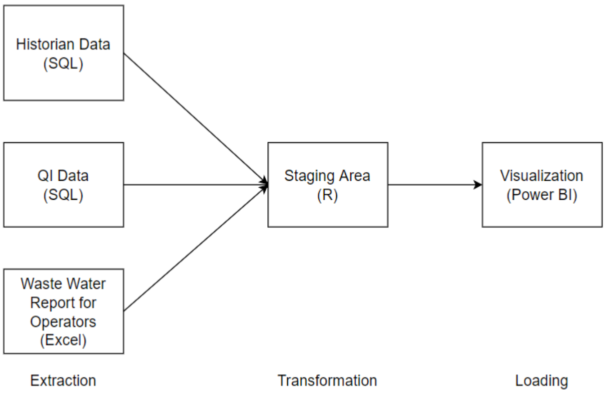


Figure 2. ETL Process for WWTS Project

Figure 2 shows the process of ETL for WWTS project. This project extracts data from three different sources. All of these data record every wastewater that is going into wastewater tank. Two data sources are obtained from SQL Servers, and the other one was retrieved from excel file that has been stored in network storage [3]. Each data source stores different types of data and has some differences in storing date/time data. In *historian database*, data is stored down to milliseconds interval period which was done automatically by sensors and has date and time values. Conversely, for *Quality Improvement (QI) database* and *Waste Water Report for Operators*, the interval period between each data is one day, where the data only has date values and were manually inputted by the users.

In transformation stage, data that has been retrieved needs to be reshaped and transformed according to the user requirements. The historian data consists of five different data that need to be joined together. Since all the historian data retrieved have the same date and time values and each value is unique, the data can be merged by date and time. Meanwhile, QI data and excel data only have date values since the data were recorded only once per day. They are merged with historian data by using the same date in order to do the correct calculation.

The next step is loading the data that has been processed and ready to be analyzed. In order to achieve the best performance, all unnecessary data from previous step is removed before data loading. The R script is run in Power BI and loads the data into Power BI. The data then is presented for the user.

2.3. Data Science Predictive Modeling

Predictive modeling is used to analyze historical data to predict the outcomes of future events [4]. A predictive model needs to be created based on the related data by applying classification/machine learning algorithm. The WWTS project aims to apply predictive modeling in order to monitor the data trends and the relationships among *parameters* or data. The predictive modeling can give WWTS project better insights and control on the quality of WWTS. Parameter is a variable that is measured or calculated within the process. It can be *quality attributes* that are associated with the performance of the process itself, or *critical parameters* that can influence the performance. In WWTS project, two machine learning algorithms are used. Random Forest is used to obtain the data trends and parameter ranking feature. Apriori Rules is used to examine the relationships among parameters.

Random Forest

Random Forest is an ensemble machine learning that can be used for classification and regression [5]. Random Forest is used in this project because of its performance. It has high accuracy and fast training time compared to other algorithms. Random Forest creates many decision trees with random features and samples from a subset of data. This way, it can increase the diversity of the data and thus makes it low in bias and variance [6]. The purpose of using Random Forest model is to get the data trends from WWTS data and to get the rank of the important parameters which contribute most for the model decision making.

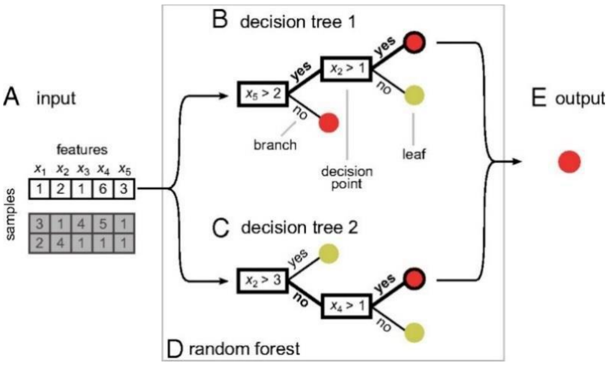


Figure 3. Random Forest Visualization

Figure 3 presents the visualization of Random Forest algorithm that consists of two decision trees when making a prediction. The output from both trees are merged together to get a more accurate prediction.

Apriori Rules

Apriori Rules algorithm is a machine learning algorithm used for data mining, where the algorithm can discover relations between variables or parameters in the data **Error! Reference source not found.**. Apriori Rules is one of the algorithms used for association rules. Association rules is a machine learning method and it aims to find "if this, then that" relations between data **Error! Reference source not found.**. It searches for frequent itemsets, which pass the minimum threshold values called *support* and *confidence*. Support tells how frequent the item appears in the data, while confidence tells the number of times the rules are found true. Apriori is used within CorPAS system to get a better understanding of the data, in this case information about relations among parameters and to get which parameter on what range can result in good results of quality attribute parameters.

2.4. Visualization in Data Visualization Tool

Power BI is used as a data visualization tool for users to get better insights into the data or data presentation. Power BI produces a visualized report or a dashboard so that any user can easily interpret and understand the information presented in the report. There are many visualizations provided by Power BI and the users can determine which types of visualization to display the data.

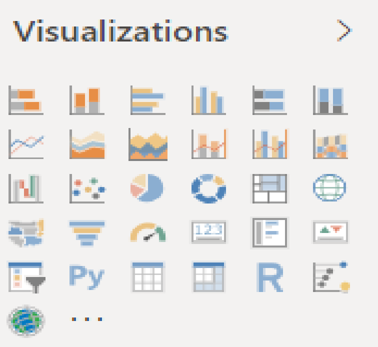


Figure 4. List of Visualizations in Power BI

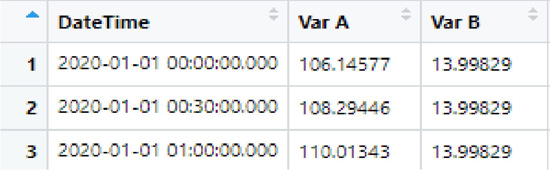
Figure 4 shows the list of visualizations provided in Power BI, from charts, table, map, Python script, to R script.

In CorPAS, the ETL process occurs with the use of Power BI by running the R script in Power BI. When the data is loaded, the user can make its visualization in Power BI. Some visualizations are defined according to the user requirements, where the visualizations are created previously by the user by making charts manually in an excel file. In Power BI, the users can create the data visualization by selecting which types of data visualization they want to see. Power BI then creates the visualization automatically based on earlier selected types of visualization and with the data selected by the user.

3. Implementation

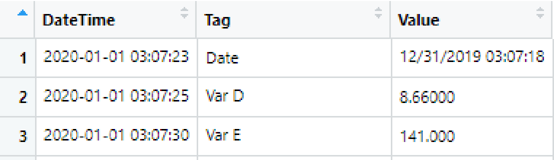
3.1. Data Extraction, Transformation, and Loading

R scripts were created to handle all the ETL processes. One main R script was created to encapsulate other R scripts (functions) to make the code cleaner and reusable. Functions were made to extract all the data from historian database, QI database, and excel files. When running the functions, the process connects to the particular database or storage, runs the query, and returns all the data based on the query to the main script.



	DateTime	Var A	Var B
1	2020-01-01 00:00:00.000	106.14577	13.99829
2	2020-01-01 00:30:00.000	108.29446	13.99829
3	2020-01-01 01:00:00.000	110.01343	13.99829

Figure 5. Historian Data

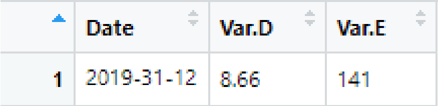


	DateTime	Tag	Value
1	2020-01-01 03:07:23	Date	12/31/2019 03:07:18
2	2020-01-01 03:07:25	Var D	8.66000
3	2020-01-01 03:07:30	Var E	141.000

Figure 6. QI Data

Figure 5 and 6 show how the historian and QI data look like after being retrieved with the R scripts from SQL database. Historian data can consist of 10 to 16 columns with a particular date and time value. As requested by the user, the data retrieved from historian database has 30 minutes interval. Historian database consists of five historian data/tables that need to be retrieved.

The next phase is to transform the data to fit the user's needs. Some functions were made to transform the data, for example, a function to change the data from long to wide format and also a function to do calculation for the data. Figure 7 shows the QI data after changing the format from long to wide format **Error! Reference source not found.**. All parameter names in column 'Tag' (Figure 6) are transformed into table header names (Figure 7), where repeated parameter names will be in a single row, and its value is in a separate column. This step was done to make the data easier to read.



	Date	Var.D	Var.E
1	2019-31-12	8.66	141

Figure 7. QI Data after Transformation

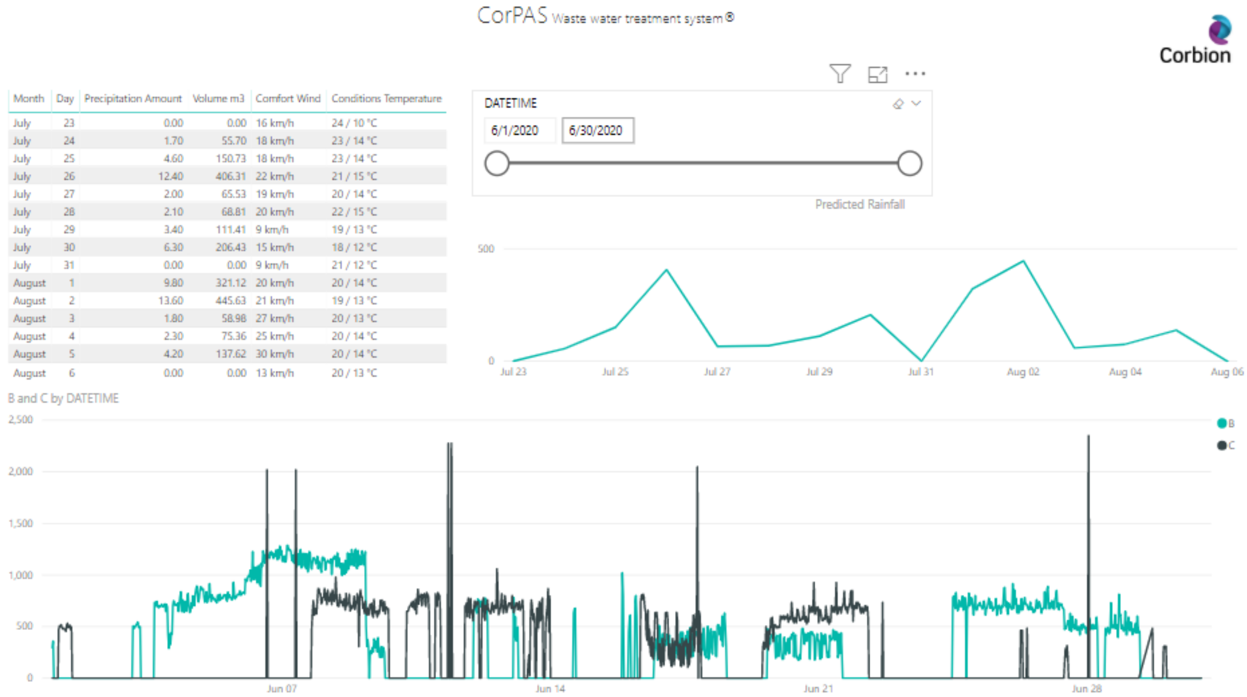
Other transformations that need to be done in this phase are to merge all the data into one considering the date and time, and calculations that were defined by the user.

The last phase is to load the data into Power BI. In Power BI, there is an option where it can get the data with R script. In this step, the main script that was made before was copied into Power BI. Then, it automatically ran the script and loaded the data into Power BI. Visualizations were made after getting the correct data and load the data into Power BI. As an initial dashboard, visualizations were made based on user requirements. Figure 8 shows the visualization for historian data.

Besides visualizing historian data (Figure 8), the data from predictive model is also visualized in Power BI. The visualization of the predicted result from random forest algorithm, along with its performance/accuracy and parameter ranking are shown in Figure 9.

3.2. Data Science Predictive Modeling

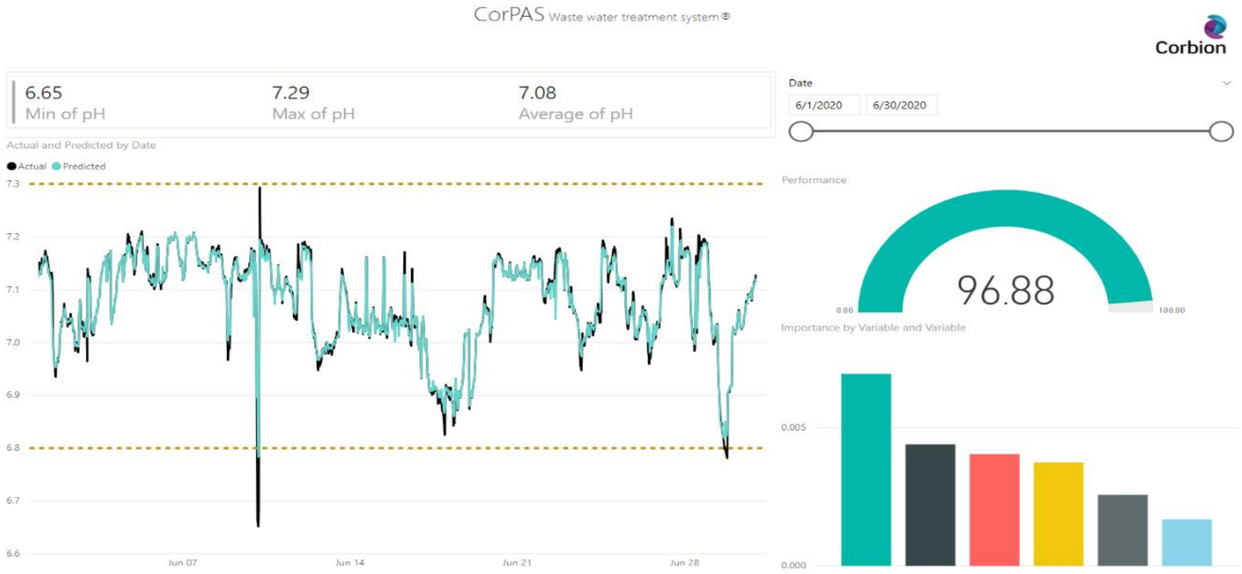
All of the data science predictive modeling was implemented in R, with the help of certain libraries such as random Forest and Arules libraries. Before applying predictive modeling, the data was cleaned by removing missing columns and rows when the data missing is more than 30%, and deleting unnecessary columns for the prediction such as date/time columns. Parameters called quality attributes were defined by the users to predict the outcome of these parameters. A quality attribute has certain range values which indicate whether those values are good or bad.



Month	Day	Precipitation Amount	Volume m3	Comfort Wind	Conditions Temperature
July	23	0.00	0.00	16 km/h	24 / 10 °C
July	24	1.70	55.70	18 km/h	23 / 14 °C
July	25	4.60	150.73	18 km/h	23 / 14 °C
July	26	12.40	406.31	22 km/h	21 / 15 °C
July	27	2.00	65.53	19 km/h	20 / 14 °C
July	28	2.10	68.81	20 km/h	22 / 15 °C
July	29	3.40	111.41	9 km/h	19 / 13 °C
July	30	6.30	206.43	15 km/h	18 / 12 °C
July	31	0.00	0.00	9 km/h	21 / 12 °C
August	1	9.80	321.12	20 km/h	20 / 14 °C
August	2	13.60	445.63	21 km/h	19 / 13 °C
August	3	1.80	58.98	27 km/h	20 / 13 °C
August	4	2.30	75.36	25 km/h	20 / 14 °C
August	5	4.20	137.62	30 km/h	20 / 14 °C
August	6	0.00	0.00	13 km/h	20 / 13 °C

DATETIME
6/1/2020
6/30/2020

Figure 8. Historian Data Visualization



6.65
Min of pH

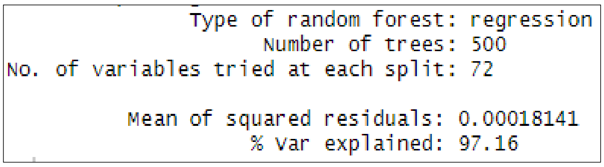
7.29
Max of pH

7.08
Average of pH

Date
6/1/2020
6/30/2020

Figure 9. Random Forest Result Visualization

Some hyperparameters can be adjusted in Random Forest. A hyperparameter is a variable that is used to control the learning process of a machine learning algorithm. In this project, the hyperparameters were set where the number of trees used was 500 and the variables used in each tree were one-third of the total parameters in the total data, which is 72.



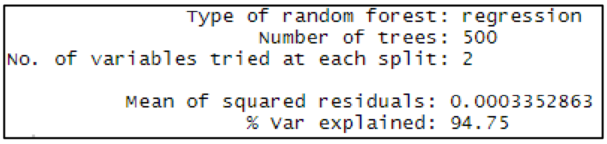
```
               Type of random forest: regression
                     Number of trees: 500
No. of variables tried at each split: 72

          Mean of squared residuals: 0.00018141
                    % Var explained: 97.16
```

Figure 10. Random Forest Result with All Parameters

Figure 10 shows the result from Random Forest model created with 500 trees and 72 variables, where it can reach 97.16% accuracy.

After getting the result of Random Forest model, the top six parameters from parameter ranking (Figure 13) from the Random Forest model were used to run the second Random Forest algorithm. Parameter ranking is one of the features that Random Forest has. The Random Forest model has its own feature where it can rank the parameter based on the influence of each parameter. It is to check whether these top six parameters truly have a high influence or not. If the accuracy does not change much, that means these parameters have high influence in the prediction. Meanwhile if the accuracy differs too much, it means these parameters do not have high influence in the prediction.



```
               Type of random forest: regression
                     Number of trees: 500
No. of variables tried at each split: 2

          Mean of squared residuals: 0.0003352863
                    % Var explained: 94.75
```

Figure 11. Random Forest Result with Top Six Parameters

Figure 11 shows the result from second Random Forest application with only top six parameters. The accuracy does not differ too much from the previous Random Forest model, which means that these parameters are important and have high impact for predicting the corresponding quality attributes.

By applying random forest algorithm, the model can predict the value of quality attribute parameter (Figure 12) and rank parameters based on their impact on the outcomes of quality attribute parameter (Figure 13).

Figure 12, shows the Random Forest prediction of quality attribute parameter. The y-axis shows the value from quality attribute parameter, where the black line indicates the actual value, while the blue line indicates the predicted value from Random Forest model. The x-axis shows the date/time value.

The top six parameters were then used, as per request from the users (where they want to focus only on six parameters), for Apriori algorithm to see which parameters and on what ranges that the results lead to good quality attribute parameters. In order to apply Apriori, all continuous values were converted to categorical value, and quality attribute parameter was defined whether it was in a good or bad range. In this project, hyperparameters tuned are thresholds called support and confidence, and the target is only set to good quality attribute that the users want to focus on.

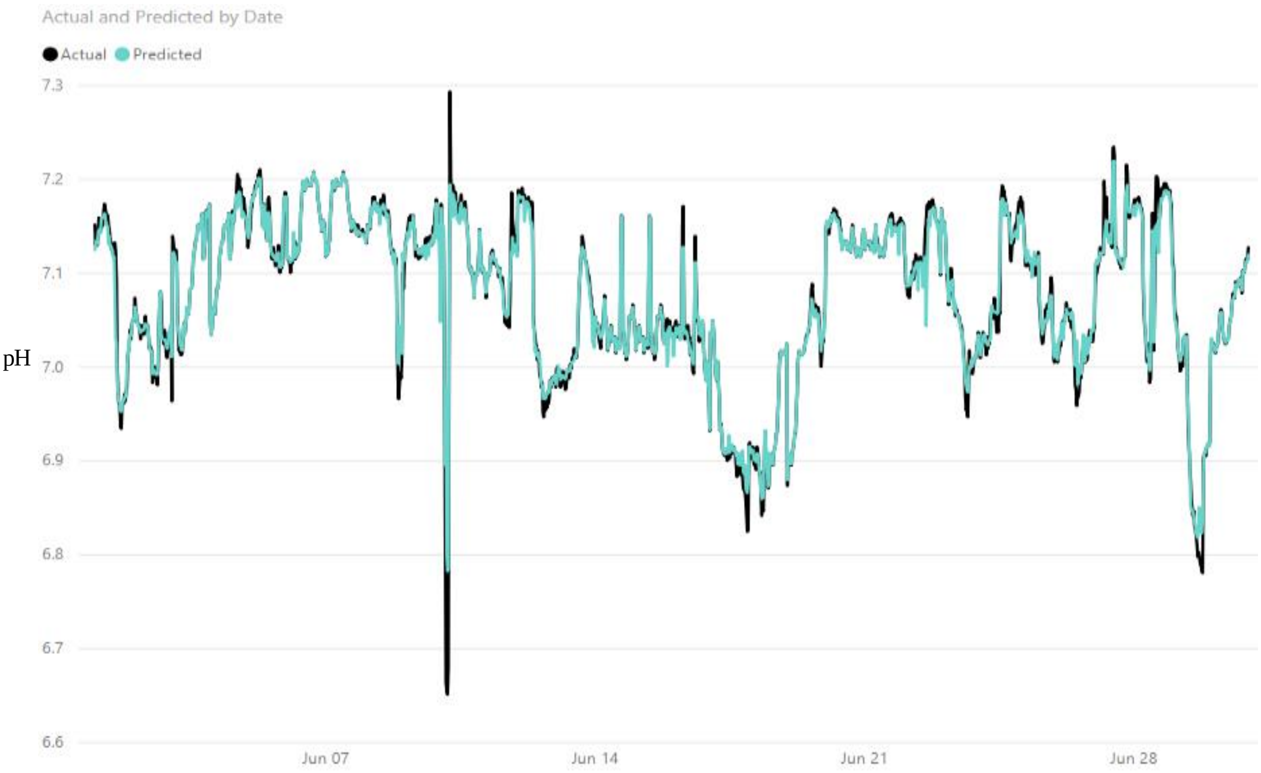


Figure 12. Random Forest Prediction

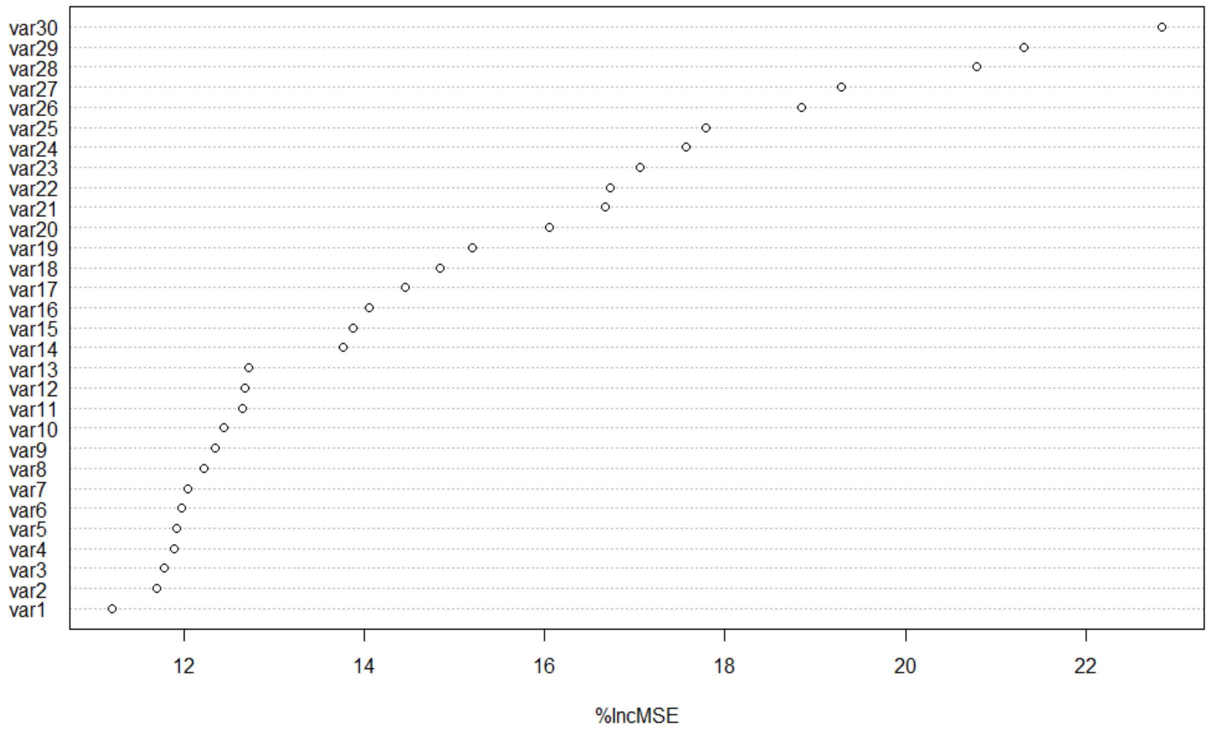


Figure 13. Random Forest Parameter Ranking

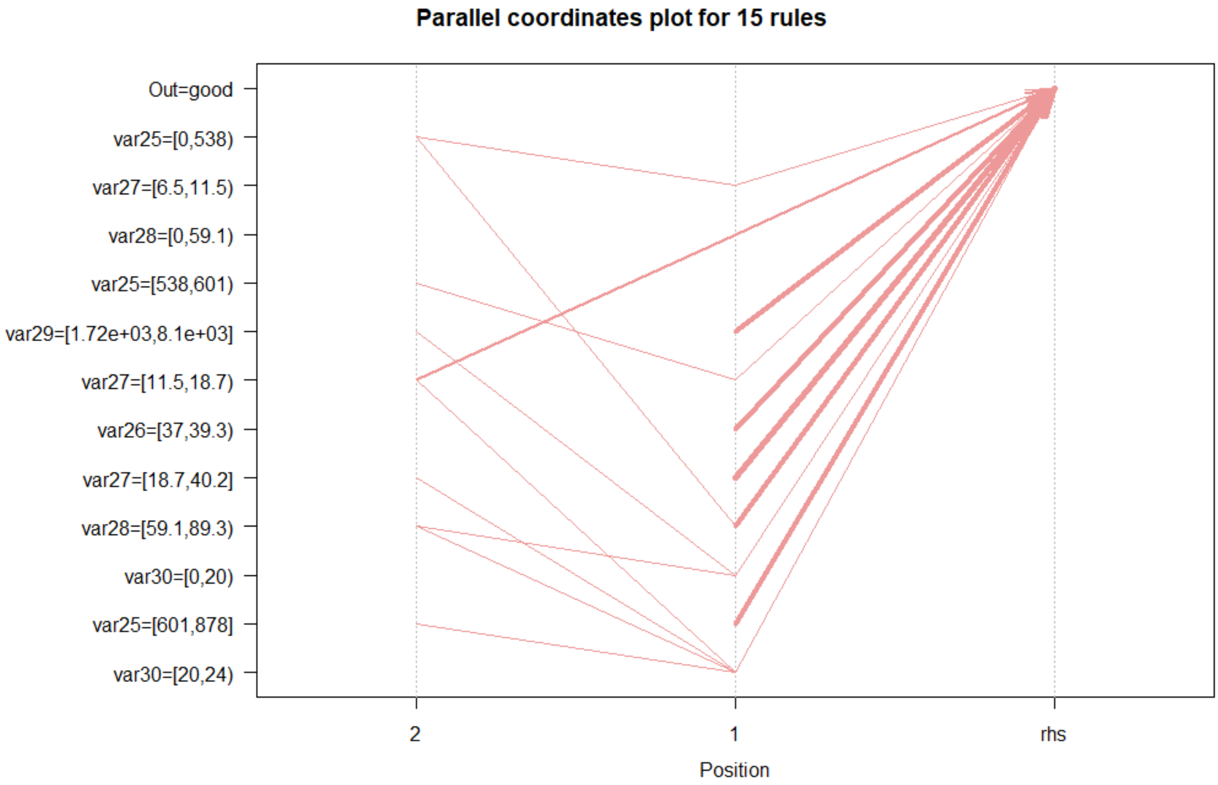


Figure 14. Apriori Algorithm Visualization

Figure 14 shows the visualization made from apriori algorithm rules. Position 1 shows the direct impact of the parameter that affects quality attribute. Meanwhile position 2 shows relations (‘if this, then that’ rule) with parameter in position 1. For example, var25 (second most bottom variable) whose value falls between 601 and 878, infrequently causes var30 (most bottom variable) value to fall between 20 and 24. Var30 whose value falls between 20 and 24 also infrequently results in good quality attribute. The line/cord thickness shows the frequency of the parameters appearing in the data. The thicker the line is, the more frequent that parameter appears. For example in position 1, var25 whose value falls between 601 and 878 appears frequently in the data. This means, var25 is most likely one of the variables that can affect the outcome of good quality attribute.

In the project, Random Forest model was built to make use of the parameter ranking feature. The accuracy of the model defines how well the model can predict in comparison with the actual data. In this project, the accuracy achieved was 97.16% with all the parameters, and 94.75% with top six parameters. Next, the top six parameters were used to look into the relations between each parameter with help of Apriori algorithm, and then plotted in parallel coordinate plot (Figure 14). By doing so, the user knows which parameter on what range can result in good quality attribute. For example, based on Figure 14, parameter var25 from range 601 to 878 can produce good output of quality attribute parameter.

3.3. Data Visualization

CorPAS makes use of Power BI to make visualizations as requested by the users. Figure 8 and 9 show the dashboard of WWTS project in CorPAS, as requested by the users. A new requirement was implemented in this project, namely weather forecast. Figure 8 shows the historian data visualization, alongside with weather forecast. The weather forecast data was retrieved by web scraping. Web scraping is one of the features that Power BI provided where it can automatically detect table and data from a website. By providing the URL in Power BI, it automatically detects and retrieves the data.

Other visualizations were based on the predictive model results. For Random Forest model, a line chart was made to compare the result of quality attribute prediction with the actual value, alongside with a gauge chart to see the performance of the Random Forest model and bar graph to visualize the top six parameters (Figure 9). For Apriori algorithm, a parallel coordinate plot was created with the help of R script since Power BI does not provide this type of visualization by default (Figure 14).

4. Conclusion

Corbion has an internal system that can monitor fermentation process called CorPAS. There were three goals defined. The first goal was to create a data flow for the ETL process, where the data extraction is automated to deliver the selective data according to the user requirements. The second objective was to implement a data science predictive modeling. The final milestone was to visualize the data in Power BI.

At the end, all of the goals were achieved and already implemented in the system. All tasks accomplished in this project can be used as a *base* in the system for further development for another project within Corbion. CorPAS 2.0. can already extract data automatically from the desired data sources, transform the data, and load it into Power BI for data visualization. Data science predictive modeling is also successfully implemented in the system and can create a quite accurate model (up to 90%) based on Figure 10 and Figure 11.

References

[1] Berners-Lee, T., Hall, W., & Hendler, A. J. (2006). *A Framework for Web Science.* Now Publishers Inc.

[2] Oracle. (n.d.). *Oracle9i Data Warehousing Guide*. Retrieved June 16, 2020, from https://docs.oracle.com/cd/B10501_01/server.920/a96520/concept.htm#50822.

[3] *Network-attached storage*. (2020, June 15). Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Network-attached_storage.

[4] MicroStrategy Incorporated. (n.d.). *Predictive Modeling: The Only Guide You'll Need*. Retrieved June 17, 2020, from https://www.microstrategy.com/us/resources/introductory-guides/predictive-modeling-the-only-guide-you-need#maintenance.

[5] Polikar, R. (2009). *Ensemble learning*. doi:10.4249/scholarpedia.2776.

[6] Vaishnavi, S. (2017, May). *Ensemble Methods and Random Forests.* Retrieved from https://courses.engr.illinois.edu/ece543/sp2017/projects/Vaishnavi%20Subramanian.pdf.