UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

RAFAEL SCHENA

A methodology for synthetic generation of failure data for data-driven prognostics and health management modeling for digital twins

Thesis presented in partial fulfillment of the requirements for the degree of Master of Computer Science

Advisor: Prof^a. Dr^a. Mara Abel

Porto Alegre
April 2023

CIP — CATALOGING-IN-PUBLICATION

Schena, Rafael

A methodology for synthetic generation of failure data for data-driven prognostics and health management modeling for digital twins / Rafael Schena. – Porto Alegre: PPGC da UFRGS, 2023.

108 f.: il.

Thesis (Master) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR–RS, 2023. Advisor: Mara Abel.

1. Synthetic Data. 2. Digital Twins. 3. Prognostics and Health Management (PHM). 4. Industry 4.0. I. Abel, Mara. II. Título.

“Progress is made by trial and failure; the failures are generally
a hundred times more numerous than the successes,
yet they are usually left unchronicled.”

— Sir William Ramsay

ACKNOWLEDGEMENTS

This work was developed in the Petwin Project (PeTWIN.org) scope, supported by the Brazilian Federal Agency for Innovation (FINEP) and the Libra consortium for petroleum production. It is also partially financed by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

A special thanks to Prof. Mara Abel, my supervisor, and Prof. João Cesar Netto, my co-supervisor, for the opportunity offered, for believing in me and having the patience to teach and provide the necessary support throughout this journey, so that I could complete this work. Thanks also to Prof. Joel Luís Carbonera, who greatly contributed to make this work more robust and complete. I hope I may have done a good job and lived up to expectations, contributing to the Petwin project.

I also thank my project colleagues, professors, and everyone from INF-UFRGS, from whom I learned a lot during my time at this University.

Thanks to my family. To my parents and brothers for always being sources of encouragement at all times. And, of course, to the most important person, my wife Rita, who was patient and understanding and offered the necessary encouragement so that I would never think about giving up even in the most difficult moments of her life.

And, of course, I thank God because nothing leads me to believe that this large number of special people who have helped me along this path is simply the work of chance.

ABSTRACT

Prognostics and Health Management (PHM) is one of the main services encompassed by Industry 4.0. However, the scarcity of failure data due to the nature of machines' operation is still a challenge to be transposed in this field. Due to recent advances in computing power, simulation, sensing, and networking technologies digital twins allow us to adopt a different approach to this problem: inserting failures into a digital replica of the real asset to train data-driven PHM models. In this work, we propose a general methodology to generate and validate synthetic failure data for PHM purposes. Also, we present an application of the proposed methodology, which produced a synthetic failure dataset validated with real data. In the experiment, we have modeled a smart petroleum well in a commercial computational fluid-dynamics simulator and injected failures into the system by modifying the expected behavior of the equipment to generate synthetic failure data. Then, we assessed the quality of the synthetic data by training machine learning algorithms on them, testing on data from a petroleum plant production, and applying fidelity metrics to verify the necessary improvements to the process. The results show the feasibility of generating useful synthetic data for PHM purposes, and the proposed methodology indicates points of enhancement in the generated data. The presented methodology still has limitations concerning its extrapolation for the general PHM case, and this work also discuss alternatives to overcome these constraints.

Keywords: Synthetic Data. Digital Twins. Prognostics and Health Management (PHM). Industry 4.0.

Metodologia para geração sintética de dados de falha para modelos de *Prognostics and Health Management* orientados a dados em gêmeos digitais

RESUMO

PHM (acrônimo na língua inglesa para *Prognostics and Health Management*) é um dos principais serviços englobados pela Indústria 4.0. Entretanto, a escassez de dados de falhas devido à natureza de operação das máquinas ainda é um desafio a ser transposto neste campo. Devido aos recentes avanços tecnológicos em poder computacional, simulação, detecção e rede, os gêmeos digitais nos permitem adotar uma abordagem diferente para esse problema: inserir falhas em uma réplica digital do ativo real para treinar modelos de PHM orientados a dados. Neste trabalho, propomos uma metodologia geral para gerar e validar dados sintéticos de falha para PHM. Além disso, apresentamos uma aplicação da metodologia proposta, produzindo um conjunto de dados sintéticos de falha validado com dados reais. No experimento, modelamos um poço de petróleo inteligente em um simulador de fluido-dinâmica computacional comercial e injetamos falhas no sistema, modificando o comportamento esperado do equipamento para gerar dados sintéticos de falha. Em seguida, avaliamos a qualidade dos dados sintéticos treinando algoritmos de aprendizado de máquina sobre eles, testando com dados reais de um poço de petróleo e aplicando métricas de fidelidade para verificar as melhorias necessárias no processo. Os resultados mostram a viabilidade de geração de dados sintéticos úteis para fins de PHM, e a metodologia proposta indica pontos de aprimoramento nos dados gerados. A metodologia apresentada ainda possui limitações quanto à sua extrapolação para o caso geral de PHM, e este trabalho também discute alternativas para superar essas restrições.

Palavras-chave: Dados Sintéticos. Gêmeos Digitais. PHM. Indústria 4.0.

LIST OF ABBREVIATIONS AND ACRONYMS

PHM	Prognostics and Health Management
DT	Digital Twin
ICV	Inflow control valve
CFD	Computational fluid-dynamics
PCD	Pairwise correlation difference
RUL	Remaining Useful Life
O&G	Oil and Gas
P/T	Pressure and temperature
MTTF	Mean Time to Failure
FMEA	Failure Mode and Effects Analysis
P&ID	Piping and instrumentation diagram
1-D	One-dimensional
3-D	Three-dimensional
KNN	K-nearest-neighbors
SVC	Support Vector Classifier
XGBoostRF	Extreme Gradient Boosting Random Forest

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

CONTENTS

1 INTRODUCTION

Prognostics and Health Management (PHM) is an engineering discipline whose main objective is to provide an integrated view of the health state of a machine or an overall system. The foundation for PHM has its roots in maintenance engineering concepts such as preventative maintenance, reliability-centered maintenance, and condition-based maintenance (LEE et al., 2014). To achieve such an objective, PHM uses sensors to monitor the system of interest and then apply different algorithms to assess the asset's condition. In this way, PHM systems support technical decisions to improve the asset's profitability by predicting and preventing possibly costly or even catastrophic failures.

In the context of Industry 4.0, PHM is one of the primary services to enhance the reliability and productivity of industrial assets, allowing for detection, diagnosis, assessment, and prediction (JIA et al., 2018), improving reliability and reducing the downtime of the asset. PHM is based on concepts and techniques used in predictive maintenance[1], which has been already used to support industrial asset management decisions for decades. The evolution of predictive maintenance was enabled by the advances in computing, sensoring, and communication technologies, which have transformed strategies and techniques used to maintain industrial assets, from the first industrial offline applications at the end of the last century, evolving to the concept of e-maintenance in the early 2000s, which already integrates information and communication technologies to enable proactive maintenance decisions (MULLER; MARQUEZ; IUNG, 2008), until finally reach the current stage with the industrial internet of things (IIoT) and PHM.

A modern PHM data-driven approach may benefit from machine learning modeling (LUO et al., 2020), enabling the modeling of more complex systems using not only theoretical models and equipment data but complex process data as well. A data-driven approach for such a problem has the advantage of using the pattern-recognition power of machine learning, to detect complex non-linear failure patterns in the process data. Other advantage is the real-time response of a trained machine learning model for detecting or predicting failures, instead of waiting a new simulation to be done for every condition change, which sometimes may take hours and even days. However, the problem is that this approach requires a large amount of failure data that are not usually available. The rarity of failure data is a major challenge in the PHM area nowadays (MAUTHE; HAGMEYER; ZEILER, 2021). That is an inherent feature of machinery health data, mainly

[1] According to (MOBLEY, 2002), predictive maintenance is a condition-driven preventive maintenance.

because machines generate normal-state data most of the time.

Regarding that problem, the digital twin concept, implemented inside the Industry 4.0 context, may offer a feasible solution. Intuitively, a digital twin is a system composed of a physical asset and its digital replica (virtual entity) that can interact with each other. (TAO; ZHANG, 2017) states that a complete digital twin should encompass five dimensions: a physical part, a virtual part, connection, data, and service. For PHM purposes, a digital twin should use multiple sources of data: real-time sensors, maintenance history, maintenance plans, failure analysis, machine manufacturer's manuals, and simulation models. This way, digital twins enable the creation of multiple what- if scenarios in its virtual world to support the operational optimization of the real-world asset.

Based on this principle of digital twins, this work proposes a solution for the lack of equipment failure data, whose main contributions are:

- A general methodology for generating synthetic failure data;
- An experiment description applying the proposed methodology;
- A synthetic failure dataset produced in the experiment described that was validated using real data.

To achieve those results, we used process data from a smart petroleum well (including sensor measurements, plant diagrams, and operational annotations) and an one-dimensional (1D) computational fluid-dynamics (CFD) simulator software with an API to automate the simulation process to generate failure data. Later on, we validated the synthetic data using application fidelity metrics to verify how well machine learning models trained on synthetic data would perform when tested on real unseen test data. Finally, we used the pairwise correlation difference (PCD) to evaluate how well the synthetic data fit to the real test data.

The rest of this work is organized as follows: section 2 introduces the main theoretical concepts used in this work, section 3 presents the related works, section 4 details the methodology used in the experiment, section 5 shows the application of the proposed methodology and its main results, followed by a discussion in section 6 and, finally, conclusions and future work in section 7. Appendix A is an expanded abstract in Portuguese language; appendix B presents an exploratory data analysis of the boundary conditions used to run the simulations; appendix C contains the simulation parameters; appendix D compares the produced synthetic dataset to the corresponding real data; and appendix E exposes details about the machine learning modeling process and validation.

2 THEORETICAL FOUNDATION

Before discussing the experiment and its results, we discuss the concepts of a digital twin, PHM, and how data-driven models can be useful to predict failures, the problem caused by the lack of failure data, and how a digital twin may be a feasible solution. It is also important to understand the basics of how a petroleum smart well and its main components work and how the failures occur.

2.1 Digital Twins

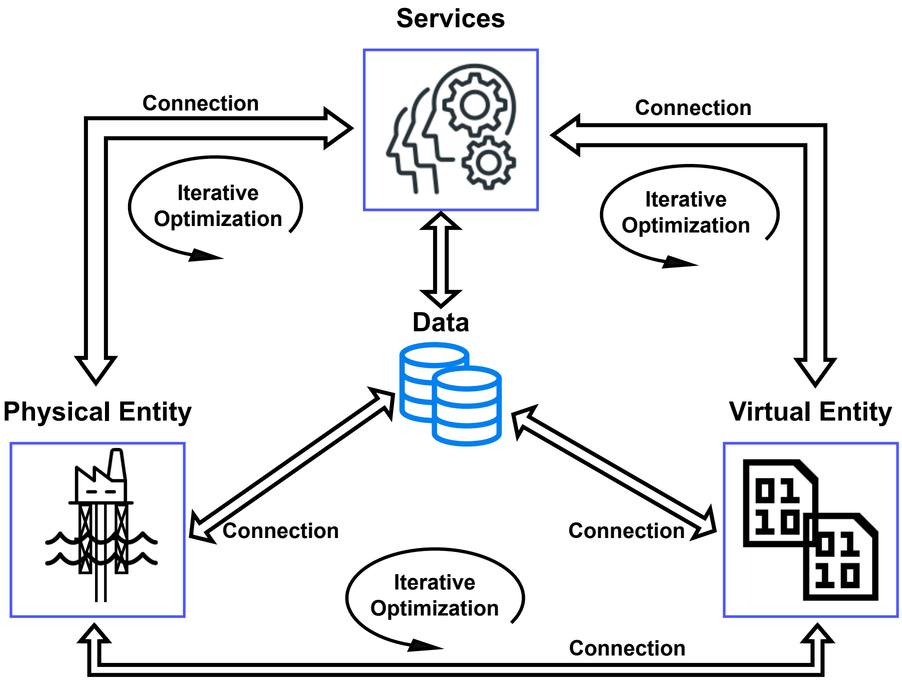
The term *Digital Twin* (DT) was first introduced in the aeroespacial industry, attributed to John Vickers of the National Aeronautics and Space Administration (NASA). Later, in the early 2000s, its application was extended to Product Lifecycle Management (PLM). However, by that time, the concept was considered immature, with the information about physical products being limited, manually collected, and mostly paper-based (GRIEVES, 2014).

Beyond the intuitive idea of digital twins presented in section 1, here we present a more formal definition. Min and colleagues (MIN et al., 2019) define a digital twin as a realization of a cyber-physical system. The work of (WANASINGHE et al., 2020) refers to this concept as a cyber-physical interaction and simulation. Nevertheless, until now, there has been no consensus about a digital twin and its essential parts and characteristics. While some authors understand that a DT is only a simulation of reality, others state that a DT is a complex entity comprised of at least three dimensions: a physical entity, a virtual entity, which is a virtual representation of the physical entity, and the connections between them.

Tao and Zhang (TAO; ZHANG, 2017) proposed a DT framework of a complete digital twin composed of five dimensions: the physical entity, the digital entity, the connections between them, the data, and the provided services. Figure 2.1 shows the Tao architecture of a DT. In the industry context, PHM is one of the services provided by a digital twin, as well as the simulations or production optimizations.

It is important to notice that a digital twin is not a static model. Namely, it is not just a simulation of scenarios based on history and real-time data collected from sensors. It is a live model that allows the virtual entity to act in the physical entity for process optimization and vice-versa. This continuous connection between physical and virtual spaces differs between a digital twin and a traditional simulation model that develops offline analysis. The continuous physical-virtual connection enables a cycle of state-change monitoring that captures not only the possible changes in the physical environment but also the changes caused by interventions done by the virtual entity itself, making the digital twin able to assess the effects of its own actions (JONES et al., 2020).

Figure 2.1 – Five-dimension digital twin model, adapted from (TAO et al., 2018).



A digital twin has, therefore, a modeling process that feeds back itself. Such a process, called the *twinning process* in the literature, is the capturing of state changes by metrology techniques, transferring it to the virtual or physical environment, and the state realization in the virtual or physical entity by a parameter synchronization. The physical and virtual environments have means for metering and realizing the state changes. The *twinning rate* is the frequency of the state synchronization between physical and digital entities (JONES et al., 2020). Therefore, the detection and learning of previous events in the *twinning process*, which is the focus of this work, is an essential step in the construction of a digital twin itself.

2.2 Prognostics and Health Management (PHM)

According to (ZIO, 2022), Prognostics and Health Management (PHM) is a computation-based paradigm that enables detecting equipment and process anomalies of machines and systems, diagnosing its degradation states and faults, predicting the evolution of degradation to failure. This is done by means of physical knowledge, information and data of the operation and maintenance of the assets.

PHM systems may have four distinct functions (JIA et al., 2018):

- Detection: identification *if* a failure has occurred without knowledge about its root cause;
- Diagnosis: determining the *root causes* of a detected failure;
- Assessment: evaluating the *risk of failure* of a system based on its recent behaviors;
- Prognosis: prediction of the future health states and *when* a failure will happen.

A frequently used indicator for prognosis purposes in PHM is the Remaining Useful Life (RUL). The work of (SI et al., 2011) defined RUL as the lasting from the current time to the end of the useful life, with the definition of useful life depending on the context. In a more precise definition, (BANJEVIC, 2009) defines RUL as the remaining time until the next failure occurs. Other indicators are useful, like a customized health index, or simply a boolean faulty/not-faulty status in the case of detecting anomalous conditions. However, quantifying the time left until the machine stops working properly offers the advantage of planning.

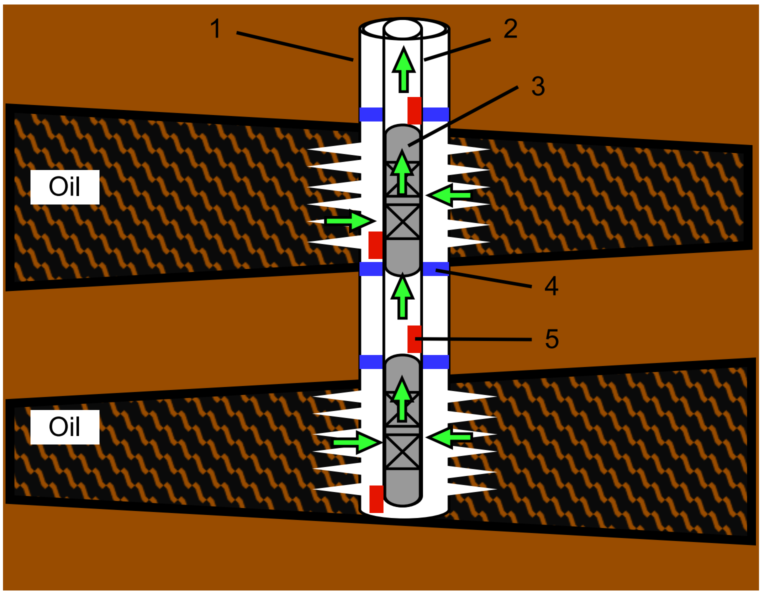
In recent years, many initiatives have proposed digital twins or cyber-physical systems for PHM both in industry and academia. The work of (JONES et al., 2020) highlights PHM as a well-established research area that predates and underpins concepts for digital twins. In addition, according to (TAO et al., 2018), most DT applications are related to PHM, and DT-driven PHM has many advantages compared to the traditional PHM, overcoming its shortcomings of relying mainly on empirical data, leading to more accurate and timely predictions.

2.3 Smart wells

In the O&G industry, a smart well is a well with downhole instrumentation (sensors and valves for inflow control) that allows the tuning of production by the continuous monitoring of fluid flow rates and pressures, and that supports adjustments of the valve configuration (YETEN et al., 2004). Such equipment assures flexibility and predictability of the well production since it makes it possible to blend fluid stream sources from different reservoirs with different chemical and physical properties, such as the oil quality and proportions of oil, gas, water, and sediments.

Figure 2.2 shows a simple scheme of the basic elements of a smart well. The figure also shows two zones of reservoir rocks containing oil and water. The key elements for understanding the operation of a smart well are the inflow control valves (ICVs), represented by the number 3 in figure 2.2. The ICVs may free or block the flow from the reservoir rock to the tubing (number 2 in figure 2.2) that carries the produced fluids to the surface. The isolation packers (number 4 in figure 2.2) seal the space between the casing (number 1 in figure 2.2) and the tubing, isolating segments and hence the fluid to enter the tubing through only one ICV, avoiding the mix of fluids from the different zones without the control of the operators. P/T gauges (number 5 in figure 2.2) measure temperature and pressure inside and outside the tubing, allowing the control system to calculate the flow from each ICV.

Figure 2.2 – Basic elements of a smart well. 1 - Casing, 2 - Tubing, 3 - Inflow Control Valves (ICV), 4 - Isolation packer and 5 - P/T gauges.



The work of (SHAW, 2011) presents the evolution of ICV technologies, including a comparison between generations, reliability, and cost issues of all-electric, electric-hydraulic, and all-hydraulic ICVs. ICVs are complex machines with instrumentation and moving parts that must be able to work whenever required, generally operating in severe conditions (high pressures and temperatures, which cause a higher likelihood of electronic components failures). Therefore, considering the ICV as the system's weakest point in terms of reliability is not unreasonable.

Nevertheless, being the less reliable point in the system does not necessarily mean absolute low reliability. In (LANGLI et al., 2001), experts estimated the mean time to failure (MTTF) of first-generation petroleum production ICVs to be two years, as no data were found by authors thus far. Later, (JOUBRAN, 2018) calculated the equipment MTTF as 148 years, with data from 1,120 first-generation ICVs. The same study presented the reliability improvement achieved in the latest-generation ICVs, with an MTTF of 425 years (based on data from 734 units over three years). The analysis of the universe size of 734 ICVs needed in that work required to observe a total of 5 failures during three years indicates the sparse failure data of this type of equipment.

Despite being rare, the consequences of an ICV failure are costly, requiring a restoration process without guaranteed results, and may lead to real production losses or, in some cases, with the valves declared as no longer useful (AL-HAJRI et al., 2021). Depending on the kind of failure or the valve configuration, the valve may stick in its current setting due to loss of control from a surface or mechanical stuck, or else automatically open or close (YETEN et al., 2004). Because of that, spurious actions are not impossible. Furthermore, the unpredictability of the need to operate the equipment (LANGLI et al., 2001) increases the difficulty of predicting when a failure will take place.

However, such unpredictability does not mean the causes of failures in that equipment are unknown. The work of (YETEN et al., 2004) presents a model of ICV failures probability density function as the *bathtub curve*[1], where, at the early stage of equipment life, the high failure rates refer predominantly to inappropriate installation, such as cables and badly assembled mateable connectors. In the medium term, the failure probability decreases until it reaches a stable plateau, when the failures refer to connections, tubing hangers, packers, and gauges, among others. In a late stage, the probability of failures increases, with one of its main causes being short circuits at gauges or connections. Rahman and colleagues (RAHMAN; ALLEN; BHAT, 2012) describe a complete failure mode and effect analysis (FMEA) of the second generation ICVs, where the main causes of failures are debris produced during the well completion[2] phase, erosion, corrosion, and wear.

[1] The bathtub curve is a typical pattern of the failure probability densities of most equipment, representing high failure rates in early life, followed by a stability period before the rates rise rapidly in the wear-out stage. Such patterns are explained in (NARESKY, 1970).

[2] Well completion is defined as a single operation involving the installation of production casing and equipment to bring the well into production, including the perforation of the casing (IANNUZZI, 2011),

In order to avoid or mitigate the impact of ICV failures, there are some good practices to be adopted in the completion and operational phases. The first one to be done before well completion is the removal of debris using junk baskets and magnets, eliminating one of the main causes of failures according to (AL-RABEH; AL-NOAIMI; BROWN, 2018). Another recommendation is to actuate the valves periodically since ICVs malfunction if not actuated for a long time (AL-HAJRI et al., 2021). Last, the good practices recommend opening all ICVs before closing any of them, eliminating the possibility of all valves stuck at a closed position (AL-RABEH; AL-NOAIMI; BROWN, 2018), which would stop production completely.

which may cause some debris (AL-RABEH; AL-NOAIMI; BROWN, 2018).

3 RELATED WORK

The scarcity of data is not a new problem when it comes to PHM. In fact, as we mentioned before, it is indeed an inherent feature of the machinery operation, since the vast majority of data is produced when machines operate in a normal condition.

However, if our goal is to produce reliable failure data, reproducing real failure patterns, we need either some amount of data or domain knowledge. In (ENO; THOMPSON, 2008) the authors propose an approach that inverts the mapping generated by the application of data mining techniques to an original dataset. This approach is suitable for cases in which some failure data is available. The application of such a strategy might even detect some hidden patterns in data, not perceived previously by domain experts. However, having data that reflects all the possible data states caused by equipment failures is not a likely scenario.

Thus, the expertise and knowledge gathered by specialists are essential to generate appropriate data. Despite not being specifically related to machine failures, (VARGAS et al., 2019) create a public dataset of undesirable events in oil and gas (O&G) production context, with data from real measurements, but also from simulations and graphs designed by specialists.

In (LEE; JO; HWANG, 2017), the authors use generative adversarial networks (GAN) to generate augmented failure data from failure patterns described in literature, inserted over normal condition data collected in bench tests. The disadvantage of this approach may be the computational cost to generate realistic data and the specificity of conditions: it covers only machines with degraded failures, failures that are gradual or partial[1] according to some output measurement, and, therefore, whose decay can be measured over time, or even fully monitored by online sensors (which is still not a likely scenario for downhole equipment).

The work of (RAO, 2020) describes a digital twin based on the Simulink model of a triplex pump[2] with parameters calibrated with real measured data. Then, the authors applied the machine operation knowledge and its failure modes[3] to simulate those failures based on how the simulation parameters change after that failure. Although it is a simple and efficient approach, it relies on a model provided by a specific simulator that may not

[1] A failure that does not cease all function but compromises that function (CCPS, 2023).

[2] Triplex pump: a common type of pump in oil and gas industry used in both drilling and well service operations.

[3] A failure mode is the effect by which a failure is observed (NARESKY, 1970).

include equipment models or allow the simulation of all the failure modes of interest.

The work presented by (WANG et al., 2021) uses a digital twin to enhance the failure prediction process for an autoclave by generating failure data and using it to train neural network predictive models. In contrast with the approach we used, that work applied a digital twin composed of a 3D model of the focused equipment. It also used finite element analysis to solve the differential equations to determine the temperature at any point of the machine. Although it produces precise information about the simulated physical quantity at any point inside the control volume, the method is computationally expensive and provides a precision that may not be necessary in most cases.

Finally, the method requires some criteria to measure how close the synthetic data are to the actual data it intends to emulate. A simple and efficient approach is to directly compare the actual and the simulated data graphically and tune the parameters so the synthetic data fit the curve of real data, as done in (RAO, 2020). However, the difficulty of this method increases with the number of parameters to be set and outputs to fit. Wang and colleagues (WANG et al., 2021) used the coefficient of determination (r-squared) to compare the simulation output with the corresponding real measurement, given the same input data. This method, though, is limited by the number of real measurements to compare with the generated data.

The work of (DANKAR; IBRAHIM; ISMAIL, 2022) presents a large set of metrics to evaluate synthetic data, which includes univariate, bivariate, population, and application fidelity metrics. In univariate fidelity metrics, like Hellinger distance, the distribution of real and synthetic variables is compared, ignoring the other variables. The bivariate fidelity metric considers the relations between variables within the real and synthetic datasets. In this work, the pairwise correlation distance (one metric we chose to evaluate the synthetic data) shows if the relations between variables follow the same trends in real and synthetic data. There are also the population fidelity metrics, which compare the similarity of the entire real and synthetic datasets, and application fidelity score, which is basically a machine-learning model trained with synthetic data that is tested on real data, using the appropriate metric (also a metric we used in the evaluation of synthetic data generated in this work).

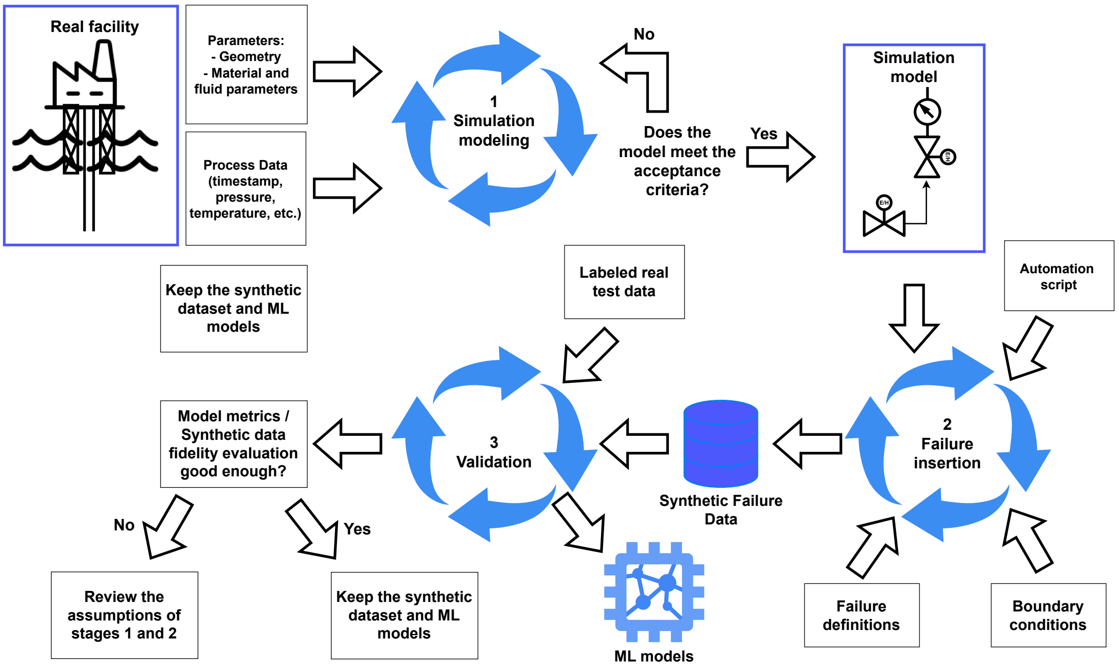
4 METHODOLOGY

Our work aims to generate synthetic failure data that captures the common behavior of an industrial asset over the years of operation. The synthetic data will help in the training stage of a machine-learning algorithm intended to support the detection and prognosis functions of PHM for a petroleum well during its lifetime.

The data generation methodology stands on two main pillars: understanding the physical process, ruled by a set of physical laws that must be respected, and comprehending how the equipment involved in that process works, what are its expected behaviors, and what happens when it fails. To respect those pillars, we apply a computational simulation to ensure the dependencies between variables and the correct behavior of the equipment under the desired conditions.

The methodology proposed in this work may be summarized in three stages: a simulation modeling stage, a failure insertion stage, and a validation stage. Figure 4.1 illustrates the inputs and outputs of the proposed stages. In the following, each of these stages will be detailed.

Figure 4.1 – Methodology proposed for synthetic data generation.


Real facility
Parameters:
- Geometry
- Material and fluid parameters
Process Data (timestamp, pressure, temperature, etc.)
1
Simulation modeling
No
Does the model meet the acceptance criteria?
Yes
Simulation model
Keep the synthetic dataset and ML models
Labeled real test data
Automation script
Model metrics / Synthetic data fidelity evaluation good enough?
3
Validation
Synthetic Failure Data
2
Failure insertion
No
Yes
Review the assumptions of stages 1 and 2
Keep the synthetic dataset and ML models
ML models
Failure definitions
Boundary conditions

4.1 Simulation modeling

This stage aims to obtain a simulation model for reproducing the real system with the necessary accuracy level. To do so, we propose the algorithm 1, with comments about these steps in the following:

Algorithm 1: Simulation modeling algorithm

```
begin Simulation modeling stage
    Data: $PD = (X_1, X_2, ..., X_m, Y_1, Y_2, ..., Y_n)$: Process data (e.g.,
          timestamp, pressure, temperature, fluid parameters.);
    Data: $FD = (P_1, P_2, ..., P_i)$: Facility data (e.g., P&ID diagrams,
          geometry and material parameters.);
    Data: $\epsilon$: Acceptance criteria;
    Result: $S_{model}$: Simulation model artifact; receives boundary conditions
            and returns simulated outputs;
    $S_{model} \longleftarrow BuildModel(FD)$;
    $BoundaryConditions \longleftarrow (X_1, X_2, ..., X_m)$;
    $ProcessOutputs \longleftarrow (Y_1, Y_2, ..., X_n)$;
    $SimulatedOutputs \longleftarrow S_{model}(BoundaryConditions)$;
    $\delta \longleftarrow \|ProcessOutputs - SimulatedOutputs\|$;
    while $\delta > \epsilon$ do
        $FD \longleftarrow Update(FD)$;
        $S_{model} \longleftarrow BuildModel(FD)$;
        $SimulatedOutputs \longleftarrow S_{model}(BoundaryConditions)$;
    end
end
```

1. *Data gathering*: engineering documents and data about the system, like P&ID[1] diagrams, machinery data books, and process data, for instance.

2. *Definition of the simulator*: there is a wide range of simulators available, using different simulation techniques, and suitable for different levels of abstraction. Many simulators provide standard models for common industrial machines like pumps, motors, and valves and make it possible to integrate several of these elements into a system. The simulator must offer the possibility of automating the simulation process.

3. *Assembly of the simulation model*: the engineering documents of the system and the elements provided by the simulation tool may allow the building of a model that represents the system.

[1] A piping and instrumentation diagram (P&ID) is an engineering drawing that describes a process plant with the process equipment and pipes represented by conventional symbols without and no geographical orientation (TOGHRAEI, 2019).

4. *Definition of the simulation parameters*: constants of the system, required to reproduce the asset's characteristics, such as geometry, friction losses, and material compositions. The engineering documents contain the information needed to customize the standard elements provided by the simulator.
5. *Definition of the simulation variables*: the simulation process needs the specification of boundary conditions[2] and simulation outputs. Boundary conditions must come from process data history to check if the simulation model behaves as expected.
6. *Definition of the acceptance criteria*: different levels of precision must be achieved depending on the user's requirements. However, it is important to keep in mind that industrial processes have statistical variability and a narrow criterion may not be adequate to evaluate the simulation model.
7. *Simulation test*: given several boundary conditions scenarios, the outputs must be coherent with the process data available.
8. *Evaluation of simulation model*: the results of the previous step must be consistent with the process data, within the margin defined by the acceptance criteria, before advancing to the next stage. If not, it is necessary to check the simulation parameters and the boundary conditions.

At the end of the application of those steps, an adequate simulation model to generate the synthetic data must be built.

4.2 Failure insertion

The failure insertion stage begins when the simulation outputs are coherent with the real measured process data from the real asset. The expected result is a dataset with a failure or degrading process label to be used for PHM purposes. To accomplish that, the simulation must be run not only with the system working in normal condition, but also in failure or degrading conditions. This approach may be applied for both failure condition and degrading process data generation, with the necessary adaptions. We propose the algorithm 2 followed by comments about the process:

[2] Boundary conditions are constraints necessary for the solution of a boundary value problem. A boundary value problem is a differential equation (or system of differential equations) to be solved in a domain on whose boundary a set of conditions is known (SIMSCALE, 2023).

Algorithm 2: Failure insertion algorithm

```
begin Failure insertion stage
    Data: $failure\_index \longleftrightarrow f(PD, FD)$: failure definition as a relation
          with process variables and facility data;
    Result: $D_{synth}$: Synthetic failure dataset with $N_{obs}$ observations;
    $BC_{train}, BC_{test} \longleftarrow TrainTestSplit(BoundaryConditions)$;
    $BC_{train} \longleftarrow InsertNoise(BC_{train})$;
    for $N_{obs}$ times do
        $failure\_index \longleftarrow None$
        $BC_{train} \longleftarrow InsertNoise(BC_{train})$;
        $S_{model} \longleftarrow BuildModel(FD, failure\_index)$;
        $SimNormalOutputs \longleftarrow S_{model}(BC_{train})$;
        $BC_{train} \longleftarrow InsertNoise(BC_{train})$;
        $FD \longleftarrow Update(FD, failure\_index)$;
        $S_{model} \longleftarrow BuildModel(FD)$;
        $SimFailureOutputs \longleftarrow S_{model}(BC_{train})$;
        $D_{synth} \longleftarrow Append(D_{synth}, BC_{train}, SimNormalOutputs,$
            SimFailureOutputs, failure_index)
    end
end
```

1. *Definition of failure / degrading process*: what is considered a failure or a degraded process in the simulated system context. A common approach is to consider a failure a condition where a system does not execute a required function and each failure is related to a failure mode. In its turn, a degrading process is related to a physical process called failure mechanism[3]. Failure modes insertion allows only detecting failures while degrading process insertion allows also the construction of predictive models.

2. *Relation of failure / degrading process with simulation parameters*: the implications of the failure modes or the failure mechanisms in the simulation parameters. When inserting failure modes, it is acceptable to change only the machine response (a valve not opening, for example). However, to reproduce a failure mechanism, it is necessary to change the system's physical properties (metal fatigue causing cracking, for instance) with the consequent behavior change.

3. *Attribution of a failure label / degradation measure*: for failure detecting purposes a binary label to indicate failure is enough, but a degrading process requires a quantification scale.

4. *Automate the simulation process to generate data, including the following steps*:

[3] Physical or chemical process that results in failure (NARESKY, 1970).

1. *Set boundary conditions for baseline condition*: the boundary conditions inserted into the simulator must be physically consistent with the real conditions.
2. *Simulate baseline condition*: serves as a reference to compare to the fault condition / degraded mode. When inserting failure modes, the baseline must be the normal condition; when inserting degrading processes, the baseline may be a less degraded state.
3. *Register inputs and outputs of the simulation*: the dataset must include the simulation parameters, boundary conditions, and outputs for the baseline scenario.
4. *Set boundary conditions for failure condition / degraded mode*: using the same boundary conditions as in the baseline simulation, but inserting a compatible noise for the period typical for the simulated scenario. For example, if a degrading process evolves from a normal condition within one month, then compatible randomness must be inserted to represent the possible variability of conditions in such a period.
5. *Insert failure / degraded mode*: if inserting a failure mode, it must be a random binary variable (failure or normal), so it makes it possible to compare normal and failure states under similar variability conditions; when inserting a degraded mode, it must be a degradation measure. The step requires the change of the corresponding parameters in the simulator before running the simulation.
6. *Simulate possible failure condition / degrading process*: simulates a possible scenario taking into consideration the randomness of the process and a possible failure or degrading process.
7. *Register the inputs and outputs of the simulation*: the same variables and parameters registered in the baseline simulation, for comparison purposes, as well as the failure label or degradation measure.

At the end of the execution of these steps, a synthetic failure dataset will be produced.

4.3 Synthetic Data Validation

Once the failure insertion stage is complete and produces a synthetic dataset, it is time to validate the generated data and all the assumed premises of emulating failures inside the simulated system. The assessment of the synthetic data is based on the metrics detailed in the work of (DANKAR; IBRAHIM; ISMAIL, 2022). We propose the algoritm 3 followed by explanatory comments:

Algorithm 3: Synthetic data validation algorithm

```
begin Validation stage
    Data: $label$: failure label related to $BC_{test}$;
    Data: $Req$: performance requirements for the machine learning models;
    Result: $ML_{model}$: machine learning model for PHM trained on synthetic
            data;
    $ML_{model} \longleftarrow Train(D_{synth})$;
    $D_{test} \longleftarrow (BC_{test}, failure\_label)$;
    $pred \longleftarrow ML_{model}(D_{test})$;
    $metrics \longleftarrow EvaluateMetrics(pred, label)$;
    if $metrics > Req$ then
        keep the machine learning models
    else
        go back to failure insertion stage and check premises;
        rebuild the synthetic dataset;
        optimize machine learning models;
    end
end
```

1. *Train machine learning models with the synthetic data*: serves as a base for testing the quality of generated dataset with its ultimate purpose of training PHM models. Classification algorithms are used for failure detection and regression algorithms for failure prediction.

2. *Test the trained models on real labeled data*: comparison between the models' prediction and the real output for the same real input data. It is important to remark that the data used in this step must not be the same data used to extract the boundary conditions to produce the synthetic data described in section 4.2.

3. *Evaluate the application fidelity metrics*: the performance of the trained models when applied to real data. For failure detection models, the global accuracy[4] may not be sufficient to validate the produced data due to the imbalanced characteris-

[4] Accuracy is one of the most used metrics to evaluate a classification model, and is defined as the ratio of the number of correct predictions and the number of total predictions (THARWAT, 2020).

tic of failure data, so the F1-score[5] for the failure class should be also taken into consideration.

4. *Evaluate other metrics*: population, bivariate, and attribute metrics compare the synthetic and real datasets enabling adjustments in the process of data generation.

Once the results of application fidelity metrics are considered adequate for the needs of the application, the synthetic data may also be considered suitable for training PHM models.

[5] F1-score is a classification metric that is defined by the harmonic mean of precision (proportion of positive samples that were correctly classified to the total number of positive predicted) and recall (proportion of positive correctly classified to the total number of positive samples) (THARWAT, 2020).

5 EXPERIMENT DESCRIPTION AND RESULTS

This section describes the experiment execution and results obtained in the same order described in chapter 4. Our repository (BDI-UFRGS, 2023) contains the simulation model, scripts, notebooks, and datasets produced for repeatability purposes.

5.1 Simulation modeling

The experiment started gathering data and information about the system to be modeled. The selected asset for our project for the study was a petroleum smart well in production mode[1] with two ICVs. Process data, operational records, equipment data books, and P&ID diagrams were used to model the process in the simulation environment. The data used in this experiment were provided by Libra Consortium, our partner in Petwin Project[2], and were collected from a smart petroleum well located 160 km offshore Rio de Janeiro, Brazil.

The process data were collected by a Plant Information Management System (PIMS), which collects and integrates data from different sources in an industrial plant (KRAFT, 2008). Among the sources used by the PIMS system, pressures and temperatures measured by P/T gauges are of special interest, since these measures allow the calculation of other physical quantities along the system using the laws of fluid dynamics. Pressures and temperatures used in this work were sampled in the PIMS system every 15 seconds for 598 days. However, because of unsynchronized timestamps, we have flattened the timestamps of these data in windows of 30 seconds, so that different measures could be considered synchronized if they were at the same time window, resulting in a dataset with 569003 samples (rows) and 10 features (columns) containing a timestamp and 9 pressure and temperature measures along the fluid's path.

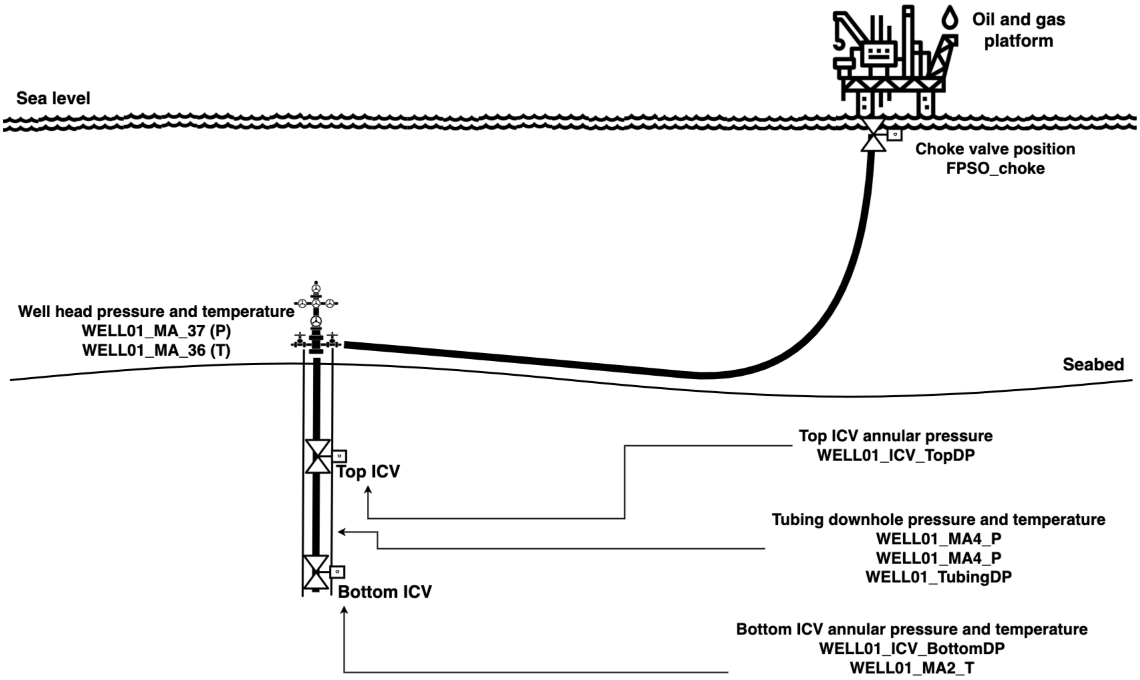
The dataset with process data has been finally combined by date with the information provided in operational reports containing ICVs actuation tests history, which contained labels for normal and failed ICVs actuation tests.

[1] Petroleum wells can also operate in injection mode, injecting water or gas into the reservoir.

[2] For more information, please visit petwin.org.

Figure 5.1 shows the schematic diagram of the modeled system with the measured variables location, table 5.1 shows the metadata of the measured data without any aggregation. Figure 5.2 shows the distribution of null values[3] in the measured dataset as blank spaces in the variable's columns.

Figure 5.1 – Schematic diagram of the modeled system with the location of each measured variable.

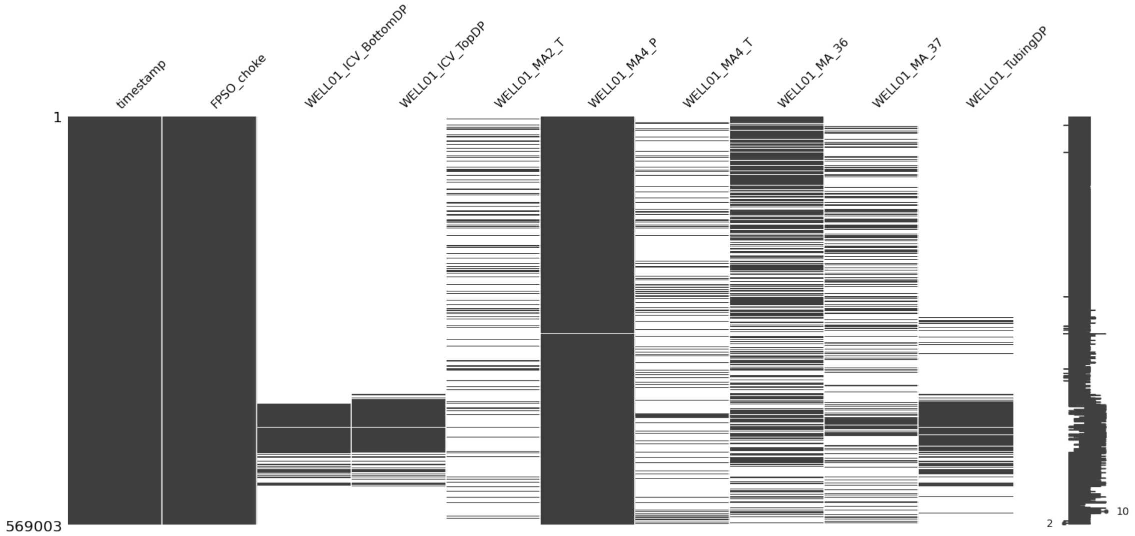


[3] There are different reasons why time series data could possibly be missing when retrieved from a PIMS system. One of the possibilities is that the data were omitted during a period of time due to negligible variance, as a data compression strategy to save storage space. In such a case, the information of that period could be reconstructed with a minimum information loss. That situation differs from others which cause significative information loss, such as sensor failures. It is important to assure if that decompression process could be applied before defining any data imputation strategy.

Table 5.1 – Metadata of the measured data without any aggregation.

Column	Description	Non-Null Count
timestamp	Unsynchronized timestamps.	569003 non-null
FPSO_choke	Choke valve position (%)	569003 non-null
WELL01_ICV_BottomDP	Bottom ICV annular pressure (kgf/cm²)	91563 non-null
WELL01_ICV_TopDP	Top ICV annular pressure (kgf/cm²)	100270 non-null
WELL01_MA2_T	Bottom ICV annular temperature (ºC)	106119 non-null
WELL01_MA4_P	Tubing downhole pressure (kgf/cm²)	568577 non-null
WELL01_MA4_T	Tubing downhole temperature (ºC)	98564 non-null
WELL01_MA_36	Wellhead temperature (ºC)	328529 non-null
WELL01_MA_37	Wellhead pressure (kgf/cm²)	199141 non-null
WELL01_TubingDP	Downhole pressure difference between the annular and the tubing (kgf/cm²).	113496 non-null

Figure 5.2 – Diagram of missing values distribution in raw measured dataset (non-null values are represented as black horizontal lines and missing data are blank spaces in the columns, while the line graph at the right shows the number of non-null values per row).



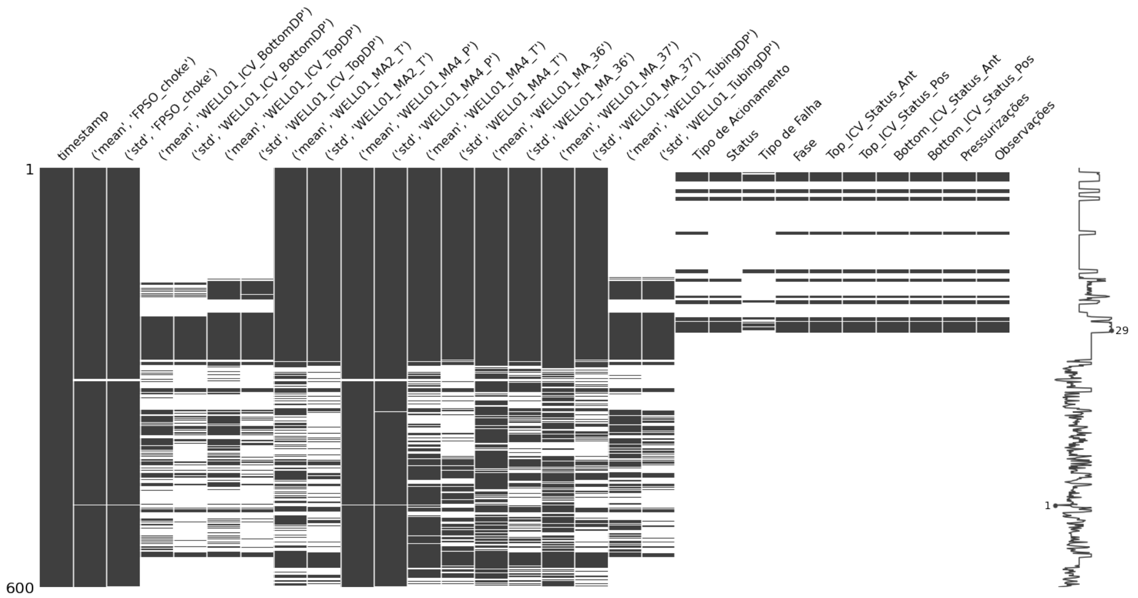
ICVs actuation tests were done on 235 different days, containing 23 failure days and 212 days with normal behavior. However, the available operational reports contain the date of the tests, but not the exact moment when the valves' actuation occurred. Therefore, the time resolution of labels is 1 day, in contrast to the resolution of 30 seconds of the process dataset described earlier. To achieve an appropriate choice of process data to merge with the ICVs actuation tests history, we have proceeded to an aggregation of process data in hourly and daily means. For each variable of the process dataset, we have registered the daily means, the first and last hour mean for each day, as well as the hourly standard deviations. This period of 235 days combined with the corresponding process data has been set apart as a test dataset for the validation stage detailed in section 5.3, while the data of the remaining 363-day period served as the base for the boundary conditions for the simulations executed in section 5.2. Table 5.2 shows the Metadata of the measured data aggregated on a daily basis with means and standard deviations combined with the operational report data, and figure 5.3 shows the distribution of null values[4] in this combined dataset. Appendix B shows an exploratory data analysis of the data used as boundary conditions for the simulations.

[4] The variables WELL01_ICV_BottomDP, WELL01_ICV_TopDP and WELL01_TubingDP have been discarded due to the number of missing values in the first 235 days that served as test data, which was intended to remain with as little manipulation as possible. All the remaining measured data had no missing value in the same period. We have adopted a data imputation with the mode of each measured variable for the latter 363-day period, which had no significant influence in the final achieved result. Nevertheless, the proposed methodology establishes checkpoints to analyze and review assumptions like this in the case of unsatisfactory results.

Table 5.2 – Metadata of the measured data aggregated on a daily basis with means and standard deviations combined with the operational report data.

Column	Description	Non-Null Count
timestamp	Timestamps aggregated with 1-day resolution	600 non-null
(mean, FPSO_choke)	Daily mean of choke valve position (%)	596 non-null
(std, FPSO_choke)	Daily std deviation of choke valve position (%)	594 non-null
(mean, WELL01_ICV_BottomDP)	Daily mean of bottom ICV annular pressure (kgf/cm²)	199 non-null
(std, WELL01_ICV_BottomDP)	Daily std deviation of bottom ICV annular pressure (kgf/cm²)	134 non-null
(mean, WELL01_ICV_TopDP)	Daily mean of bottom ICV annular pressure (kgf/cm²)	221 non-null
(std, WELL01_ICV_TopDP)	Daily std deviation of bottom ICV annular pressure (kgf/cm²)	157 non-null
(mean, WELL01_MA2_T)	Daily mean of bottom ICV annular pressure (kgf/cm²)	445 non-null
(std, WELL01_MA2_T)	Daily std deviation of bottom ICV annular pressure (kgf/cm²)	374 non-null
(mean, WELL01_MA4_P)	Daily mean of tubing downhole pressure (kgf/cm²)	596 non-null
(std, WELL01_MA4_P)	Daily std deviation of tubing downhole pressure (kgf/cm²)	593 non-null
(mean, WELL01_MA4_T)	Daily mean of tubing downhole temperature (ºC)	490 non-null
(std, WELL01_MA4_T)	Daily std deviation of tubing downhole pressure (kgf/cm²)	421 non-null
(mean, WELL01_MA_36)	Daily mean of wellhead temperature (ºC)	528 non-null
(std, WELL01_MA_36)	Daily std deviation of wellhead temperature (ºC)	433 non-null
(mean, WELL01_MA_37)	Daily mean of wellhead pressure (kgf/cm²)	503 non-null
(std, WELL01_MA_37)	Daily std deviation of wellhead pressure (kgf/cm²)	406 non-null
(mean, WELL01_TubingDP)	Daily mean of downhole pressure difference between annular and tubing (kgf/cm²)	241 non-null
(std, WELL01_TubingDP)	Daily std deviation of downhole pressuredifference between annular and tubing (kgf/cm²)	182 non-null
Tipo de Acionamento	Type of valve actuation	65 non-null
Status	Status of actuation	55 non-null
Tipo de Falha	Type of failure	42 non-null
Fase	Stage of the field implementation	65 non-null
Top_ICV_Status_Ant	Status of top ICV before actuation	65 non-null
Top_ICV_Status_Pos	Status of top ICV after actuation	65 non-null
Bottom_ICV_Status_Ant	Status of bottom ICV before actuation	65 non-null
Bottom_ICV_Status_Pos	Status of bottom ICV after actuation	65 non-null
Pressurizações	Pressurization cycle	65 non-null
Observações	Notes	65 non-null

Figure 5.3 – Diagram of missing values distribution in the combined dataset with measured data aggregated with 1-day resolution and operational report data (non-null values are represented as black horizontal lines and missing data are blank spaces in the columns, while the line graph at the right shows the number of non-null values per row).



In our experiment, we used Flownex® Simulation Environment[5], in which the system can be represented by a P&ID diagram. The simulation model has been adjusted to the real data so that the simulation outputs are compatible with real observations, given the same inputs (boundary conditions).

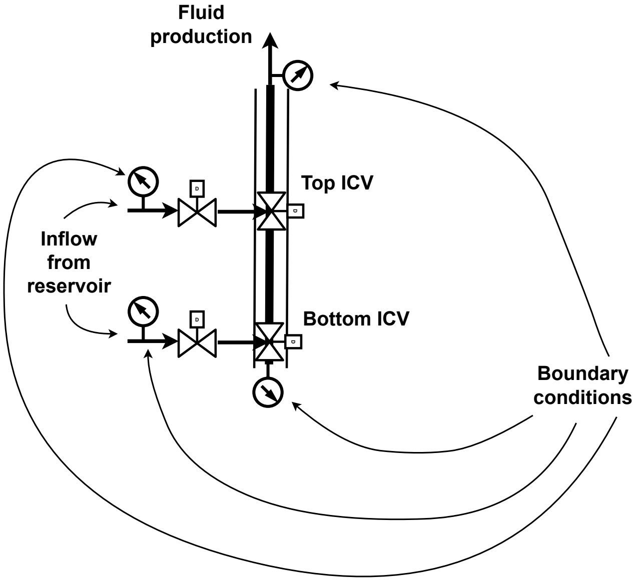
Using such a high-level simulation tool enables the modeling of large systems without concern about describing the details of the system geometry. The simulation parameters reflect all these details in the model components, such as tubes, valves, and heat exchangers. Since ICVs are not included in the simulator's default equipment model database, we have modeled each ICV as a set of two valves as shown in figure 5.4, one of them controlling the influx of fluid from the reservoir to the tubing that carries it to the surface. At the same time, the other one enables the isolation or restriction of flux from upstream production zones inside the tubing, as shown in figure 2.2. Figure 5.5 shows the smart well modeled in the simulator. Tables C.1, C.2, C.3, C.4, and C.5 in appendix C show the main parameters used for each simulation modeling element.

When the simulation model outputs were adequate, we started the synthetic data generation, advancing to the second stage, the failure insertion stage. The simulation

[5] Flownex® Simulation Environment is a one-dimensional simulator that allows for computational fluid dynamic simulations of complete industrial systems, for both steady-state or transient cases. For more details: https://flownex.com/

model produced is available in our repository (BDI-UFRGS, 2023).

Figure 5.4 – Modeling of ICVs in a 1-D simulator.



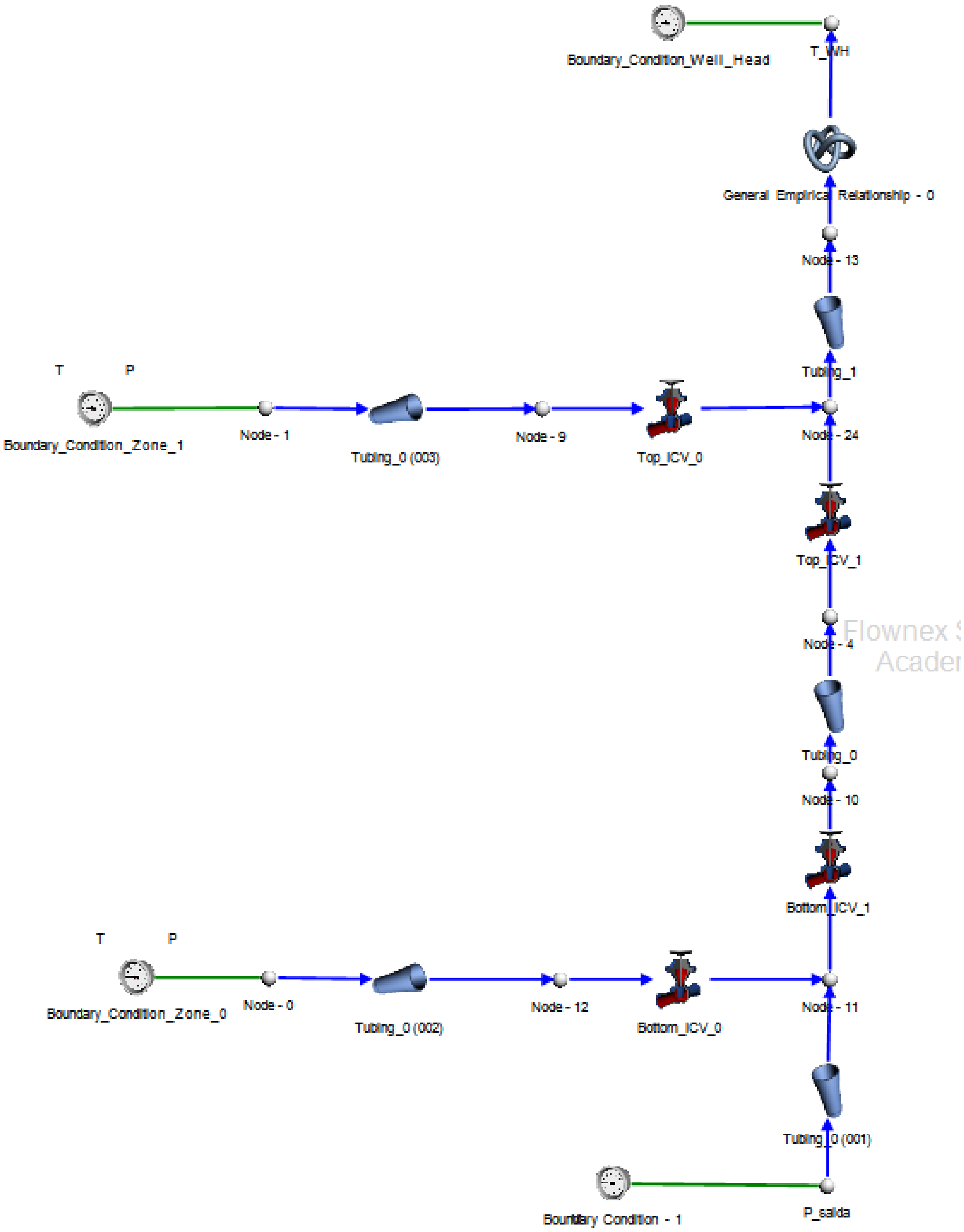
5.2 Failure insertion

In the failure insertion stage, we have inserted failure modes for PHM failure detection models. We have inserted the failures by emulating an abnormal behavior (a failure mode) in one of the two ICVs. An automation script has executed the whole process, interacting with the simulator, changing the simulation parameters, running the stedaystate simulations, collecting the results and registering them in the synthetic dataset. The script randomly decides the action of any of the ICVs, as well as the ocurrence of a failure on that action. This random choice determines the label (failure or normal) of each observation of the synthetic dataset.

The failure definitions adopted for the ICVs were: valves do not respond to the command at all, incomplete actions for both opening or closing, and spurious action when the valves are not required to act. The action may occur or not (50% chance for each possibility), and for each case, a failure may occur or not (also 50% chance for each possibility).

We have modeled these cases in the simulation with a parameter that describes the valve opening. That parameter is called "valve-lift" and varies from 0 for the valve

Figure 5.5 – Modeling of a smart well system in a 1-D simulator.



completely shut to 1 for the valve completely open. Even though ICVs can be set as partially open, in this experiment we considered that an opening actuation changes the "valve-lift" parameter from 0 to 1, and analogously, a closing actuation changes the same parameter from 1 to 0. An incomplete (failed) actuation may have a "valve-lift" between 0.3 and 0.7. Both the statistical distribution of failure modes and the valves' behaviors during the failures are assumptions subject to change according to the results obtained in the next stage.

The steady-state simulation has run twice for each observation created: a baseline simulation in normal condition and a second simulation with a possible inserted failure. The script simulates the baseline steady-state scenario with a set of boundary conditions based on a tuple composed by the corresponding variables at the same instant, chosen randomly from the real data, and then the results are registered. Then the script simulates a second steady-state scenario, with a compatible noise inserted in the boundary conditions. At this point, the script determines the valves' actions and failures at random, avoiding the situation in which both valves may stuck at a closed position, causing a complete production stop as described by (AL-RABEH; AL-NOAIMI; BROWN, 2018). At the end of each this cycle, the script selects another tuple of boundary conditions and repeats the process 10,000 times. All the generated data have been registered in a dataset, available in the project's repository (BDI-UFRGS, 2023).

Our data repository contains the data generated in this stage for download (BDI-UFRGS, 2023). Table 5.3 describes the dataset metadata.

Table 5.3 – Metadata of the synthetic failure dataset generated.

Column	Description	Non-Null Count	Dtype
P_Z0_before	Pressure on the production zone 0, upstream bottom ICV before any valve action.	10000 non-null	float64
P_Z0_after	Pressure on the production zone 0, upstream bottom ICV after any valve action.	10000 non-null	float64
P_Z1_before	Pressure on the production zone 1, upstream top ICV before any valve action.	10000 non-null	float64
P_Z1_after	Pressure on the production zone 1, upstream top ICV after any valve action.	10000 non-null	float64
P_WH_before	Pressure on the wellhead before any valve action.	10000 non-null	float64
P_WH_after	Pressure on the well after any valve action.	10000 non-null	float64
T_Z0_before	Temperature on the production zone 0, upstream bottom ICV before any valve action.	10000 non-null	float64
T_Z0_after	Temperature on the production zone 0, upstream bottom ICV after any valve action.	10000 non-null	float64
T_Z1_before	Temperature on the production zone 1, upstream top ICV before any valve action.	10000 non-null	float64
T_Z1_after	Temperature on the production zone 1, upstream top ICV after any valve action.	10000 non-null	float64
top_icv_status_before	Status of top ICV (open or closed), before any valve action True for open, False for closed.	10000 non-null	bool
top_icv_status_after	Status of top ICV (open or close), after any valve action True for open, False for closed.	10000 non-null	bool
bottom_icv_status_before	Status of bottom ICV (open or close), before any valve action True for open, False for closed.	10000 non-null	bool
bottom_icv_status_after	Status of bottom ICV (open or close), after any valve action True for open, False for closed.	10000 non-null	bool
P_bottom_before	Pressure on the bottom of the tubing. before any valve action (Output of the simulation).	10000 non-null	float64
P_bottom_after	Pressure on the bottom of the tubing. after any valve action (Output of the simulation).	10000 non-null	float64
T_WH_before	Temperature on wellhead before any valve action (Output of the simulation).	10000 non-null	float64
T_WH_after	Temperature on wellhead after any valve action (Output of the simulation).	10000 non-null	float64
command_type	Type of action simulated, with 5 possible values: Open/Close Top or Bottom ICV or no-action.	10000 non-null	object
failure	Indicates if a failure has occurred (True) or not (False).	10000 non-null	bool

5.3 Synthetic Data Validation

The main objective of this stage is to verify the suitability of the synthetic dataset for constructing data-driven PHM models. To achieve this goal, we have used application fidelity metrics and bivariate metrics. Application fidelity metrics use machine learning models trained with synthetic data to classify failure events based on real data. Bivariate fidelity metrics compare the synthetic and real data directly, evaluating how alike the relations between variables are in both datasets.

To assess the application fidelity metrics, we have trained nine machine learning classification algorithms with the data described in table 5.3: Logistic Regression, Decision Tree, Random Forest, KNN, Naive-Bayes, SVC, Adaboost, XGBoost RF, and Multi-layer perceptron. For machine learning modeling we have used the library scikit-learn[6]. The target variable was the failure label, and the models trained on synthetic data should detect if a failure happened in a real dataset. We have also included a dummy model that classifies all the test samples as the most common class (normal). Appendix E contains details about the machine learning modeling process.

The dataset used for testing the models is completely disjoint with the dataset used to extract boundary conditions in the description of section 5.2. The test dataset had to be adequated to the same data structure shown in table 5.3 so that we could generate predictions with the models fed with real data.

We have chosen the global accuracy and also and the F1-score for both failure and normal class as the application fidelity metrics. The pursued goal was to evaluate if the models trained with synthetic data could perform well in detecting failures in both cases. Applying this metric, we verified how well the synthetic data transcripts the desired failure pattern to the point of being detectable by machine learning. Table 5.4 presents the test metrics, while tables 5.5 to 5.14 show the confusion matrices[7] of the tested models. Complete test reports are presented in appendix E.

However, as the application fidelity metrics do not provide any clue about what could be improved in the data generation process, we have chosen a bivariate metric to evaluate if the synthetic dataset respects the relations between the variables of the real dataset. So, we applied the pairwise correlation difference (PCD), a bivariate fidelity met-

[6] More information about the implementation of the chosen algorithms is available in the Scikit-learn library documentation (SCIKIT-LEARN, 2022).

[7] A confusion matrix in a two-class classification problem is a 2 x 2 matrix that shows the counting the four possible outputs of a 2-class classification model (true positive, false positive, true negative, false negative) (THARWAT, 2020)

ric, in its more intuitive form described in (DANKAR; IBRAHIM; ISMAIL, 2022) as the difference between the correlation matrices of the real and the synthetic dataset. As each pair of correlations may vary from -1 to 1, the difference of correlations for each pair of variables will lay in the range from -2 to +2. Ideally, the smaller the absolute value of each PCD element, the better, which means that the synthetic data represents well the relationship of that pair of variables. Analogously, greater values of PCD elements indicate that the model requires an adjustment at that exact point. The intention in applying such a technique was to capture the relationship among the data variables. The simple analysis of excursion range occupation for each variable may not be enough to describe the system correctly. It is important to highlight that physical quantities in real systems frequently cannot be considered independent, since they tie to physical laws that must be respected, e.g., the pressures and temperatures in different points of an oil field. Figure 5.6 shows the pairwise correlation difference between the synthetic and the real dataset heatmap as a bivariate fidelity metric. Appendix D contains univariate comparisons between synthetic and corresponding real data.

Table 5.4 – Metrics of the machine learning models fit to the synthetic data and tested on the real data.

ML Classification Algorithm	**Global accuracy**	**F1-score (macro avg)**	**F1-score (weighted avg)**	**F1-score (failure)**	**F1-score (normal)**	**Precision (failure)**	**Precision (normal)**	**Recall (failure)**	**Recall (normal)**	**Support (failure)**	**Support (normal)**
Dummy	0.902	0.474	0.856	0.000	0.949	0.000	0.902	0.000	**1.000**	23	212
Logistic Regression	0.902	0.474	0.856	0.000	0.949	0.000	0.902	0.000	**1.000**	23	212
Decision Tree	0.940	0.849	0.943	0.731	0.967	0.655	0.981	0.826	0.953	23	212
Random Forest	0.940	0.843	0.943	0.720	0.967	0.667	0.976	0.783	0.958	23	212
KNN	0.102	0.096	0.034	0.173	0.019	0.095	0.667	0.957	0.009	23	212
Naive-Bayes	0.902	0.474	0.856	0.000	0.949	0.000	0.902	0.000	**1.000**	23	212
SVC	0.902	0.474	0.856	0.000	0.949	0.000	0.902	0.000	**1.000**	23	212
Adaboost	**0.953**	**0.890**	**0.957**	**0.807**	**0.973**	**0.676**	**1.000**	**1.000**	0.948	23	212
XGBoost RF	**0.953**	**0.890**	**0.957**	**0.807**	**0.973**	**0.676**	**1.000**	**1.000**	0.948	23	212
Multi-layer perceptron	0.183	0.176	0.236	0.103	0.250	0.058	0.727	0.478	0.151	23	212

Table 5.5 – Confusion matrix of the Dummy model trained on synthetic data and tested on real data.

		Predicted label		
		Failure	Normal	Total
True label	Failure	0	23	23
	Normal	0	212	212
	Total	0	235	235

Table 5.6 – Confusion matrix of the Logistic Regression model trained on synthetic data and tested on real data.

		Predicted label		
		Failure	Normal	Total
True label	Failure	0	23	23
	Normal	0	212	212
	Total	0	235	235

Table 5.7 – Confusion matrix of the Decision Tree model trained on synthetic data and tested on real data.

		Predicted label		
		Failure	Normal	Total
True label	Failure	19	4	23
	Normal	10	202	212
	Total	29	206	235

Table 5.8 – Confusion matrix of the Random Forest model trained on synthetic data and tested on real data.

		Predicted label		
		Failure	Normal	Total
True label	Failure	18	5	23
	Normal	9	202	212
	Total	27	208	235

Table 5.9 – Confusion matrix of the KNN model trained on synthetic data and tested on real data.

		Predicted label		
		Failure	Normal	Total
True label	Failure	22	1	23
	Normal	210	2	212
	Total	232	3	235

Table 5.10 – Confusion matrix of the Naive-Bayes model trained on synthetic data and tested on real data.

		Predicted label		
		Failure	Normal	Total
True label	Failure	0	23	23
	Normal	0	212	212
	Total	0	235	235

Table 5.11 – Confusion matrix of the SVC model trained on synthetic data and tested on real data.

		Predicted label		
		Failure	Normal	Total
True label	Failure	0	23	23
	Normal	0	212	212
	Total	0	235	235

Table 5.12 – Confusion matrix of the Adaboost model trained on synthetic data and tested on real data.

		Predicted label		
		Failure	Normal	Total
True label	Failure	23	0	23
	Normal	11	201	212
	Total	34	201	235

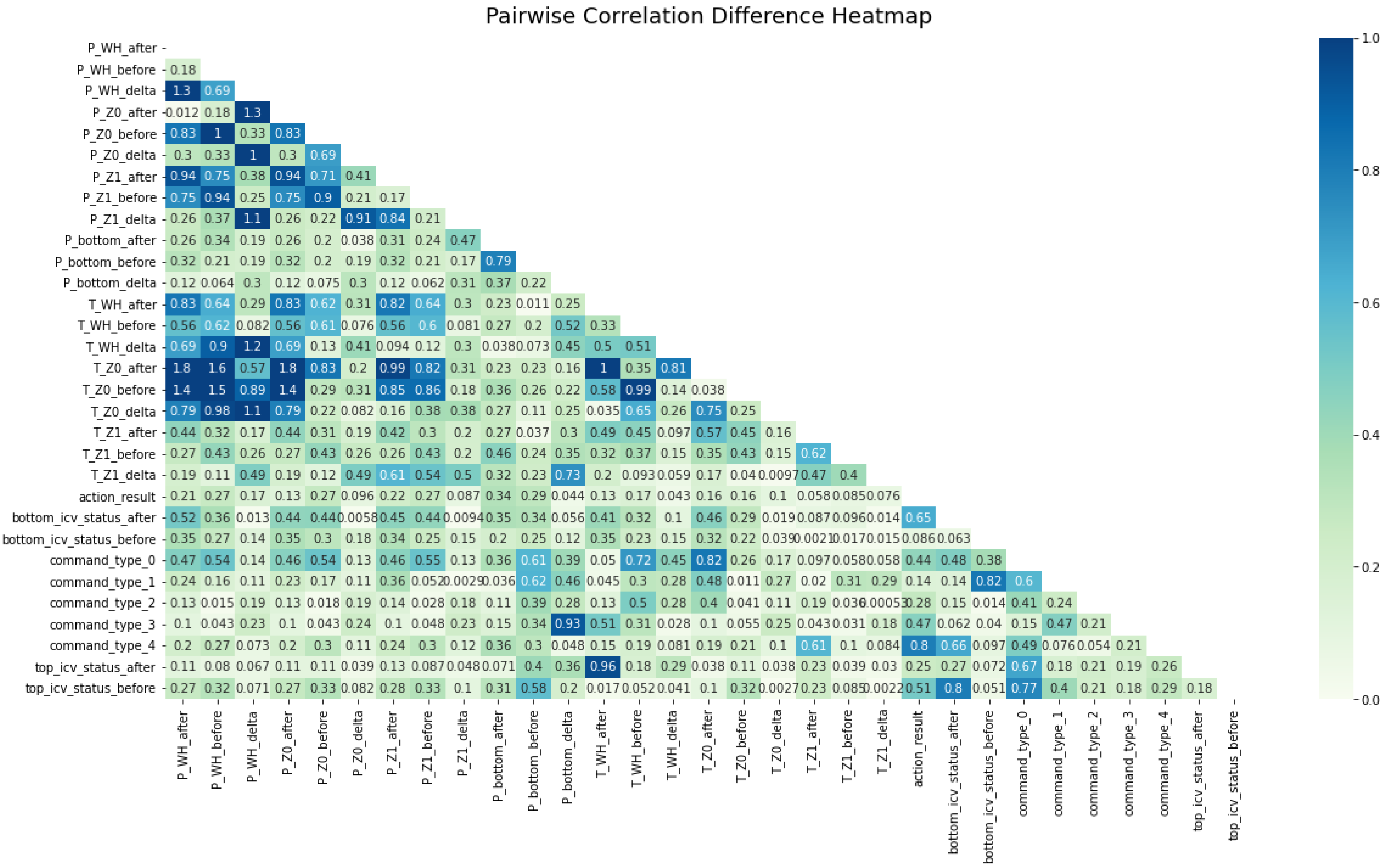
Table 5.13 – Confusion matrix of the XGBoost Random Forest model trained on synthetic data and tested on real data.

		Predicted label		
		Failure	Normal	Total
True label	Failure	23	0	23
	Normal	11	201	212
	Total	34	201	235

Table 5.14 – Confusion matrix of the Multi-layer Perceptron model trained on synthetic data and tested on real data.

		Predicted label		
		Failure	Normal	Total
True label	Failure	11	12	23
	Normal	180	32	212
	Total	191	44	235

Figure 5.6 – Pairwise correlation difference between the synthetic and the real dataset heatmap.



6 DISCUSSION

The metadata about the synthetic data in table 5.3 show, at first inspection, what is expected from a synthetic dataset: no missing values, clean data, and variable types specifically designed for machine learning modeling. Since the data types of synthetic data match the data types of real data, there is not any noticeable difference between them at that stage of the experiment. A thorough comparison between real and synthetic data was not convenient at this point since it would not overcome the need for application fidelity metrics with real labeled data. The validation stage runs a deeper comparison between real and synthetic data by including a bivariate metric.

The application fidelity metrics give the final definition of the suitability of the generated data for producing data-driven failure-detecting models. Table 5.4 displays metrics of machine learning modeling of nine classification algorithms fitted on the synthetic dataset and tested on a real dataset containing 23 ICV action failures registered in 235 days. The first important observation about the metrics in table 5.4 is not analyzing the global accuracy by itself, as more than 90% of the observations in the test dataset refer to the normal class, a typical scarcity scenario of failure data. Because of this scenario, even a Dummy classifier that attributed the normal condition to the inputs had an accuracy score greater than 90%. So, the F1-score of the failure class is a better indicator of how well the synthetic data replicate the real data patterns. It is important to notice that the definition of what is a failure in terms of the equipment's behavior is crucial for the quality of synthetic failure data. In this work, we have taken into account not only no-response failure modes but also half-open valves as failures, which represented 95% of the failures in the synthetic dataset. This definition differs from that used in (LANGLI et al., 2001) and (JOUBRAN, 2018), whose failure definitions were more strict. We considered that a wider failure definition could represent an anomaly in the process that a machine learning model can detect if the failure definition of the synthetic dataset was close enough to the observed in the real data. The results in table 5.4 confirm the hypothesis that at least four algorithms can recognize the failure pattern in real data based on the training on synthetic data with an F1-score metric for failure status greater than 0.7. The confusion matrices of tables 5.7, 5.8, 5.12, and 5.13 also confirm this interpretation. An eventual poor performance of all models fitted to the synthetic data may indicate that the synthetic data did not correctly represent the failure modes.

We can infer from tables 5.4, 5.12, and 5.13 that Adaboost and XGBoost Random

Forest classifiers were the algorithms that have best captured the failure pattern in synthetic data. Both identified correctly 23 out of 34 failure events, while classified all 201 normal situations correctly. The algorithms had no mistakes assigning a normal condition label to a failure condition event, which we believe is the most expensive type of error of a failure detection model. The 11 misclassified events were the same for both algorithms, which brings us an important clue to be further examined. These events may carry a similar failure pattern that was not correctly described in the failure insertion stage. An ideal scenario would be to have more real data to avoid the risk of overfitting the synthetic dataset to a small amount of real data.

Additionally, section 5.3 details a comparison between synthetic and real data using the pairwise correlation difference technique. The analysis of the results for this metric brings insights to enhance the failure insertion process. Figure 5.6 shows a heatmap where the darker colors mean higher differences in correlations for that pair of variables. The map spots pairs of variables whose relationships are not coherent, and it possibly would improve by adjusting the adopted premises for the data generation.

If we highlight the simulation outputs variables from figure 5.6, the result will be the map showed in figure 6.1, from where we can spot the greater correlation differences between correlations of inputs and outputs of the simulations. That may indicate which pairs of variables should be investigated to improve the simulation model, possibly leading us to adjust specific simulation parameters that model the physical relationship between those variables, if needed to improve the synthetic dataset.

On the other hand, highlighting the simulation inputs from figure 5.6 results in the map showed in figure 6.2, which shows us several improvement opportunities. Some of them are autocorrelation differences, i.e., differences between the correlations of the variable itself in different times. However, the majority of high difference spots lay in the zones that relate boundary conditions. All of those difference spots led us to identify an inaccurate premise that may have affected those relationships. The premise was that those variables are independent, but what happens in the real world is that this premise is not true. The boundary conditions in the physical world cannot vary freely like it was assumed in the experiment, because the physical quantities in a petroleum reservoir are tied together by physical laws. The addition of noise between the two instants when the simulations were run is probably the cause of the autocorrelation differences, indicating that the boundary conditions are time-related, i.e., they cannot vary freely from the previous value, as assumed in the experiment. In this case, this was not critical to affect the machine-learning performance of some models because of well-behaved noise inserted in the boundary conditions. However, it remains an opportunity for enhancement if it requires better algorithm performance in this scenario.

The exploration of improvement opportunities may continue as long as necessary for the application of synthetic data. In appendix D we have performed an exploratory data analysis of the synthetic dataset compared with its corresponding real measured data, which may serve as a guide of what is necessary to change in the data generation process.

In the end, the performance obtained in the metrics for assessing the synthetic data will be as good as the applied knowledge in the process modeling, with that knowledge possibly coming from data or domain area experts. A possible drawback of reverse engineering the real failure data too closely is the possible overfitting of the machine learning models to the available data. We can avoid this with a larger set of real data to monitor the fitting process. Without enough data to model the desired process, the approach would rely only upon theoretical or empirical knowledge. The produced data will not be so reliable, but still better than having no data.

Figure 6.1 – Pressure and temperature simulation improvement opportunities (highlighted in black and red, respectively) from the PCD matrix of figure 5.6.

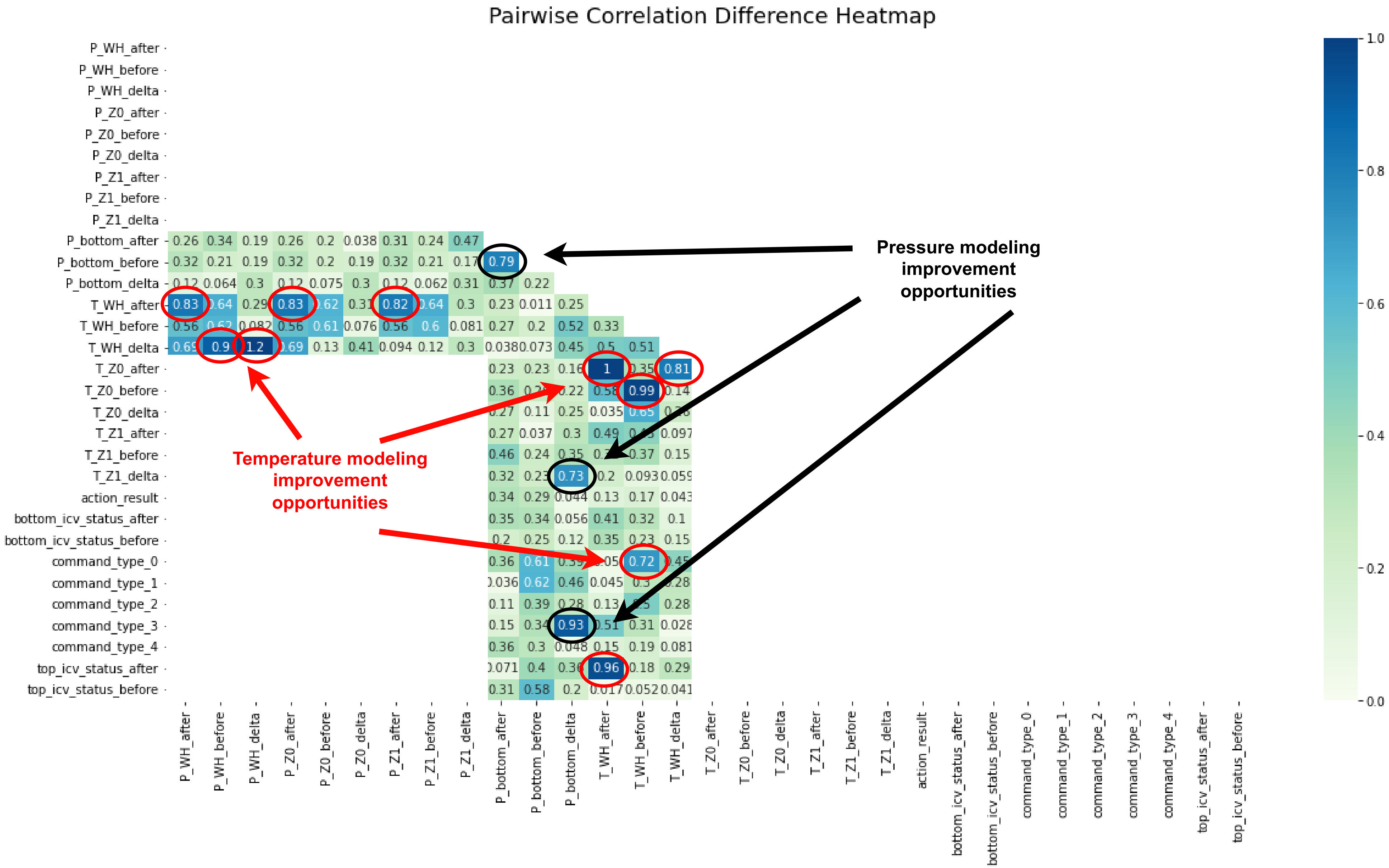
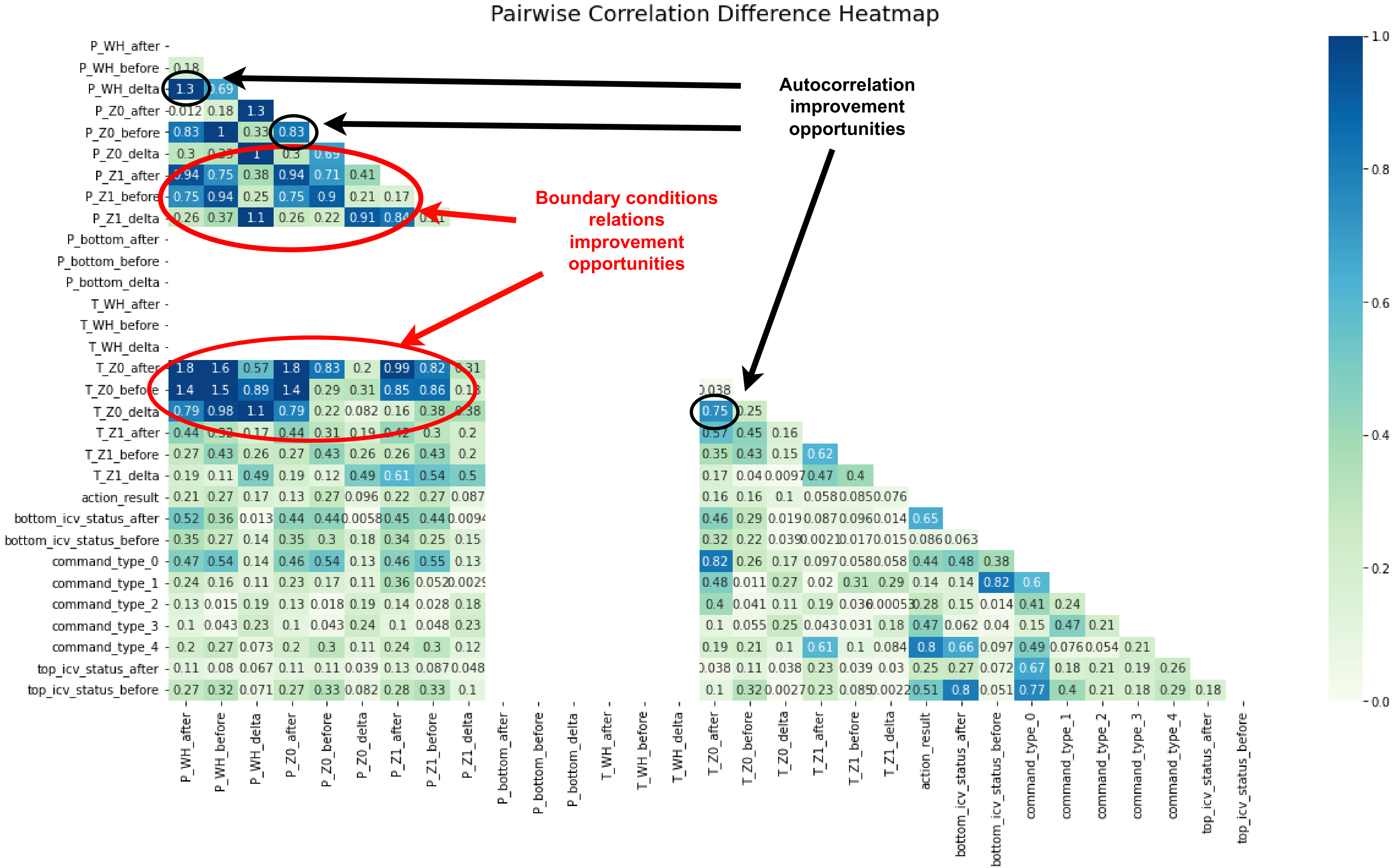


Figure 6.2 – Autocorrelation and boundary conditions relations improvement opportunities (highlighted in black and red, respectively) from the PCD matrix of figure 5.6.



7 CONCLUSIONS AND FUTURE WORK

In this work, we presented a methodology for generating synthetic failure data that can be used for data-driven PHM models validated with an experiment based on real data of a petroleum production smart well. We defined a three-stage process to generate synthetic failure data. The first step is the simulation modeling stage, where a computational fluid dynamics simulator models the process in normal conditions. The method proceeds with a failure insertion stage, where we represent the desired failure modes based on the previous-stage simulation model. The simulation emulates the failure data by modifying the behavior of the equipment that composes the system in the simulation. The last step is the validation stage, in which we assess the synthetic produced data using two approaches. One approach considers the application fidelity, the capacity of training machine-learning models that captures the patterns of the failures in real data, while the other applies a bivariate technique that evaluates the pairwise relations between variables in synthetic and real datasets. In the experiment, we have produced a contribution of a dataset of 10,000 observations available in a public repository, along with the source code of the entire project. We aim that further experiments may use the data for comparison.

Our approach has limitations concerning the generalization of the method for a generic PHM case. The correspondence between the information gathered in the engineering documents and the parameters to build the simulation model has to be made by a specialist, since the documents do not contain all the simulation parameters needed and some inference is usually necessary. Still regarding the generalization of the method, the precise specification of all the necessary engineering information to create a simulation model of a facility allows for the independence of any simulation software. In this moment, the choice of a commercial simulator speeded up the first development. On the other hand, it has limited the customization of the equipment models and, therefore, the possibility of simulating the desired phenomena not covered by the simulator.

Another limitation of the work lies in the assumption of independence between the physical quantities that served as boundary conditions in the simulations. That simplifying hypothesis works well with well-behaved noise conditions but certainly not for the general case, possibly leading to a valid mathematical solution that makes no sense in the real world. For that reason, work is still necessary in the description of the relationships between the variables not handled by the simulator.

It is also important to remark that this work used a simplified simulation model that do not reflect all the possibilities offered by the latest-generation ICVs used in the facility of the use-case. One of these simplifications concerns the valve's opening movement being done in just one step, while the real ICVs of the facility do it in 10 steps. The modeling of such a complex movement would require, therefore, the generation of time-series data encompassing the simulation of transient states between each step.

Also, as the focus of this work is not machine learning, but rather using machine learning to determine if the real data patterns are correctly replicated in synthetic data, the machine learning models are not optimized. Although the proposed goal for this work has been acomplished, we still consider the machine learning solution for generating data-driven PHM models as a limitation.

Future work in this research should address problems in two main areas: simulation techniques and machine learning. Regarding the simulation process, the main issues are the correct specification of engineering parameters (and the consequent independence of any specific simulator) and the automation of the simulation modeling process. There is plenty of research opportunities in the semantic and interoperability of system's engineering data. It is also possible to apply artificial intelligence to automatically test and learn about the influence of different simulation parameters in synthetic data. The solution to these problems would scale up the capacity of twinning different systems and would bring flexibility in generating data at a lower level, modeling the physical phenomena that cause the failures, and not only observing its consequences in the process data. And this would also enable the generation of time series of degraded failures process and time-evolving actions of the machines and their transient reflections, allowing the prediction of the remaining useful life (RUL). Regarding the machine learning issues, there is a wide universe of algorithms to be tested, hyperparameter optimizations, and cross-validation, not to mention the application of time series prediction techniques. In the case of having some more real labeled data, there is a research opportunity in the assessment of the results of models trained with a mix of real and synthetic data compared to models trained with purely real or synthetic data. This way, we could assess the real gain of using synthetic data for data-driven PHM models.

REFERENCES

AL-HAJRI, N. M. et al. Big data analytics maximizes value from smart well completions. In: ONEPETRO. **Abu Dhabi International Petroleum Exhibition & Conference**. [S.l.], 2021.

AL-RABEH, M.; AL-NOAIMI, K.; BROWN, J. A review of industry-wide advanced completion best practices. In: ONEPETRO. **Abu Dhabi International Petroleum Exhibition & Conference**. [S.l.], 2018.

BANJEVIC, D. Remaining useful life in theory and practice. **Metrika**, Springer, v. 69, n. 2, p. 337–349, 2009.

BDI-UFRGS. **BDI-UFRGS Repository**. 2023. <https://github.com/BDI-UFRGS/phm_data_gen>. Accessed in 2023-01-04.

CCPS. **Process Safety Glossary: Degraded Failure**. 2023. <https://www.aiche.org/ccps/resources/glossary/process-safety-glossary/degraded-failure>. Accessed on Jan 17th 2023.

DANKAR, F. K.; IBRAHIM, M. K.; ISMAIL, L. A multi-dimensional evaluation of synthetic data generators. **IEEE Access**, IEEE, v. 10, p. 11147–11158, 2022.

ENO, J.; THOMPSON, C. W. Generating synthetic data to match data mining patterns. **IEEE Internet Computing**, IEEE, v. 12, n. 3, p. 78–82, 2008.

GRIEVES, M. Digital twin: manufacturing excellence through virtual factory replication. **White paper**, Florida Institute of Technology, v. 1, p. 1–7, 2014.

IANNUZZI, M. 15 - environmentally assisted cracking (eac) in oil and gas production. In: RAJA, V.; SHOJI, T. (Ed.). **Stress Corrosion Cracking**. Woodhead Publishing, 2011, (Woodhead Publishing Series in Metals and Surface Engineering). p. 570–607. ISBN 978-1-84569-673-3. Available from Internet: <https://www.sciencedirect.com/science/article/pii/B9781845696733500158>.

JIA, X. et al. A review of phm data competitions from 2008 to 2017: Methodologies and analytics. In: **Proceedings of the Annual Conference of the Prognostics and Health Management Society**. [S.l.: s.n.], 2018. p. 1–10.

JONES, D. et al. Characterising the digital twin: A systematic literature review. **CIRP Journal of Manufacturing Science and Technology**, Elsevier, v. 29, p. 36–52, 2020.

JOUBRAN, J. Intelligent completions: Design and reliability of interval control valves in the past, present, and future. In: ONEPETRO. **Offshore Technology Conference**. [S.l.], 2018.

KRAFT, T. **Systematic and holistic IT project management approach for commercial software with case studies**. Thesis (PhD) — Lawrence Technological University, 2008.

LANGLI, G. et al. Ensuring operability and availability of complex deepwater subsea installations: a case study. In: ONEPETRO. **Offshore Technology Conference**. [S.l.], 2001.

LEE, J. et al. Prognostics and health management design for rotary machinery systems—reviews, methodology and applications. **Mechanical systems and signal processing**, Elsevier, v. 42, n. 1-2, p. 314–334, 2014.

LEE, Y. O.; JO, J.; HWANG, J. Application of deep neural network and generative adversarial network to industrial maintenance: A case study of induction motor fault detection. In: IEEE. **2017 IEEE international conference on big data (big data)**. [S.l.], 2017. p. 3248–3253.

LUO, W. et al. A hybrid predictive maintenance approach for cnc machine tool driven by digital twin. **Robotics and Computer-Integrated Manufacturing**, Elsevier, v. 65, p. 101974, 2020.

MAUTHE, F.; HAGMEYER, S.; ZEILER, P. Creation of publicly available data sets for prognostics and diagnostics addressing data scenarios relevant to industrial applications. **International Journal of Prognostics and Health Management**, v. 12, n. 2, 2021.

MIN, Q. et al. Machine learning based digital twin framework for production optimization in petrochemical industry. **International Journal of Information Management**, Elsevier, v. 49, p. 502–519, 2019.

MOBLEY, R. K. **An introduction to predictive maintenance**. [S.l.]: Elsevier, 2002.

MULLER, A.; MARQUEZ, A. C.; IUNG, B. On the concept of e-maintenance: Review and current research. **Reliability Engineering & System Safety**, Elsevier, v. 93, n. 8, p. 1165–1187, 2008.

NARESKY, J. J. Reliability definitions. **IEEE Transactions on Reliability**, IEEE, v. 19, n. 4, p. 198–200, 1970.

RAHMAN, J.; ALLEN, C.; BHAT, G. Second-generation interval control valve (icv) improves operational efficiency and inflow performance in intelligent completions. In: ONEPETRO. **North Africa Technical Conference and Exhibition**. [S.l.], 2012.

RAO, S. V. Using a digital twin in predictive maintenance. **Journal of Petroleum Technology**, OnePetro, v. 72, n. 08, p. 42–44, 2020.

SCIKIT-LEARN. **Scikit-learn - Machine Learning in Python**. 2022. <https://scikit-learn.org/stable/index.html>. Accessed on Dec 10th 2022.

SHAW, J. Comparison of downhole control system technologies for intelligent completions. In: ONEPETRO. **Canadian Unconventional Resources Conference**. [S.l.], 2011.

SI, X.-S. et al. Remaining useful life estimation–a review on the statistical data driven approaches. **European journal of operational research**, Elsevier, v. 213, n. 1, p. 1–14, 2011.

SIMSCALE. **What Are Boundary Conditions?** 2023. <https://www.simscale.com/docs/simwiki/numerics-background/what-are-boundary-conditions/>. Accessed on Jan 17th 2023.

TAO, F. et al. Digital twin in industry: State-of-the-art. **IEEE Transactions on Industrial Informatics**, IEEE, v. 15, n. 4, p. 2405–2415, 2018.

TAO, F.; ZHANG, M. Digital twin shop-floor: A new shop-floor paradigm towards smart manufacturing. **IEEE Access**, v. 5, p. 20418–20427, 2017.

THARWAT, A. Classification assessment methods. **Applied Computing and Informatics**, Emerald Publishing Limited, 2020.

TOGHRAEI, M. **Piping and Instrumentation Diagram Development**. [S.l.]: John Wiley & Sons, 2019.

VARGAS, R. E. V. et al. A realistic and public dataset with rare undesirable real events in oil wells. **Journal of Petroleum Science and Engineering**, Elsevier, v. 181, p. 106223, 2019.

WANASINGHE, T. R. et al. Digital twin for the oil and gas industry: Overview, research trends, opportunities, and challenges. **IEEE Access**, IEEE, v. 8, p. 104175–104197, 2020.

WANG, Y. et al. Digital twin enhanced fault prediction for the autoclave with insufficient data. **Journal of Manufacturing Systems**, Elsevier, v. 60, p. 350–359, 2021.

YETEN, B. et al. Decision analysis under uncertainty for smart well deployment. **Journal of Petroleum Science and Engineering**, Elsevier, v. 44, n. 1-2, p. 175–191, 2004.

ZIO, E. Prognostics and health management (phm): Where are we and where do we (need to) go in theory and practice. **Reliability Engineering & System Safety**, Elsevier, v. 218, p. 108119, 2022.

APPENDIX A — RESUMO EXPANDIDO

Prognostics and Health Management (PHM) é uma disciplina de engenharia cujo principal objetivo é fornecer uma visão integrada do estado de saúde de uma máquina ou de um sistema geral. A base do PHM tem suas raízes nos conceitos de engenharia de manutenção, como manutenção preventiva, manutenção centrada na confiabilidade e manutenção baseada em condição (LEE et al., 2014). Para atingir tal objetivo, o PHM utiliza sensores para monitorar o sistema de interesse e então aplica diferentes algoritmos para avaliar a condição do ativo. Desta forma, os sistemas PHM dão suporte a decisões técnicas para melhorar a rentabilidade do ativo.

No contexto da Indústria 4.0, o PHM é um dos principais serviços para aumentar a confiabilidade e a produtividade dos ativos industriais, permitindo detecção, diagnóstico, avaliação e previsão (JIA et al., 2018), melhorando a confiabilidade e reduzindo o tempo de inatividade do ativo. O PHM é baseado em conceitos e técnicas usadas na manutenção preditiva[1], que já é usada para apoiar decisões de gerenciamento de ativos industriais há décadas. Uma abordagem moderna baseada em dados de PHM pode se beneficiar da modelagem de aprendizado de máquina (LUO et al., 2020), permitindo a modelagem de sistemas mais complexos usando não apenas modelos teóricos e dados de equipamentos, mas também dados de processos complexos. O problema é que essa abordagem requer uma grande quantidade de dados de falha que geralmente não estão disponíveis. A raridade de dados de falha é um grande desafio na área de PHM atualmente (MAUTHE; HAGMEYER; ZEILER, 2021). Esse é uma característica inerente aos dados de saúde das máquinas, principalmente porque as máquinas geram dados de estado normal na absoluta maioria do tempo.

Em relação a esse problema, o conceito de gêmeo digital, implementado no contexto da Indústria 4.0, pode oferecer uma solução viável. Intuitivamente, um gêmeo digital é um sistema composto por um ativo físico e sua réplica digital (entidade virtual) que podem interagir entre si. (TAO; ZHANG, 2017) afirma que um gêmeo digital completo deve abranger cinco dimensões: uma parte física, uma parte virtual, conexão, dados e serviço. Para fins de PHM, um gêmeo digital deve usar várias fontes de dados: sensores em tempo real, histórico de manutenção, planos de manutenção, análise de falhas, manuais do fabricante da máquina e modelos de simulação. Dessa forma, os gêmeos digitais permitem a criação de vários se cenários em seu mundo virtual para suportar a otimização

[1] De acordo com (MOBLEY, 2002), a manutenção preditiva é uma manutenção preventiva orientada por condição.

operacional do ativo do mundo real.

Com base nesse princípio dos gêmeos digitais, este trabalho propõe uma solução para a falta de dados de falha de equipamentos, cujas principais contribuições são:

- Uma metodologia geral para geração de dados sintéticos de falha;
- Descrição do experimento aplicando a metodologia proposta;
- Um conjunto de dados de falha sintético produzido no experimento descrito, validado usando dados reais.

A metodologia de geração de dados se baseia em dois pilares principais: compreender o processo físico, regido por um conjunto de leis físicas que devem ser respeitadas, e compreender como funcionam os equipamentos envolvidos nesse processo, quais os seus comportamentos esperados e o que acontece quando falha . Para garantir o respeito a esses pilares, foi utilizada na metodologia simulação computacional fluidodinâmica para garantir as dependências entre variáveis e o correto comportamento do equipamento nas condições desejadas. A metodologia pode ser resumida em três etapas: uma etapa de modelagem da simulação, uma etapa de inserção da falha e uma etapa de validação. A seguir, cada uma dessas etapas será detalhada.

O experimento conduzido para validação da metodologia proposta foi baseado na replicação de um poço inteligente de petróleo e gás natural. Um poço inteligente é um poço dotado de instrumentação no fundo de poço (sensores e válvulas para controle de vazão) que permite o ajuste da produção por monitoramento contínuo de vazões e pressões de fluidos e ajustes da configuração da válvula (YETEN et al., 2004). Esses equipamentos garantem flexibilidade e previsibilidade na produção do poço, pois permitem misturar fontes de fluxos de fluidos de diferentes reservatórios com diferentes propriedades químicas e físicas, como qualidade do óleo e proporções de óleo, gás, água e sedimentos.

Para tanto, foram utilizados dados de processo de um poço de petróleo inteligente (incluindo medições de sensores, diagramas de plantas e anotações operacionais) e um simulador comercial unidimensional (1D) de dinâmica de fluidos computacional (CFD) com uma API para automatizar o processo de simulação para gerar dados de falha. Mais tarde, foi realizada validação utilizando métricas de fidelidade do aplicativo para verificar o desempenho dos modelos de aprendizado de máquina treinados em dados sintéticos quando testados em dados de teste reais não vistos. Finalmente, foi utilizada a diferença de correlação par-a-par (PCD) para avaliar o quão bem os dados sintéticos se ajustam aos dados de teste reais.

APPENDIX B — EXPLORATORY DATA ANALYSIS OF BOUNDARY CONDITIONS

Next, we present some exploratory data analysis about the physical quantities used as boundary conditions in the simulations executed in this work. Figure B.1 shows a correlation map between the boundary conditions. Figures B.2, B.3, B.4, B.5, and B.6 show histograms of boundary conditions with 15 bins. Finally, section B.1 presents basic descriptive statistics about each variable.

Figure B.1 – Pairwise correlation map between boundary conditions aggregated by daily means.

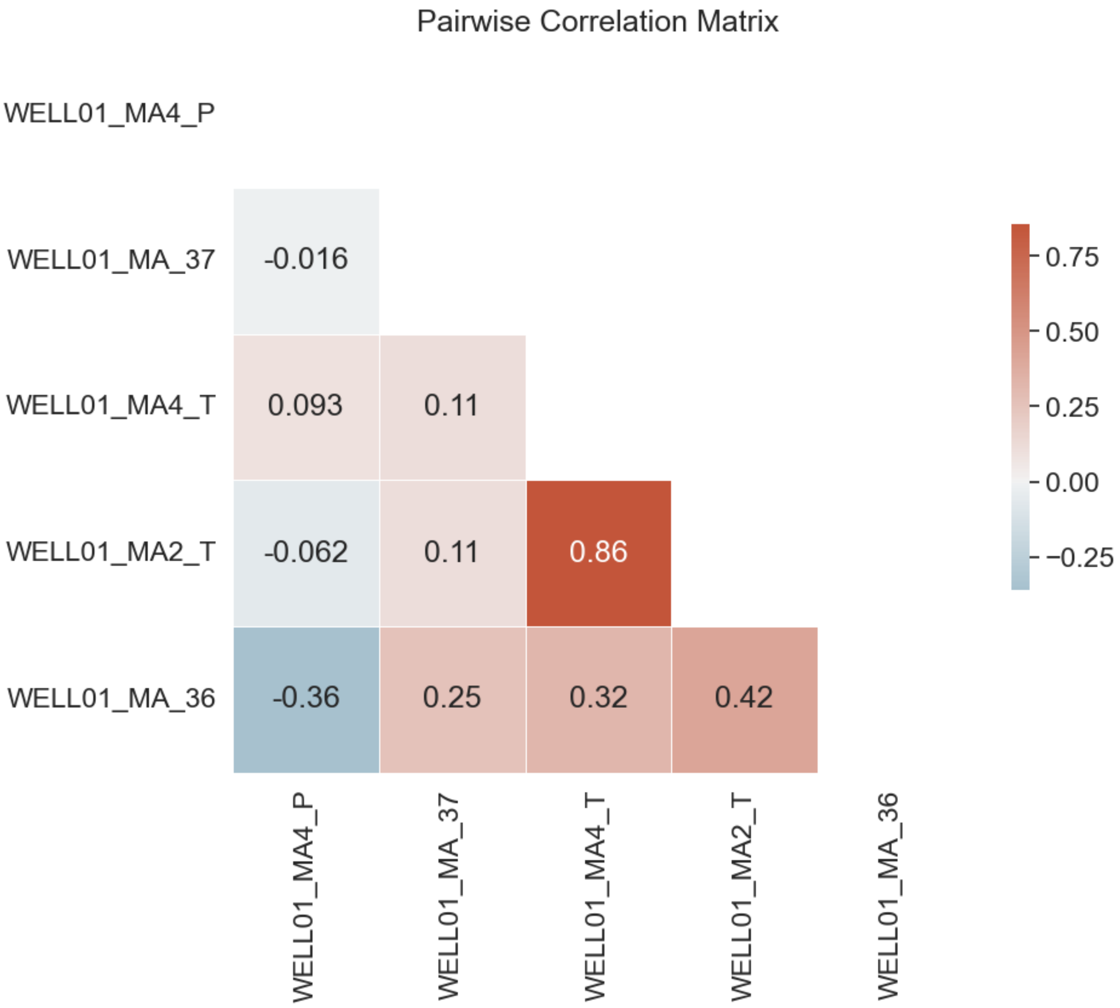


Figure B.2 – Histogram of WELL01_MA4_P with 15 bins.

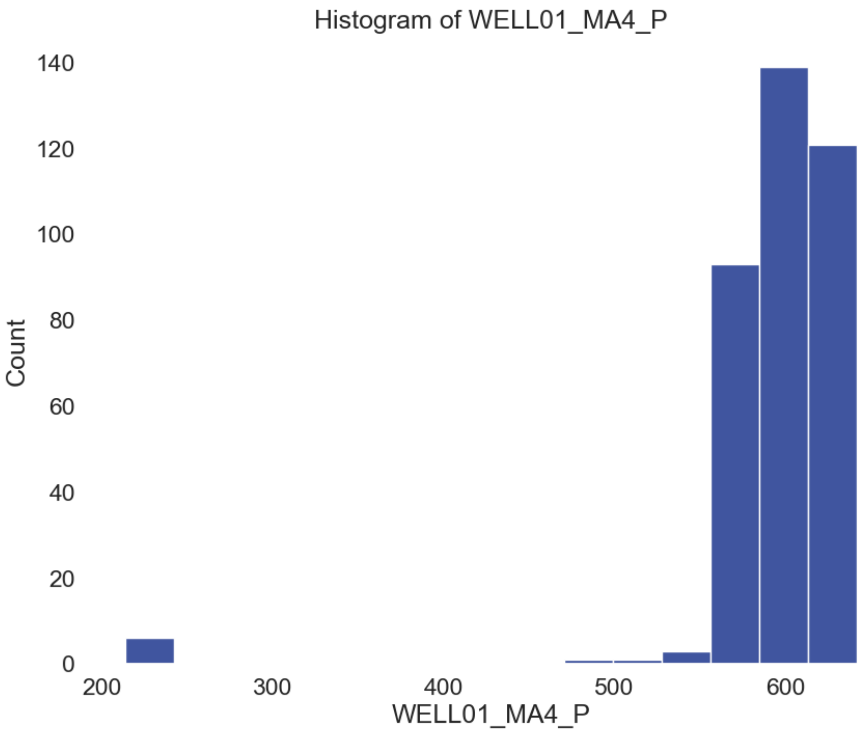


Figure B.3 – Histogram of WELL01_MA_37 with 15 bins.

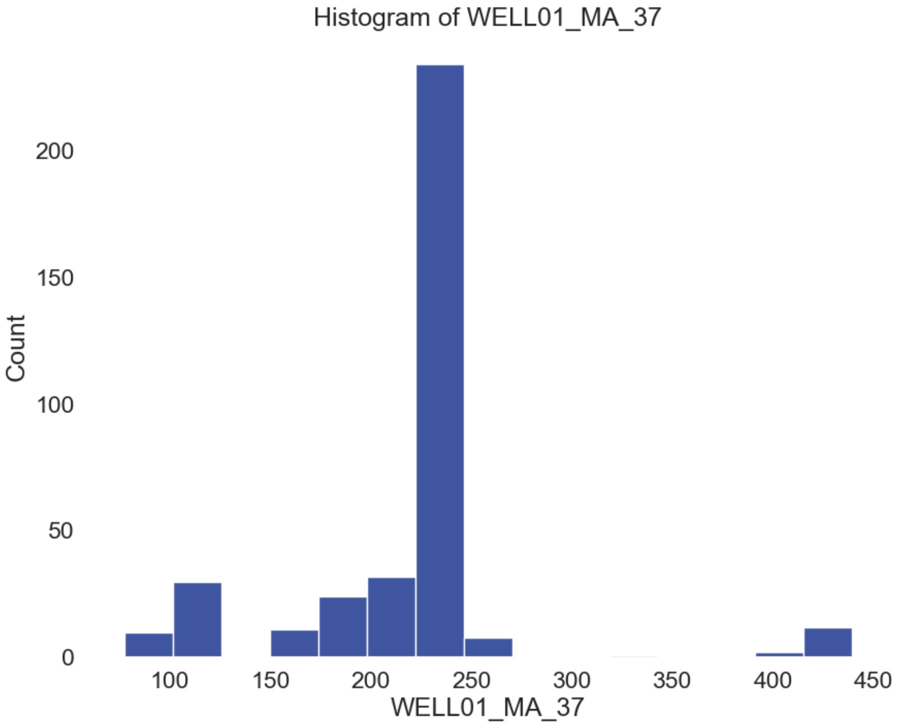


Figure B.4 – Histogram of WELL01_MA4_T with 15 bins.

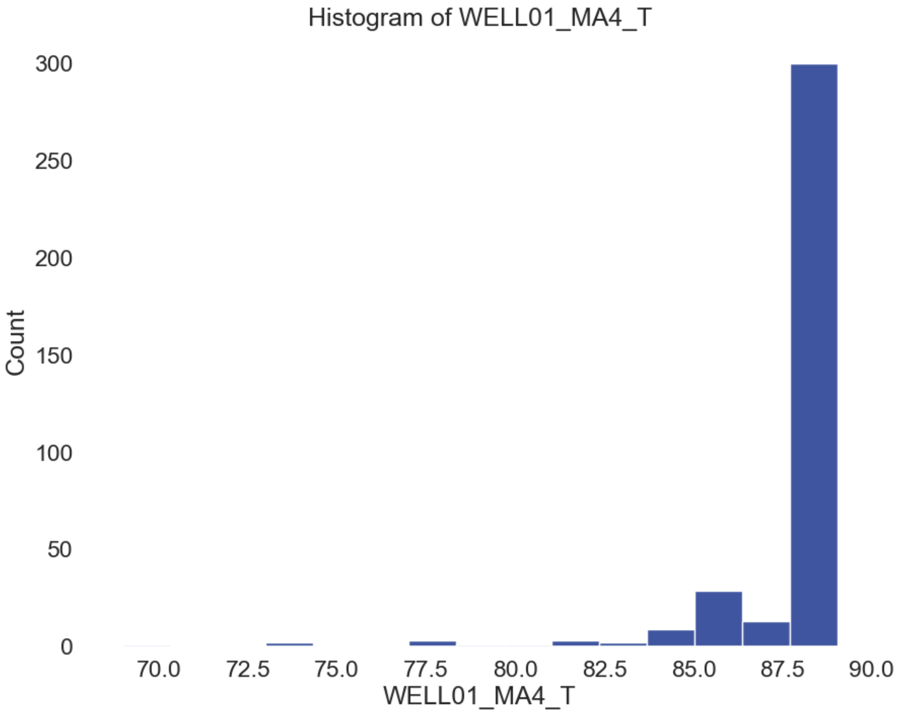


Figure B.5 – Histogram of WELL01_MA2_T with 15 bins.

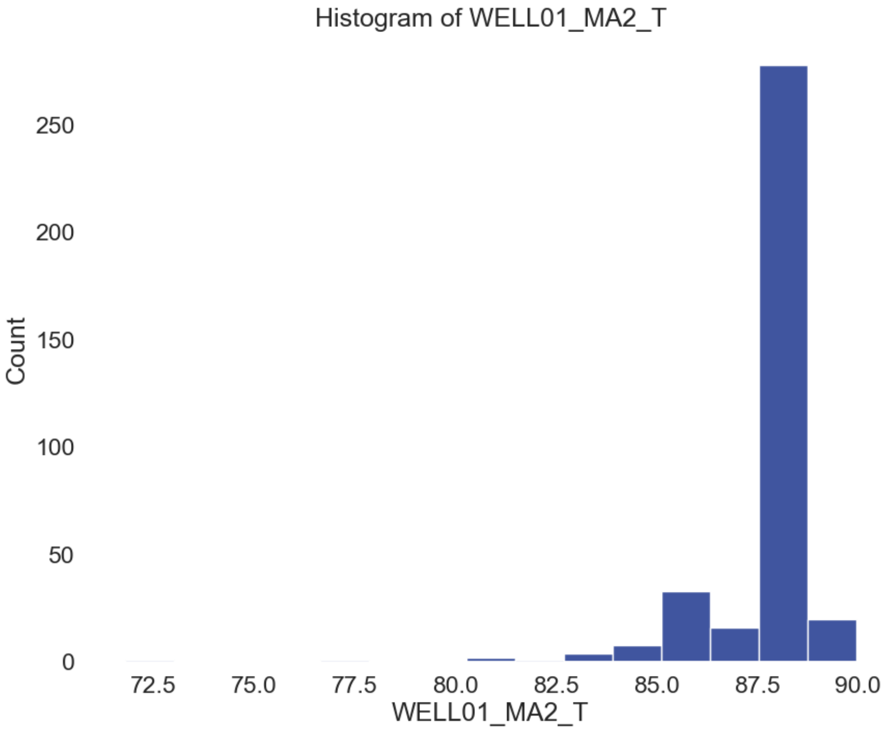
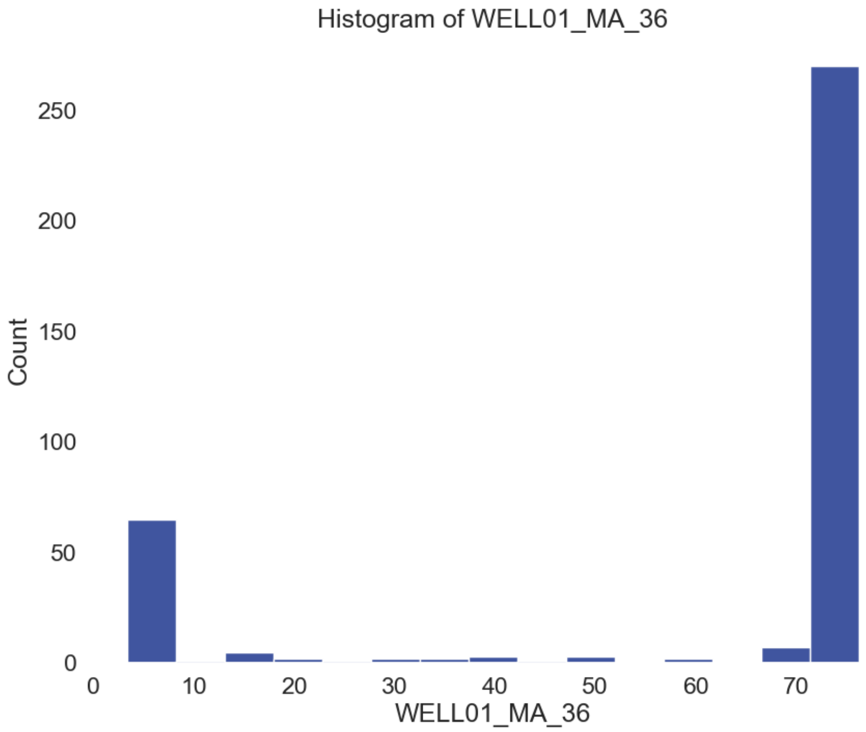


Figure B.6 – Histogram of WELL01_MA_36 with 15 bins.



B.1 Descriptive statistics

```
------------------------------------------------------------

Variable: WELL01_MA4_P

Number of observations: 364

Number of missing values: 0

Number of distinct values: 360

Number of zeroes: 0

---

Maximum value: 641.72585

Q3: 639.384645

Average: 596.5061395451381

Median: 587.2375464870826

Q1: 584.2999653846153

Minimum value: 213.78626666666665
```

```
---
Range: 427.9395833333334
IQR: 55.08467961538463
STD: 56.160705600043634
Kurtosis: 32.05672586643362
Skewness: -5.045910176555036


------------------------------------------------------------
Variable: WELL01_MA_37
Number of observations: 364
Number of missing values: 0
Number of distinct values: 189
Number of zeroes: 0
---
Maximum value: 438.95874215246636
Q3: 226.1285
Average: 215.86473455050654
Median: 226.08739739663093
Q1: 207.5724462280448
Minimum value: 77.668765
---
Range: 361.28997715246635
IQR: 18.55605377195519
STD: 56.87460288839495
Kurtosis: 5.352871930022012
Skewness: 1.0896284290154534


------------------------------------------------------------
Variable: WELL01_MA4_T
Number of observations: 364
Number of missing values: 0
Number of distinct values: 238
Number of zeroes: 0
---
Maximum value: 89.03499500000001
Q3: 88.41497249999999
Average: 87.65971531061786
Median: 88.32518
Q1: 88.21486984693775
Minimum value: 68.98426255050505
---
Range: 20.05073244949496
IQR: 0.2001026530622454
STD: 2.164474280956942
Kurtosis: 27.53903096430728
Skewness: -4.685514510112574


------------------------------------------------------------
Variable: WELL01_MA2_T
Number of observations: 364
Number of missing values: 0
Number of distinct values: 195
Number of zeroes: 0
---
Maximum value: 89.9614979264214
Q3: 88.71454
Average: 88.03292071619319
Median: 88.71454
Q1: 88.31804954545456
Minimum value: 71.81522229865772
---
Range: 18.146275627763686
IQR: 0.39649045454544307
STD: 1.6594244755813068
Kurtosis: 30.419554142787867
```

```
Skewness: -4.473348603123727


------------------------------------------------------------
Variable: WELL01_MA_36
Number of observations: 364
Number of missing values: 0
Number of distinct values: 171
Number of zeroes: 0
---
Maximum value: 76.23967
Q3: 75.61983
Average: 59.93328930983974
Median: 75.46804866666666
Q1: 69.783055
Minimum value: 3.3534633333333335
---
Range: 72.88620666666667
IQR: 5.836774999999989
STD: 28.46962311844858
Kurtosis: -0.03230227314269252
Skewness: -1.3679153569830511
```

APPENDIX C — SIMULATION PARAMETERS

Table C.1 – Parameters of pipe elements in the simulation model of figure 5.5.

	Tubing_0	Tubing_1	Tubing_2	Path_0	Path_1
Length	500 m	500 m	1727 m	500 m	500 m
Upstream Node	P_out	Node - 3	Node - 7	Node - 0	Node - 5
Downstream Node	Node - 2	Node - 4	Node - 8	Node - 1	Node - 6
Diameter	5.92 in				
Variable area	No				
Primary Loss Type	Darcy Weisbach				
Roughness	0,183				
Secondary Losses	No				

Table C.2 – Parameters of valve elements in the simulation model of figure 5.5.

	Bottom_ICV_0	Bottom_ICV_1	Top_ICV_0	Top_ICV_1
Upstream Node	Node - 1	Node - 2	Node - 6	Node - 4
Downstream Node	Node - 2	Node - 3	Node - 7	Node - 7
Valve lift / fraction opening	variable for each simulation round			
Downstream pipe diameter	5.92			
Valve diameter	5			
Upstream pipe diameter	5.92			
Force zero flow when fully closed	Yes			

Table C.3 – Elevation of nodes in figure 5.5 relative to the bottom of the well.

Node	Elevation
P_out	0
Node_0	300 m
Node_1	300 m
Node_2	500 m
Node_3	500 m
Node_4	1000 m
Node_5	800 m
Node_6	800 m
Node_7	1000 m
Node_8	2727 m
T_WH	2727 m

Table C.4 – Parameters of boundary condition elements in the simulation model of figure 5.5.

Boundary condition location	Bottom	Zone_0	Zone_1	Well_Head
Pressure boundary condition	Not specified	Fixed on user total value (value = variable for each simulation round)	Fixed on user total value (value = variable for each simulation round)	Fixed on user total value (value = variable for each simulation round)
Temperature boundary condition	Not specified	Fixed on user total value (value = variable for each simulation round)	Fixed on user total value (value = variable for each simulation round)	Not specified
Mass source boundary condition	Fixed on user value (value = 0)	Not specified	Not specified	Not specified
Enthalpy boundary condition	Not specified	Not specified	Not specified	Not specified

Table C.5 – Parameters of empirical relationship of heat exchange.

Parameter	Value
Upstream node	Node - 8
Downstream node	T_WH
Ck	1
Beta	1
Alpha	1
Heat input	10000

APPENDIX D — EXPLORATORY DATA ANALYSIS OF THE SYNTHETIC DATASET PRODUCED VERSUS ITS REAL CORRESPONDING MEASURED VARIABLES

Here we present an exploratory data analysis comparing the synthetic dataset produced in this work with its corresponding real measures. Figures D.1 and D.2 show the correlation maps between the variables of each dataset. Figures D.3 to D.23 show superposed histograms of each variable of synthetic and real datasets with 15 bins. Finally, section D.1 presents basic descriptive statistics about each variable of both real and synthetic datasets.

Figure D.1 – Pairwise correlation map between synthetic dataset variables.

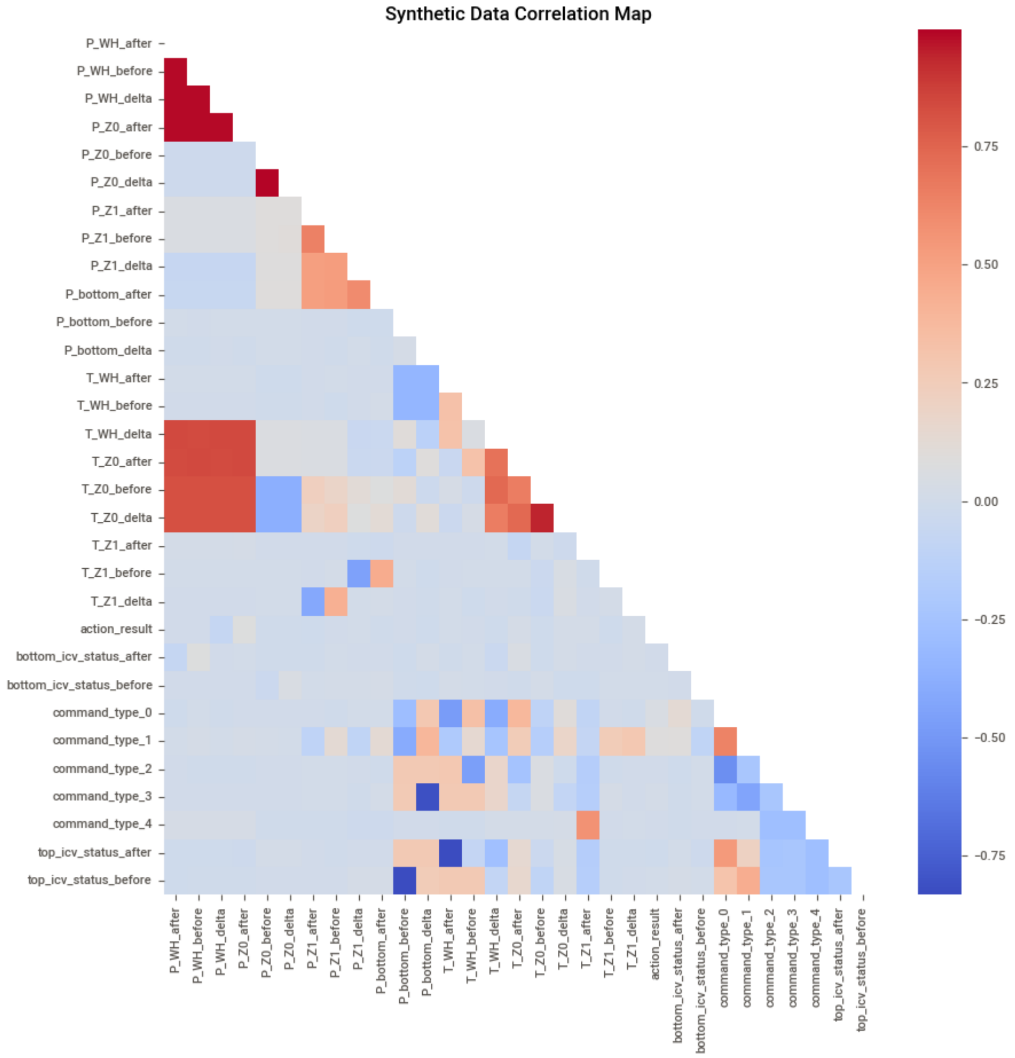


Figure D.2 – Pairwise correlation map between real dataset corresponding variables.

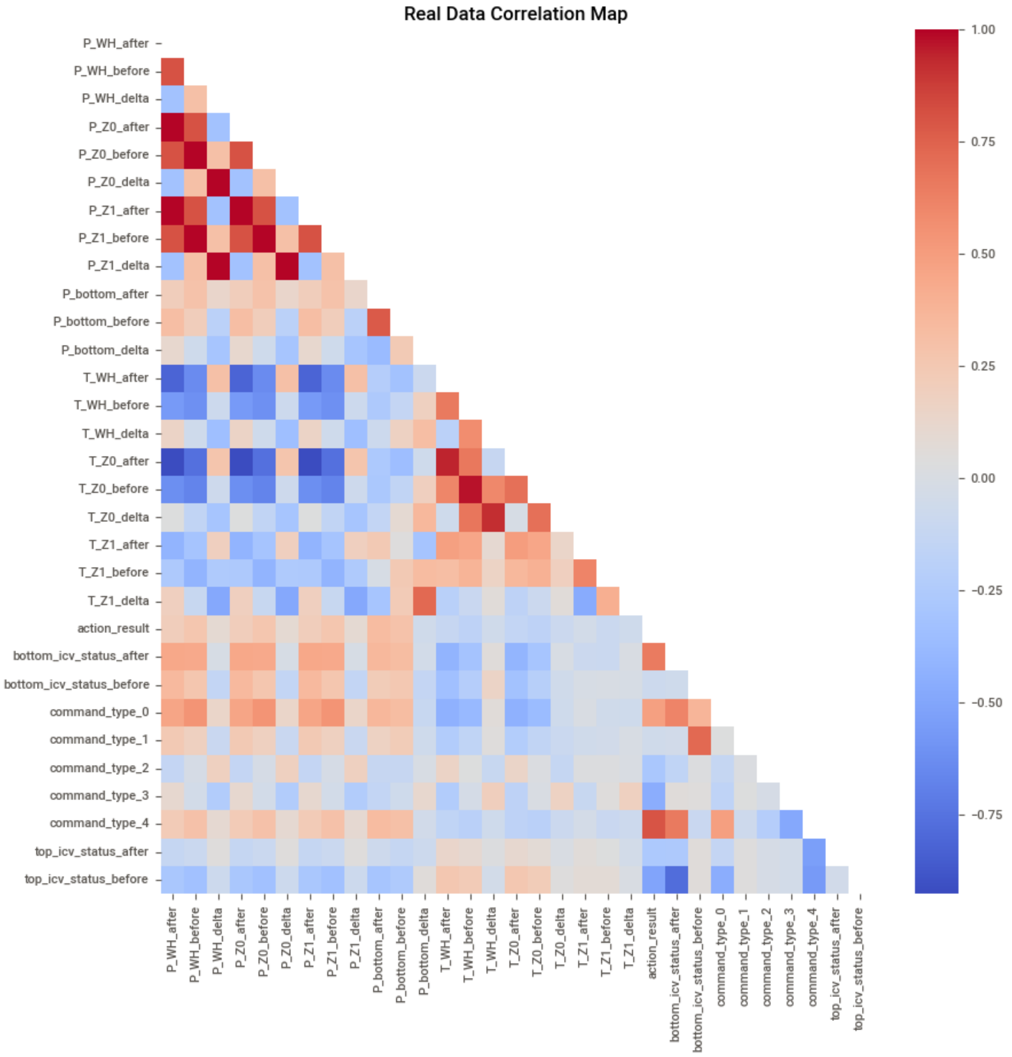


Figure D.3 – Histograms of synthetic (orange) and real (blue) P_WH_after variables with 15 bins.

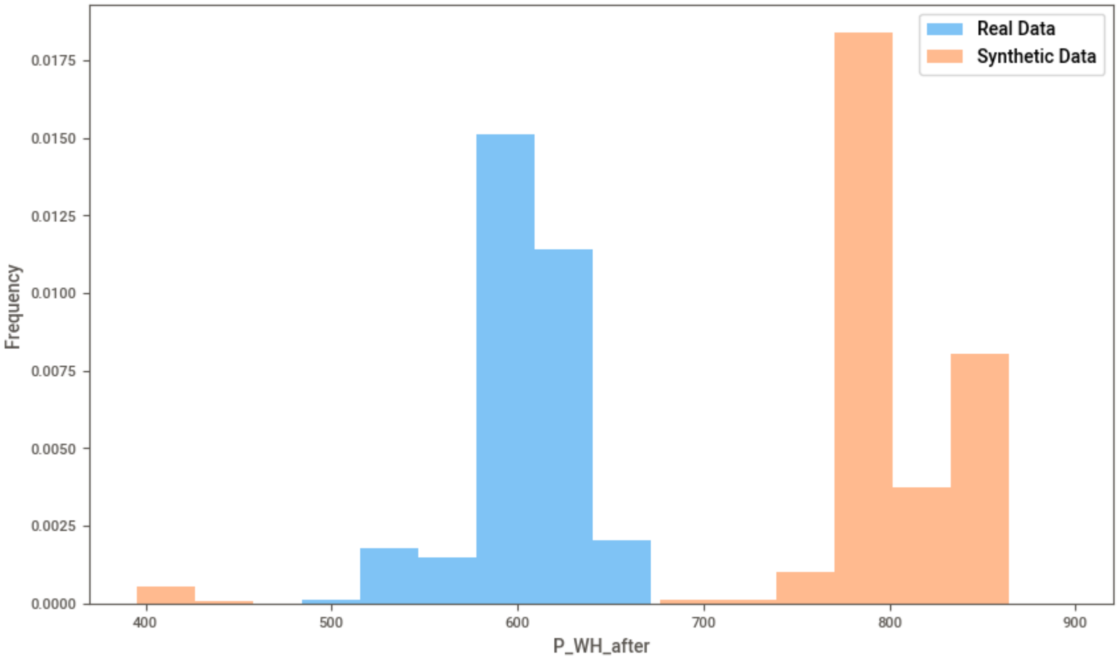


Figure D.4 – Histograms of synthetic (orange) and real (blue) P_WH_before variables with 15 bins.

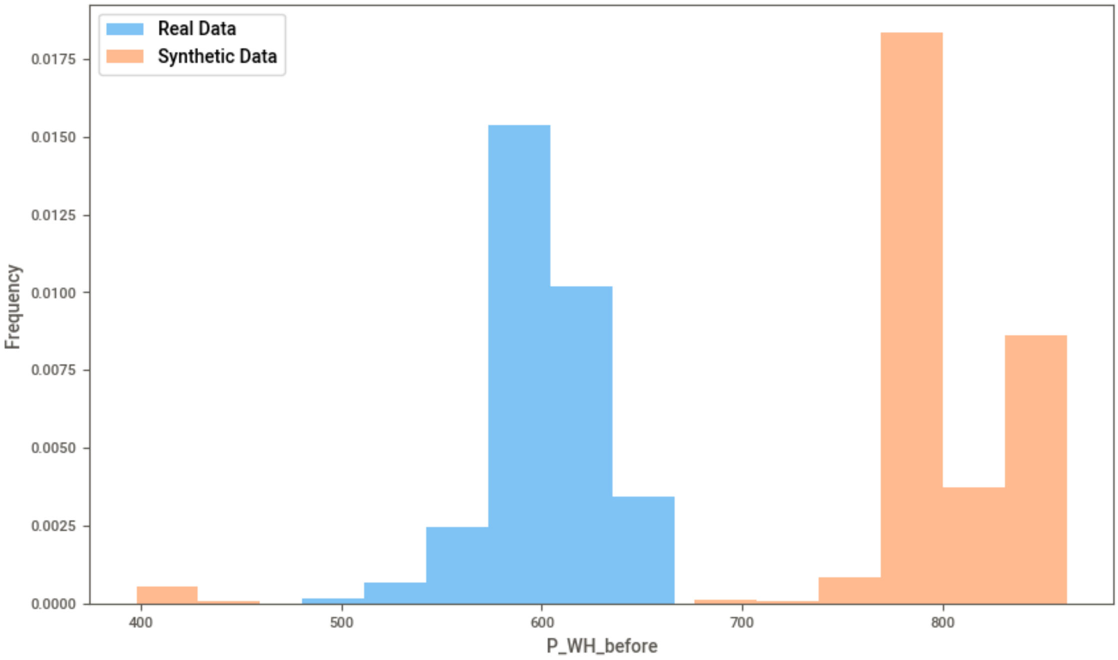


Figure D.5 – Histograms of synthetic (orange) and real (blue) P_WH_delta variables with 15 bins.

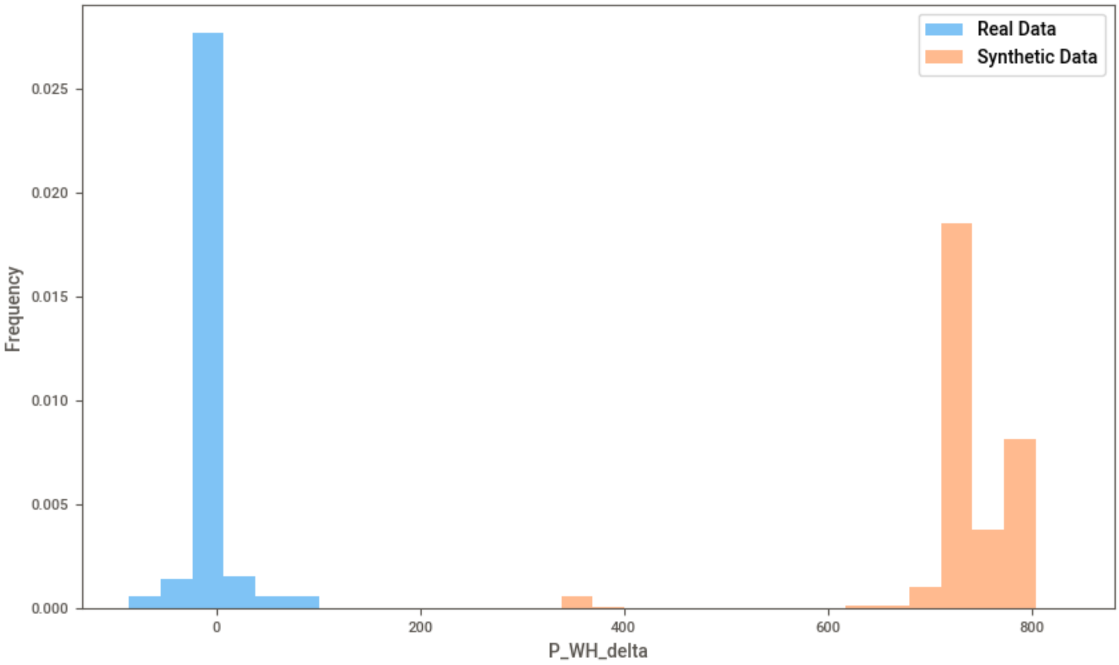


Figure D.6 – Histograms of synthetic (orange) and real (blue) P_Z0_after variables with 15 bins.

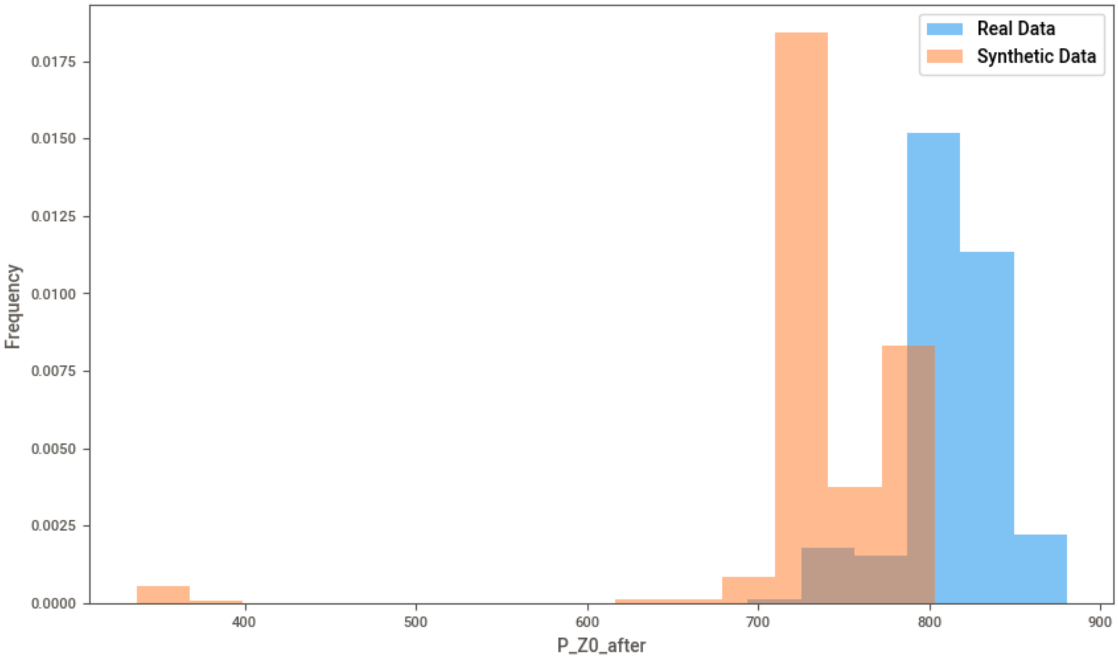


Figure D.7 – Histograms of synthetic (orange) and real (blue) P_Z0_before variables with 15 bins.

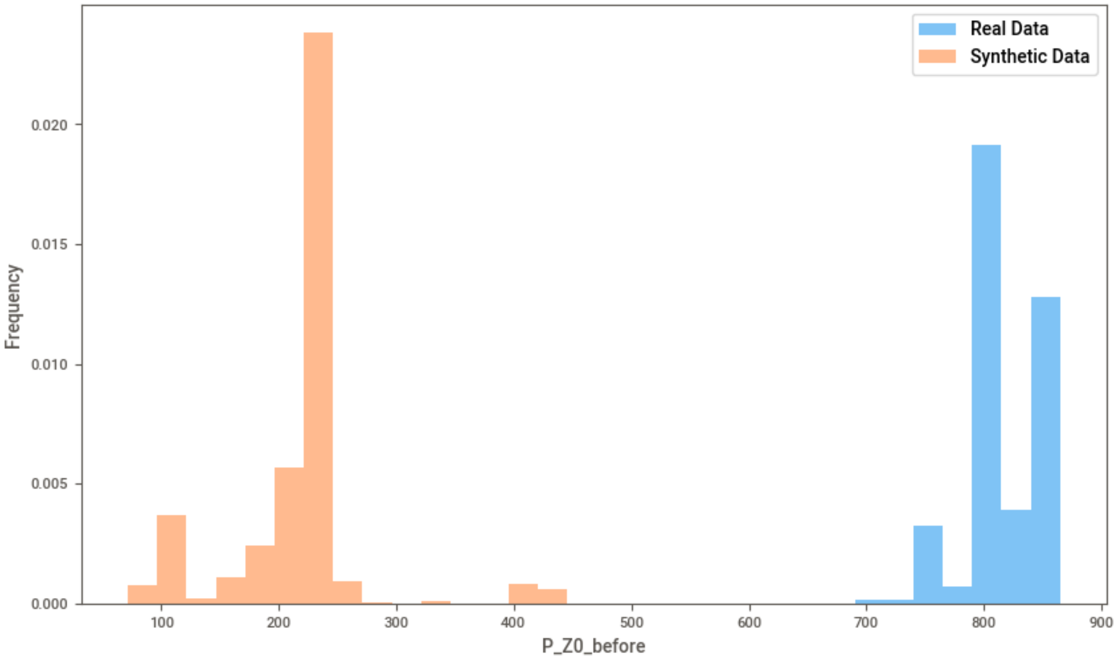


Figure D.8 – Histograms of synthetic (orange) and real (blue) P_Z0_delta variables with 15 bins.

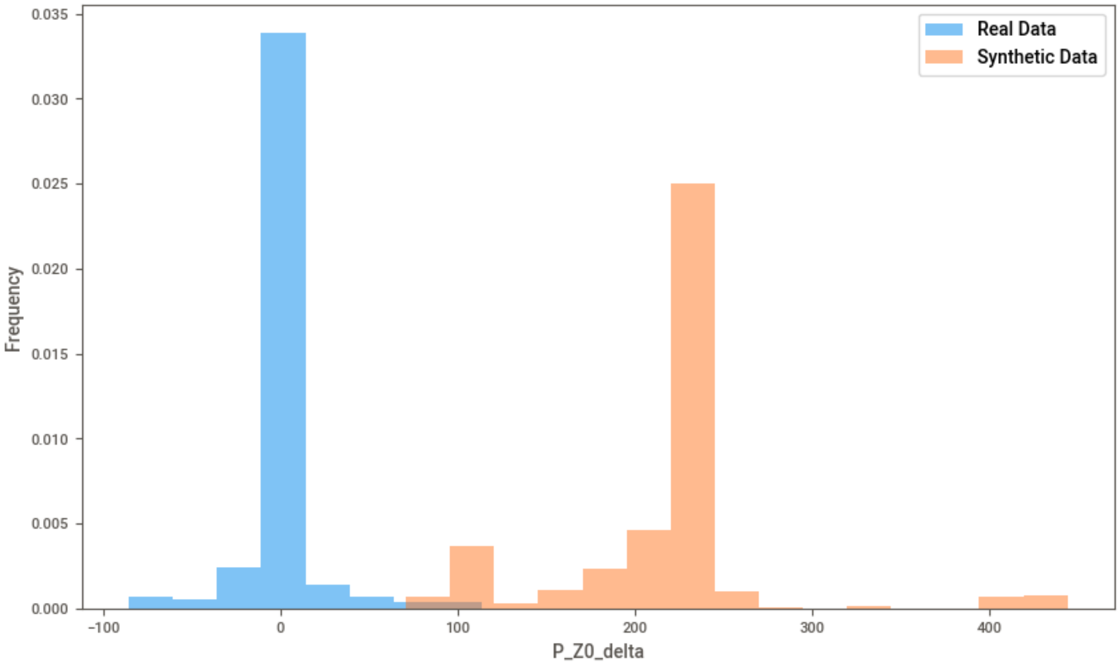


Figure D.9 – Histograms of synthetic (orange) and real (blue) P_Z1_after variables with 15 bins.

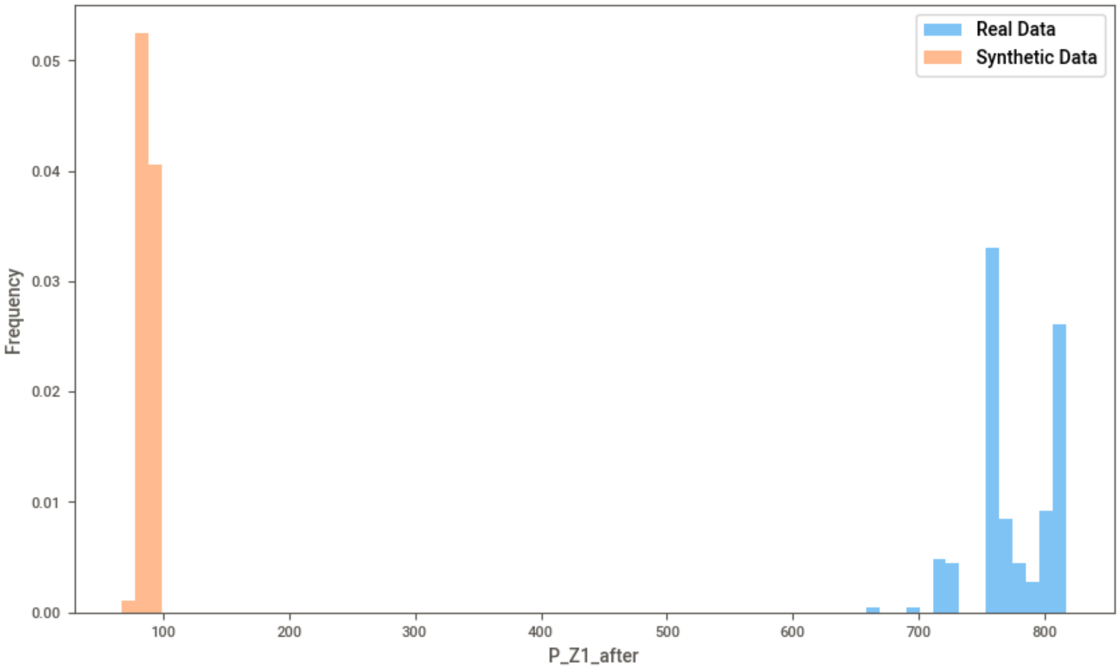


Figure D.10 – Histograms of synthetic (orange) and real (blue) P_Z1_before variables with 15 bins.

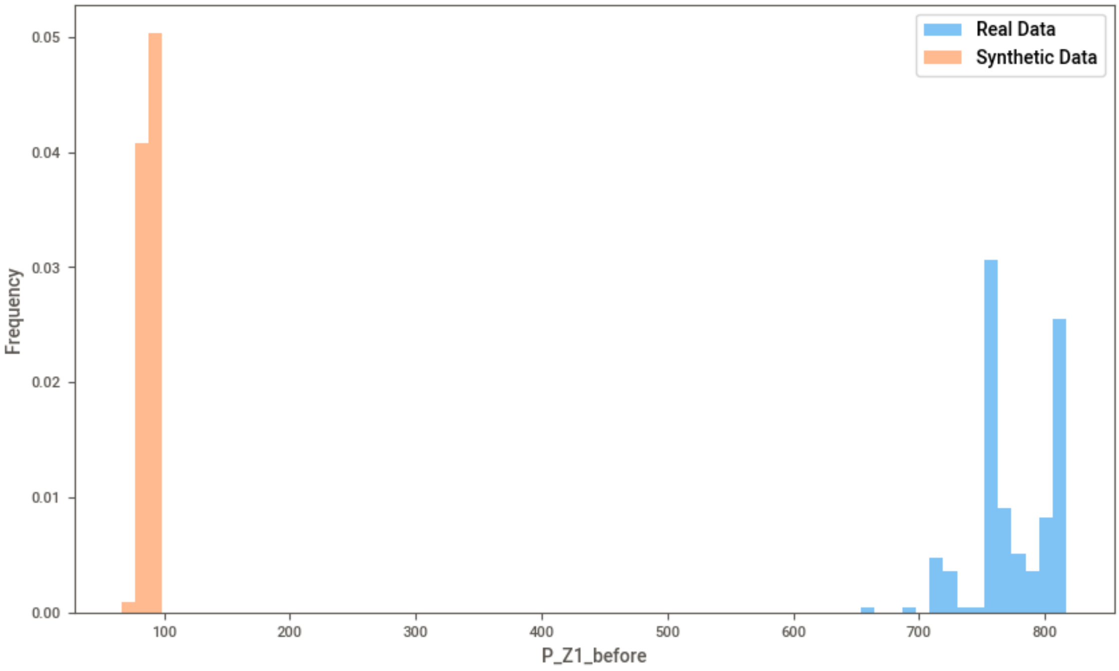


Figure D.11 – Histograms of synthetic (orange) and real (blue) P_Z1_delta variables with 15 bins.

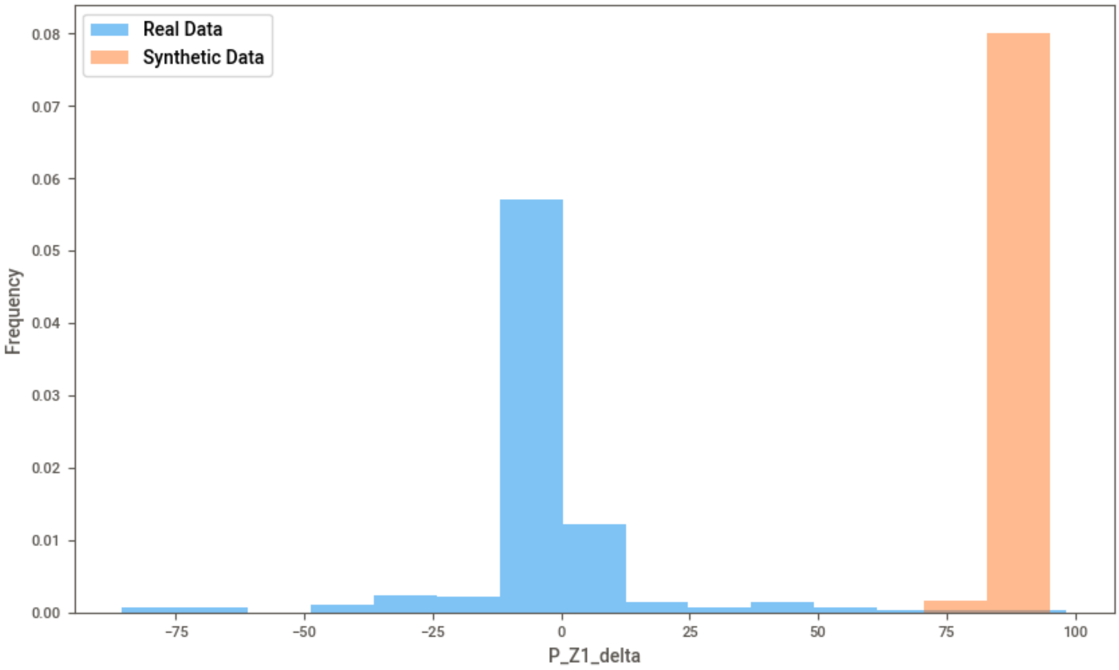


Figure D.12 – Histograms of synthetic (orange) and real (blue) P_bottom_after variables with 15 bins.

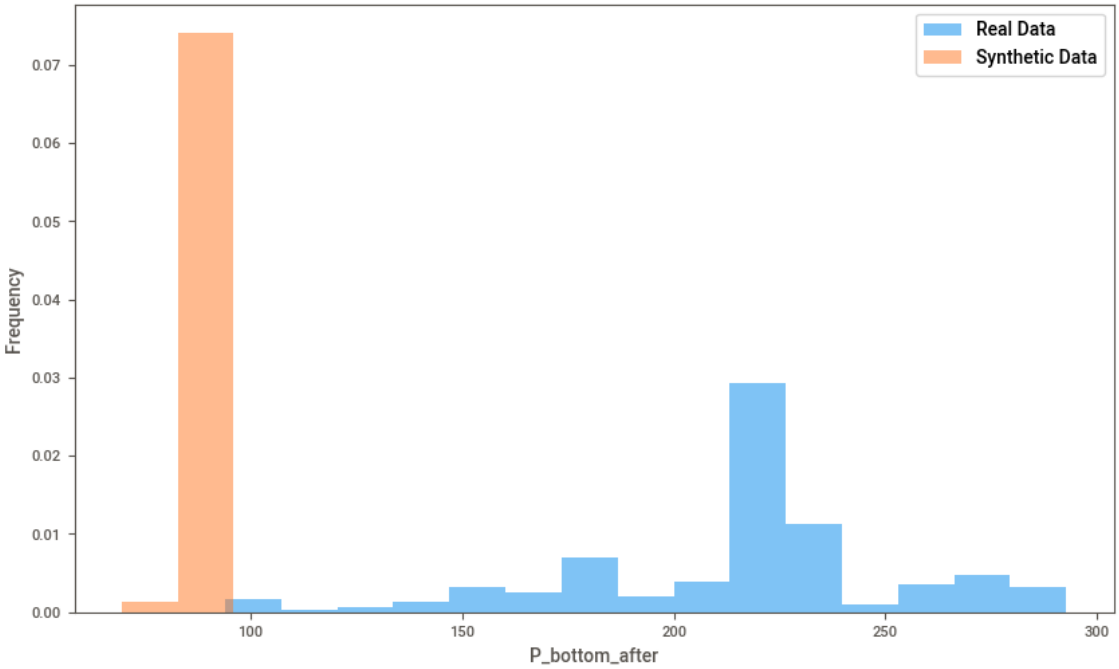


Figure D.13 – Histograms of synthetic (orange) and real (blue) P_bottom_before variables with 15 bins.

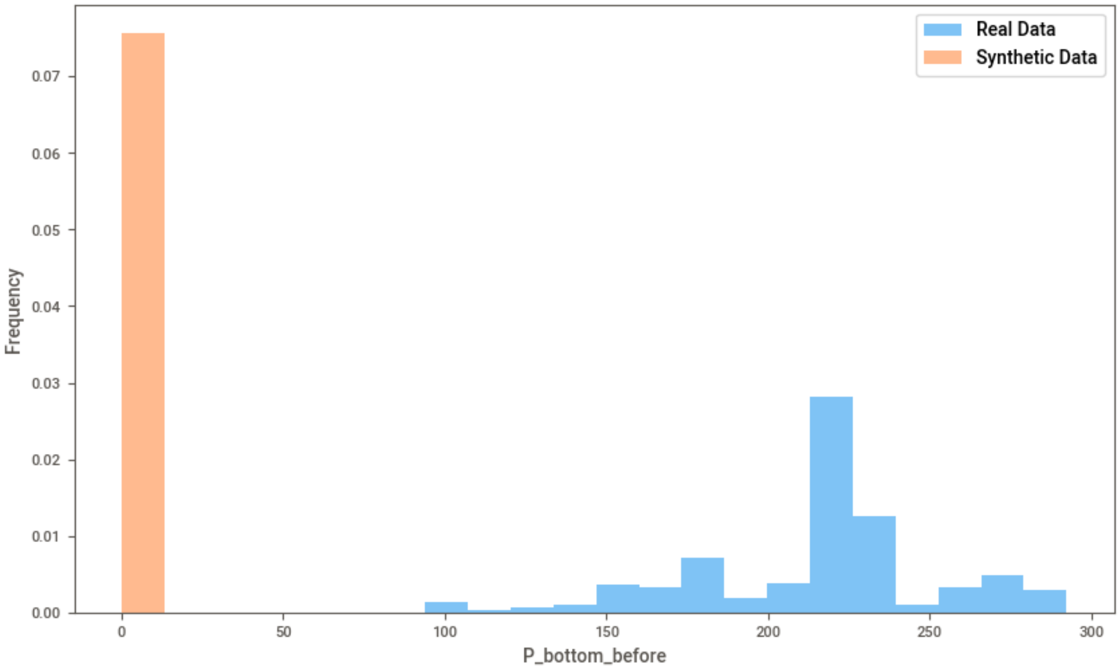


Figure D.14 – Histograms of synthetic (orange) and real (blue) P_bottom_delta variables with 15 bins.

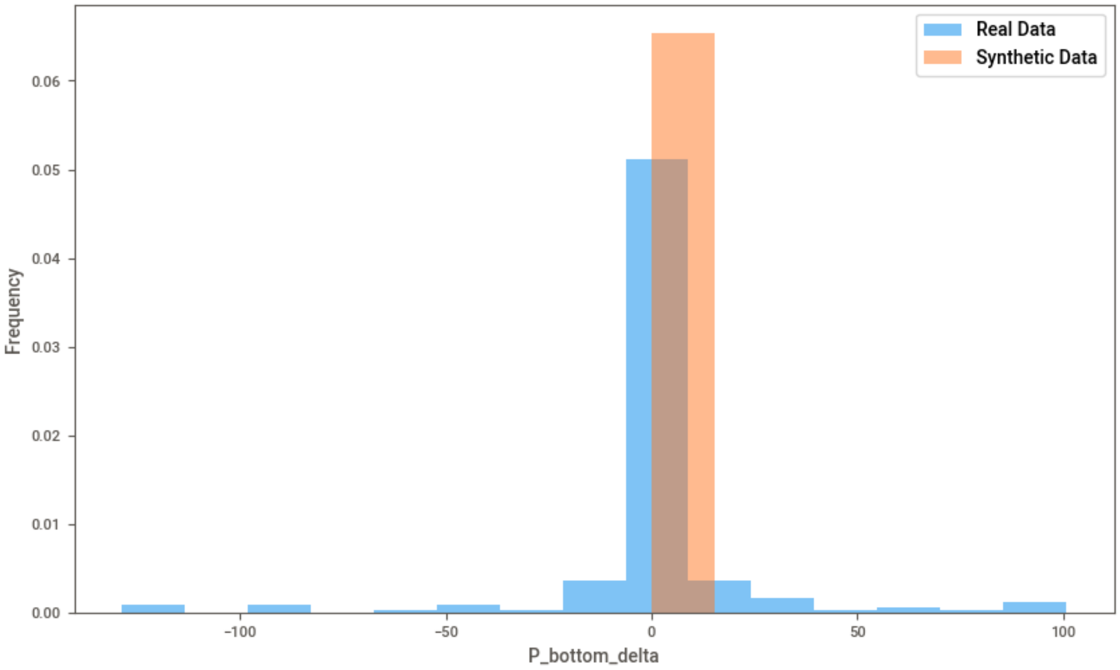


Figure D.15 – Histograms of synthetic (orange) and real (blue) T_WH_after variables with 15 bins.

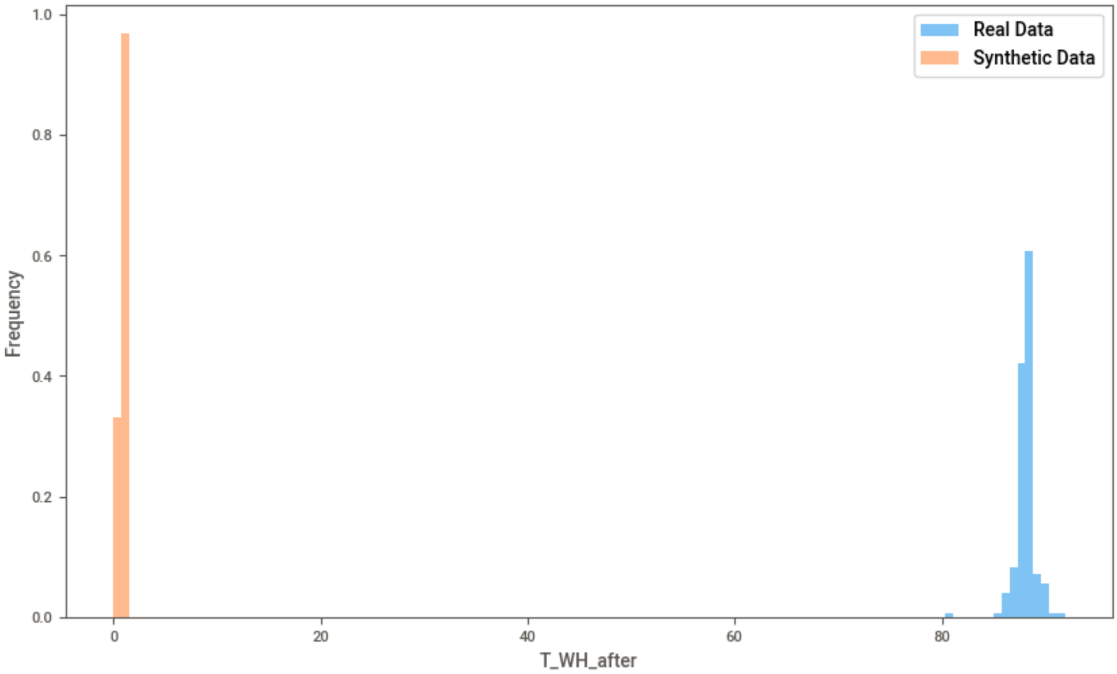


Figure D.16 – Histograms of synthetic (orange) and real (blue) T_WH_before variables with 15 bins.

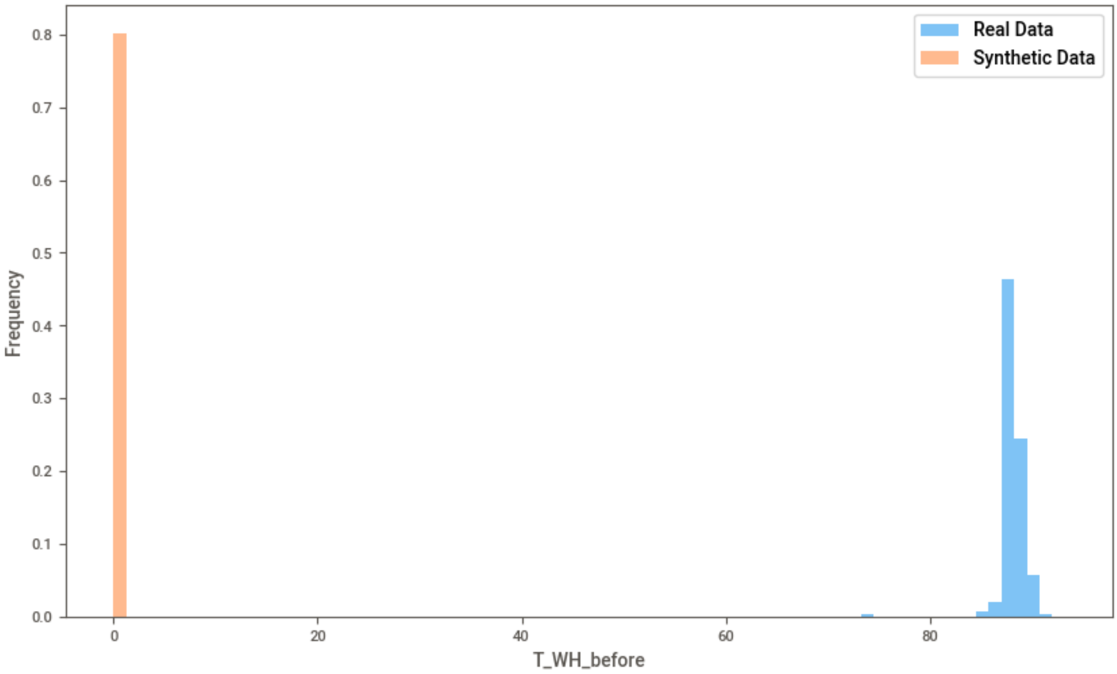


Figure D.17 – Histograms of synthetic (orange) and real (blue) T_WH_delta variables with 15 bins.

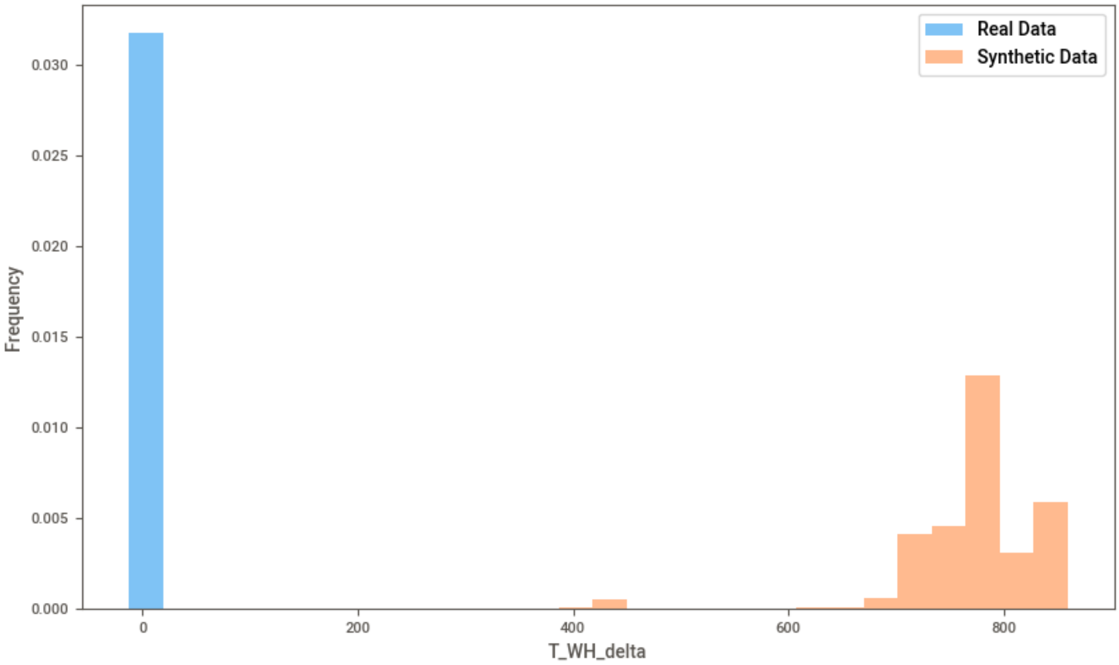


Figure D.18 – Histograms of synthetic (orange) and real (blue) T_Z0_after variables with 15 bins.

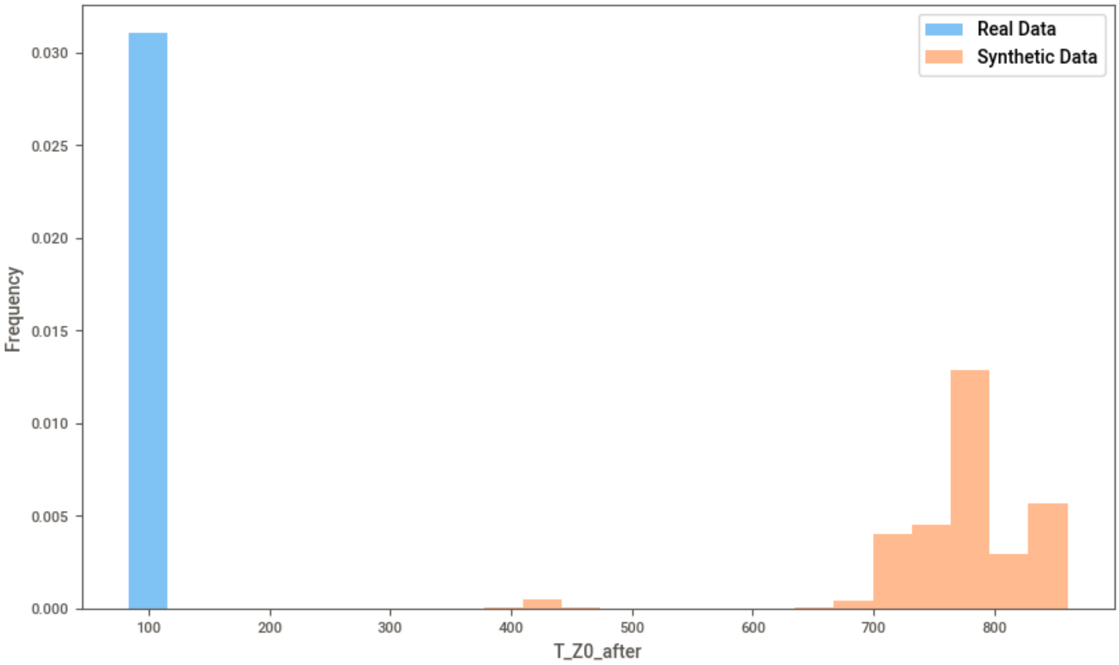


Figure D.19 – Histograms of synthetic (orange) and real (blue) T_Z0_before variables with 15 bins.

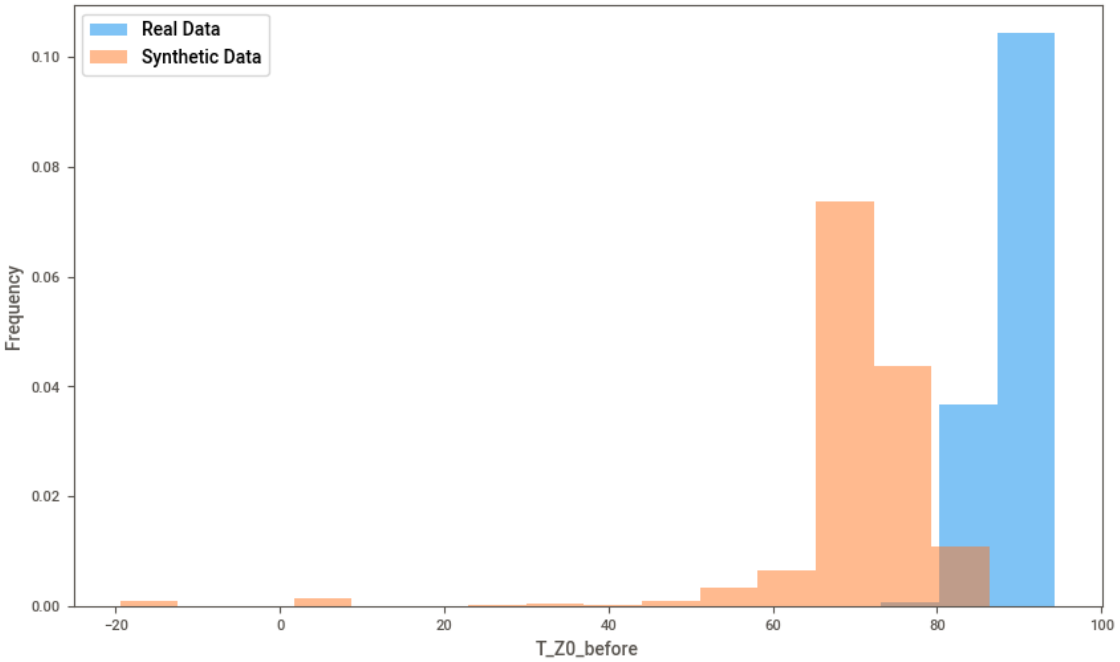


Figure D.20 – Histograms of synthetic (orange) and real (blue) T_Z0_delta variables with 15 bins.

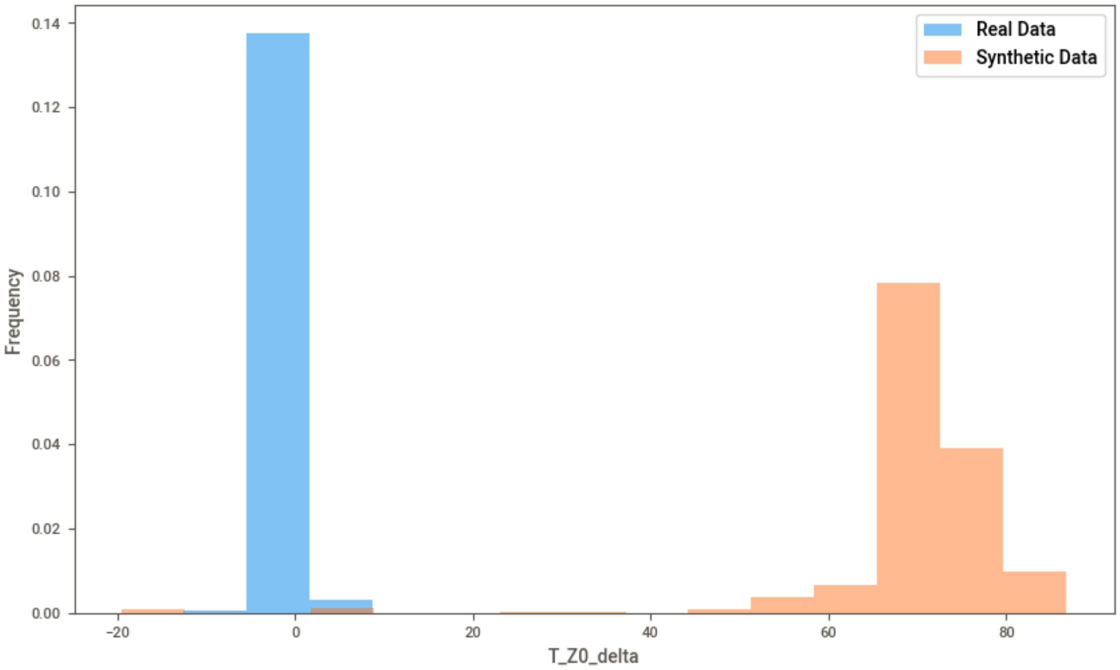


Figure D.21 – Histograms of synthetic (orange) and real (blue) T_Z1_after variables with 15 bins.

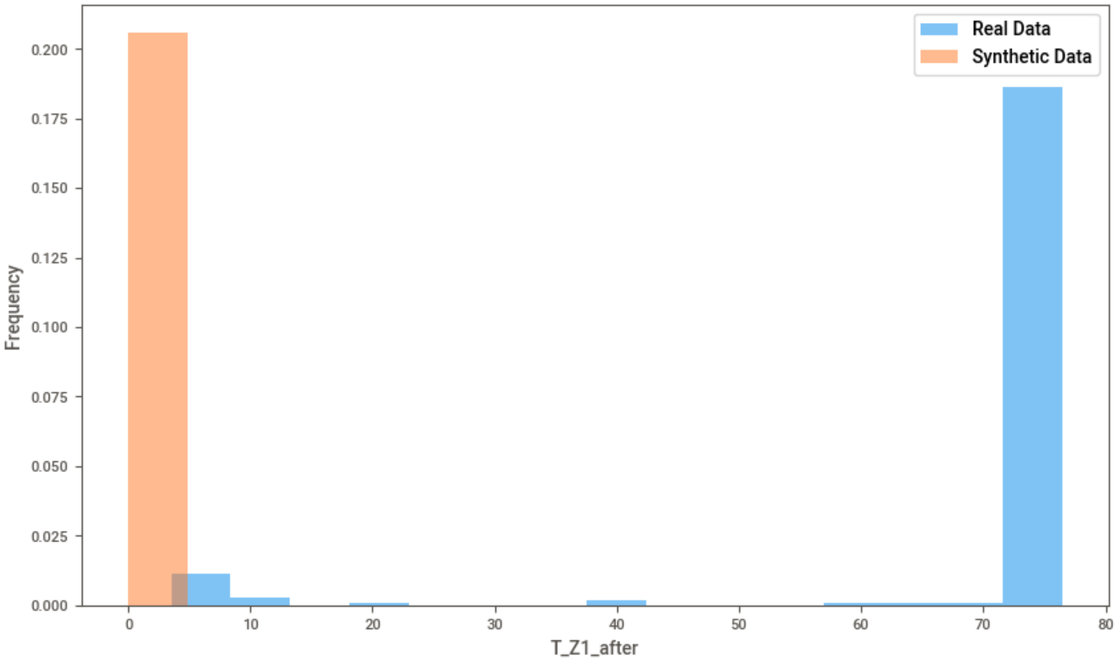


Figure D.22 – Histograms of synthetic (orange) and real (blue) T_Z1_before variables with 15 bins.

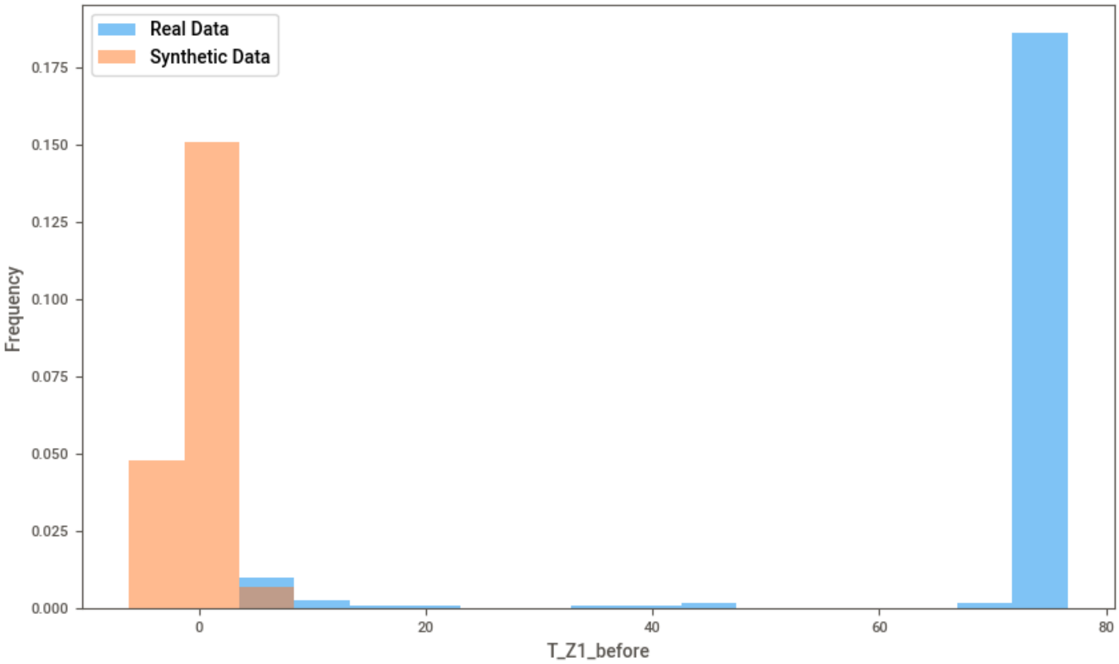
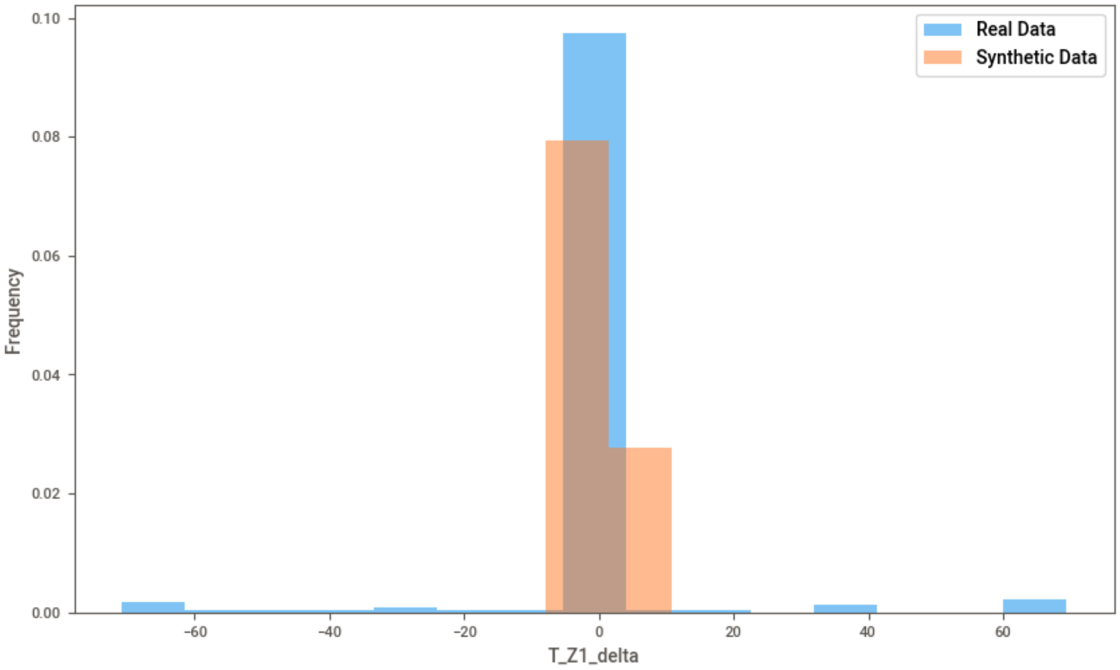


Figure D.23 – Histograms of synthetic (orange) and real (blue) T_Z1_delta variables with 15 bins.



D.1 Descriptive statistics

```
------------------------------------------------------------
                              Real Data      Synthetic Data
------------------------------------------------------------
------------------------------------------------------------
Variable: P_WH_after
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    235            10000
Number of zeroes:             0              0
-------------------------------------------------------
Max value:                    643.70         864.29
Q3:                           633.36         833.11
Avg:                          602.57         795.78
Median:                       596.00         791.05
Q1:                           585.18         782.57
```

```
Min value:                    484.35         394.99
-------------------------------------------------------
Range:                        159.35         469.30
IQR:                          48.18          50.54
STD:                          30.00          58.82
Kurtosis:                     0.09           29.43
Skewness:                     -0.53          -4.89
------------------------------------------------------------
Variable: P_WH_before
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    235            10000
Number of zeroes:             0              0
-------------------------------------------------------
Max value:                    643.70         862.22
Q3:                           633.37         833.11
Avg:                          602.43         795.90
Median:                       596.04         791.18
Q1:                           585.18         782.72
Min value:                    480.55         397.56
-------------------------------------------------------
Range:                        163.15         464.65
IQR:                          48.19          50.40
STD:                          29.84          58.85
Kurtosis:                     0.24           29.47
Skewness:                     -0.54          -4.89
------------------------------------------------------------
Variable: P_WH_delta
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    235            10000
Number of zeroes:             0              0
```

```
-------------------------------------------------------
Max value:                    98.32          804.29
Q3:                           0.04           773.41
Avg:                          -0.13          735.94
Median:                       -0.24          731.13
Q1:                           -0.78          722.71
Min value:                    -85.58         338.43
-------------------------------------------------------
Range:                        183.90         465.86
IQR:                          0.82           50.71
STD:                          18.63          58.83
Kurtosis:                     11.77          29.32
Skewness:                     0.83           -4.87
------------------------------------------------------------
Variable: P_Z0_after
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    235            10000
Number of zeroes:             0              0
-------------------------------------------------------
Max value:                    853.17         803.23
Q3:                           842.83         773.22
Avg:                          812.04         735.94
Median:                       805.48         731.39
Q1:                           794.65         722.45
Min value:                    693.82         336.54
-------------------------------------------------------
Range:                        159.35         466.69
IQR:                          48.18          50.77
STD:                          30.00          58.85
Kurtosis:                     0.09           29.38
Skewness:                     -0.53          -4.88
------------------------------------------------------------
```

```
Variable: P_Z0_before
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    235            10000
Number of zeroes:             0              0
-------------------------------------------------------
Max value:                    853.17         445.72
Q3:                           842.84         228.19
Avg:                          811.91         214.86
Median:                       805.51         224.56
Q1:                           794.65         208.66
Min value:                    690.02         71.86
-------------------------------------------------------
Range:                        163.15         373.86
IQR:                          48.19          19.53
STD:                          29.84          56.42
Kurtosis:                     0.24           5.23
Skewness:                     -0.54          0.96
------------------------------------------------------------
Variable: P_Z0_delta
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    235            10000
Number of zeroes:             0              0
-------------------------------------------------------
Max value:                    98.32          444.20
Q3:                           0.04           228.29
Avg:                          -0.13          214.89
Median:                       -0.24          224.59
Q1:                           -0.78          208.37
Min value:                    -85.58         70.81
-------------------------------------------------------
```

```
Range:                        183.90         373.38
IQR:                          0.82           19.92
STD:                          18.63          56.44
Kurtosis:                     11.77          5.23
Skewness:                     0.83           0.96
------------------------------------------------------------
Variable: P_Z1_after
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    235            10000
Number of zeroes:             0              0
-------------------------------------------------------
Max value:                    817.08         93.89
Q3:                           806.74         89.20
Avg:                          775.95         87.65
Median:                       769.38         88.00
Q1:                           758.56         86.68
Min value:                    657.73         67.07
-------------------------------------------------------
Range:                        159.35         26.82
IQR:                          48.18          2.52
STD:                          30.00          2.60
Kurtosis:                     0.09           11.00
Skewness:                     -0.53          -2.36
------------------------------------------------------------
Variable: P_Z1_before
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    235            10000
Number of zeroes:             0              0
-------------------------------------------------------
Max value:                    817.08         94.81
```

```
Q3:                           806.75         89.16
Avg:                          775.81         87.64
Median:                       769.42         88.01
Q1:                           758.55         86.66
Min value:                    653.93         66.04
-------------------------------------------------------
Range:                        163.15         28.77
IQR:                          48.19          2.50
STD:                          29.84          2.63
Kurtosis:                     0.24           11.87
Skewness:                     -0.54          -2.44
------------------------------------------------------------
Variable: P_Z1_delta
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    235            10000
Number of zeroes:             0              0
-------------------------------------------------------
Max value:                    98.32          93.42
Q3:                           0.04           89.35
Avg:                          -0.13          88.02
Median:                       -0.24          88.27
Q1:                           -0.78          87.07
Min value:                    -85.58         70.53
-------------------------------------------------------
Range:                        183.90         22.89
IQR:                          0.82           2.29
STD:                          18.63          2.10
Kurtosis:                     11.77          9.07
Skewness:                     0.83           -1.86
------------------------------------------------------------
Variable: P_bottom_after
-------------------------------------------------------
```

```
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    220            10000
Number of zeroes:             0              0
-------------------------------------------------------
Max value:                    292.75         94.05
Q3:                           228.61         89.32
Avg:                          215.79         88.04
Median:                       221.82         88.31
Q1:                           201.10         87.12
Min value:                    93.94          69.47
-------------------------------------------------------
Range:                        198.81         24.59
IQR:                          27.50          2.20
STD:                          38.19          2.09
Kurtosis:                     1.13           9.94
Skewness:                     -0.67          -1.96
------------------------------------------------------------
Variable: P_bottom_before
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    217            2
Number of zeroes:             0              2466
-------------------------------------------------------
Max value:                    292.37         1.00
Q3:                           228.64         1.00
Avg:                          215.49         0.75
Median:                       221.85         1.00
Q1:                           200.07         1.00
Min value:                    93.94          0.00
-------------------------------------------------------
Range:                        198.43         1.00
IQR:                          28.57          0.00
```

```
STD:                          37.74          0.43
Kurtosis:                     0.98           -0.62
Skewness:                     -0.59          -1.18
------------------------------------------------------------
Variable: P_bottom_delta
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    206            2
Number of zeroes:             0              2466
-------------------------------------------------------
Max value:                    100.79         1.00
Q3:                           1.43           1.00
Avg:                          0.43           0.75
Median:                       -0.27          1.00
Q1:                           -1.09          1.00
Min value:                    -128.83        0.00
-------------------------------------------------------
Range:                        229.62         1.00
IQR:                          2.52           0.00
STD:                          25.39          0.43
Kurtosis:                     11.15          -0.62
Skewness:                     -0.64          -1.18
------------------------------------------------------------
Variable: T_WH_after
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    195            2
Number of zeroes:             0              2553
-------------------------------------------------------
Max value:                    91.93          1.00
Q3:                           88.35          1.00
Avg:                          88.05          0.74
```

```
Median:                       88.21          1.00
Q1:                           87.41          0.00
Min value:                    80.39          0.00
-------------------------------------------------------
Range:                        11.54          1.00
IQR:                          0.94           1.00
STD:                          0.96           0.44
Kurtosis:                     17.79          -0.74
Skewness:                     -1.87          -1.12
------------------------------------------------------------
Variable: T_WH_before
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    202            2
Number of zeroes:             0              2554
-------------------------------------------------------
Max value:                    92.00          1.00
Q3:                           88.37          1.00
Avg:                          88.01          0.74
Median:                       88.20          1.00
Q1:                           87.40          0.00
Min value:                    73.28          0.00
-------------------------------------------------------
Range:                        18.72          1.00
IQR:                          0.97           1.00
STD:                          1.28           0.44
Kurtosis:                     74.52          -0.74
Skewness:                     -6.41          -1.12
------------------------------------------------------------
Variable: T_WH_delta
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    189            10000
Number of zeroes:             5              0
-------------------------------------------------------
Max value:                    4.21           858.96
Q3:                           0.02           801.36
Avg:                          -0.08          774.17
Median:                       0.00           785.31
Q1:                           -0.03          745.77
Min value:                    -12.45         386.39
-------------------------------------------------------
Range:                        16.65          472.58
IQR:                          0.05           55.59
STD:                          0.99           62.80
Kurtosis:                     103.83         14.71
Skewness:                     -8.01          -3.06
------------------------------------------------------------
Variable: T_Z0_after
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    227            10000
Number of zeroes:             0              0
-------------------------------------------------------
Max value:                    91.96          860.53
Q3:                           88.69          800.88
Avg:                          88.12          774.14
Median:                       88.27          785.33
Q1:                           87.30          746.30
Min value:                    83.50          377.31
-------------------------------------------------------
Range:                        8.46           483.22
IQR:                          1.39           54.58
STD:                          0.95           62.72
Kurtosis:                     2.76           14.72
```

```
Skewness:                     -0.34          -3.06
------------------------------------------------------------
Variable: T_Z0_before
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    225            10000
Number of zeroes:             0              0
-------------------------------------------------------
Max value:                    92.05          86.40
Q3:                           88.70          73.73
Avg:                          88.08          70.15
Median:                       88.27          71.48
Q1:                           87.30          69.14
Min value:                    73.19          -19.46
-------------------------------------------------------
Range:                        18.86          105.86
IQR:                          1.40           4.60
STD:                          1.32           10.95
Kurtosis:                     69.18          34.21
Skewness:                     -6.09          -5.17
------------------------------------------------------------
Variable: T_Z0_delta
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    225            10000
Number of zeroes:             0              0
-------------------------------------------------------
Max value:                    3.94           86.77
Q3:                           0.03           73.69
Avg:                          -0.02          70.15
Median:                       0.01           71.50
Q1:                           -0.01          69.22
```

```
Min value:                    -12.57         -19.54
-------------------------------------------------------
Range:                        16.50          106.31
IQR:                          0.03           4.47
STD:                          0.96           10.99
Kurtosis:                     127.72         34.35
Skewness:                     -9.36          -5.19
------------------------------------------------------------
Variable: T_Z1_after
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    229            2
Number of zeroes:             0              4952
-------------------------------------------------------
Max value:                    76.43          1.00
Q3:                           75.52          1.00
Avg:                          69.58          0.50
Median:                       75.27          1.00
Q1:                           74.04          0.00
Min value:                    3.53           0.00
-------------------------------------------------------
Range:                        72.90          1.00
IQR:                          1.48           1.00
STD:                          18.17          0.50
Kurtosis:                     8.43           -2.00
Skewness:                     -3.18          -0.02
------------------------------------------------------------
Variable: T_Z1_before
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    229            10000
Number of zeroes:             0              0
```

```
-------------------------------------------------------
Max value:                    76.61          7.54
Q3:                           75.54          1.27
Avg:                          69.79          0.02
Median:                       75.27          0.00
Q1:                           74.08          -1.23
Min value:                    3.52           -6.21
-------------------------------------------------------
Range:                        73.09          13.75
IQR:                          1.47           2.50
STD:                          17.59          1.88
Kurtosis:                     8.87           -0.01
Skewness:                     -3.23          0.02
------------------------------------------------------------
Variable: T_Z1_delta
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    230            10000
Number of zeroes:             3              0
-------------------------------------------------------
Max value:                    69.40          9.63
Q3:                           0.06           1.48
Avg:                          0.25           -0.01
Median:                       -0.00          -0.02
Q1:                           -0.04          -1.51
Min value:                    -70.84         -7.93
-------------------------------------------------------
Range:                        140.25         17.56
IQR:                          0.09           3.00
STD:                          15.77          2.22
Kurtosis:                     12.96          -0.02
Skewness:                     0.32           0.02
------------------------------------------------------------
```

```
Variable: action_result
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    2              10000
Number of zeroes:             23             0
-------------------------------------------------------
Max value:                    1.00           30.37
Q3:                           1.00           6.07
Avg:                          0.90           -0.00
Median:                       1.00           0.06
Q1:                           1.00           -6.00
Min value:                    0.00           -33.29
-------------------------------------------------------
Range:                        1.00           63.65
IQR:                          0.00           12.07
STD:                          0.30           8.95
Kurtosis:                     5.47           -0.04
Skewness:                     -2.72          -0.01
------------------------------------------------------------
Variable: bottom_icv_status_after
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    2              10000
Number of zeroes:             16             0
-------------------------------------------------------
Max value:                    1.00           36.26
Q3:                           1.00           6.05
Avg:                          0.93           0.12
Median:                       1.00           -0.03
Q1:                           1.00           -5.83
Min value:                    0.00           -35.91
-------------------------------------------------------
```

```
Range:                        1.00           72.17
IQR:                          0.00           11.88
STD:                          0.25           8.89
Kurtosis:                     10.00          0.06
Skewness:                     -3.45          0.03
------------------------------------------------------------
Variable: bottom_icv_status_before
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    2              10000
Number of zeroes:             13             0
-------------------------------------------------------
Max value:                    1.00           18.90
Q3:                           1.00           3.18
Avg:                          0.94           0.03
Median:                       1.00           0.08
Q1:                           1.00           -3.23
Min value:                    0.00           -19.55
-------------------------------------------------------
Range:                        1.00           38.46
IQR:                          0.00           6.41
STD:                          0.23           4.78
Kurtosis:                     13.45          0.01
Skewness:                     -3.92          -0.00
------------------------------------------------------------
Variable: command_type_0
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    2              10000
Number of zeroes:             22             0
-------------------------------------------------------
Max value:                    1.00           123.21
```

```
Q3:                           1.00           18.36
Avg:                          0.91           -0.03
Median:                       1.00           0.01
Q1:                           1.00           -18.62
Min value:                    0.00           -124.45
-------------------------------------------------------
Range:                        1.00           247.66
IQR:                          0.00           36.97
STD:                          0.29           48.72
Kurtosis:                     5.94           -0.34
Skewness:                     -2.81          -0.00
------------------------------------------------------------
Variable: command_type_1
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    2              10000
Number of zeroes:             7              0
-------------------------------------------------------
Max value:                    1.00           24.67
Q3:                           1.00           2.00
Avg:                          0.97           0.00
Median:                       1.00           0.01
Q1:                           1.00           -1.98
Min value:                    0.00           -24.72
-------------------------------------------------------
Range:                        1.00           49.39
IQR:                          0.00           3.98
STD:                          0.17           3.64
Kurtosis:                     29.25          7.44
Skewness:                     -5.57          -0.06
------------------------------------------------------------
Variable: command_type_2
-------------------------------------------------------
```

```
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    2              2
Number of zeroes:             233            8113
-------------------------------------------------------
Max value:                    1.00           1.00
Q3:                           0.00           0.00
Avg:                          0.01           0.19
Median:                       0.00           0.00
Q1:                           0.00           0.00
Min value:                    0.00           0.00
-------------------------------------------------------
Range:                        1.00           1.00
IQR:                          0.00           0.00
STD:                          0.09           0.39
Kurtosis:                     114.97         0.53
Skewness:                     10.77          1.59
------------------------------------------------------------
Variable: command_type_3
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    2              2
Number of zeroes:             226            8151
-------------------------------------------------------
Max value:                    1.00           1.00
Q3:                           0.00           0.00
Avg:                          0.04           0.18
Median:                       0.00           0.00
Q1:                           0.00           0.00
Min value:                    0.00           0.00
-------------------------------------------------------
Range:                        1.00           1.00
IQR:                          0.00           0.00
```

```
STD:                          0.19           0.39
Kurtosis:                     21.63          0.64
Skewness:                     4.84           1.62
------------------------------------------------------------
Variable: command_type_4
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    2              2
Number of zeroes:             34             7486
-------------------------------------------------------
Max value:                    1.00           1.00
Q3:                           1.00           1.00
Avg:                          0.86           0.25
Median:                       1.00           0.00
Q1:                           1.00           0.00
Min value:                    0.00           0.00
-------------------------------------------------------
Range:                        1.00           1.00
IQR:                          0.00           1.00
STD:                          0.35           0.43
Kurtosis:                     2.15           -0.69
Skewness:                     -2.03          1.15
------------------------------------------------------------
Variable: top_icv_status_after
-------------------------------------------------------
Number of observations:       235            10000
Number of missing values:     0              0
Number of distinct values:    2              2
Number of zeroes:             224            8098
-------------------------------------------------------
Max value:                    1.00           1.00
Q3:                           0.00           0.00
Avg:                          0.05           0.19
```

```
Median:                       0.00          0.00
Q1:                           0.00          0.00
Min value:                    0.00          0.00
-------------------------------------------------------
Range:                        1.00          1.00
IQR:                          0.00          0.00
STD:                          0.21          0.39
Kurtosis:                     16.79         0.49
Skewness:                     4.32          1.58
------------------------------------------------------------
Variable: top_icv_status_before
-------------------------------------------------------
Number of observations:       235           10000
Number of missing values:     0             0
Number of distinct values:    2             2
Number of zeroes:             223           8152
-------------------------------------------------------
Max value:                    1.00          1.00
Q3:                           0.00          0.00
Avg:                          0.05          0.18
Median:                       0.00          0.00
Q1:                           0.00          0.00
Min value:                    0.00          0.00
-------------------------------------------------------
Range:                        1.00          1.00
IQR:                          0.00          0.00
STD:                          0.22          0.39
Kurtosis:                     14.98         0.64
Skewness:                     4.11          1.62
```

APPENDIX E — MACHINE LEARNING MODELING AND VALIDATION DETAILS

Below we present more details about the machine learning modeling with the synthetic data. Machine learning algorithms and encoders used are instances of sckit-learn classes (SCIKIT-LEARN, 2022). All the codes and results are available in our repository (BDI-UFRGS, 2023).

E.1 Simulation Results Metadata

```
Int64Index: 10000 entries, 0 to 9999
Data columns (total 20 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   P_Z0_before                10000 non-null  float64
 1   P_Z0_after                 10000 non-null  float64
 2   P_Z1_before                10000 non-null  float64
 3   P_Z1_after                 10000 non-null  float64
 4   P_WH_before                10000 non-null  float64
 5   P_WH_after                 10000 non-null  float64
 6   T_Z0_before                10000 non-null  float64
 7   T_Z0_after                 10000 non-null  float64
 8   T_Z1_before                10000 non-null  float64
 9   T_Z1_after                 10000 non-null  float64
 10  top_icv_status_before      10000 non-null  bool
 11  top_icv_status_after       10000 non-null  bool
 12  bottom_icv_status_before   10000 non-null  bool
 13  bottom_icv_status_after    10000 non-null  bool
 14  P_bottom_before            10000 non-null  float64
 15  P_bottom_after             10000 non-null  float64
 16  T_WH_before                10000 non-null  float64
 17  T_WH_after                 10000 non-null  float64
 18  command_type                 10000 non-null  object
 19  failure                      10000 non-null  bool
dtypes: bool(5), float64(14), object(1)
memory usage: 1.3+ MB
```

E.2 Data Engineering

For each measure (float64 variales) we calculated the difference after/before any action and store it in a column of our dataset.

```
float_vars = set(
    [col.split('_')[0]+'_'+col.split('_')[1]
        for col in list(sim_results.columns) if
        sim_results[col].dtype == 'float64'])

for name in float_vars:
    cols = [col for col in list(sim_results.columns)
        if col.startswith(name)]

    sim_results[name+'_delta'] =
    sim_results[cols[1]] - sim_results[cols[0]]

sim_results.info()
```

```
Int64Index: 10000 entries, 0 to 9999
Data columns (total 27 columns):
 #   Column                       Non-Null Count  Dtype
---  ------                       --------------  -----
 0   P_Z0_before                  10000 non-null  float64
 1   P_Z0_after                   10000 non-null  float64
 2   P_Z1_before                  10000 non-null  float64
 3   P_Z1_after                   10000 non-null  float64
 4   P_WH_before               10000 non-null  float64
 5   P_WH_after                10000 non-null  float64
 6   T_Z0_before               10000 non-null  float64
 7   T_Z0_after                10000 non-null  float64
 8   T_Z1_before               10000 non-null  float64
 9   T_Z1_after                10000 non-null  float64
 10  top_icv_status_before     10000 non-null  bool
 11  top_icv_status_after      10000 non-null  bool
 12  bottom_icv_status_before  10000 non-null  bool
 13  bottom_icv_status_after   10000 non-null  bool
 14  P_bottom_before           10000 non-null  float64
 15  P_bottom_after            10000 non-null  float64
 16  T_WH_before               10000 non-null  float64
 17  T_WH_after                10000 non-null  float64
 18  command_type              10000 non-null  object
 19  action_result             10000 non-null  object
 20  P_Z1_delta                10000 non-null  float64
 21  P_Z0_delta                10000 non-null  float64
 22  T_WH_delta                10000 non-null  float64
 23  T_Z1_delta                10000 non-null  float64
 24  P_bottom_delta            10000 non-null  float64
 25  P_WH_delta                10000 non-null  float64
 26  T_Z0_delta                10000 non-null  float64
dtypes: bool(4), float64(21), object(2)
memory usage: 1.9+ MB
```

E.3 Encoding for Categorical Variables

Scikit-learn algorithms expect numeric features. So, categorical features need to be encoded before inserted into the model.

```
dic_encoders = {’action_result’: LabelEncoder(),
               ’command_type’: OneHotEncoder(*, categories=’auto’,
               drop=None, sparse=False, sparse_output=True,
               dtype=<class ’numpy.float64’>,
               handle_unknown=’error’, min_frequency=None,
               max_categories=None)}

dic_encoders[’action_result’].fit(sim_results[’action_result’])
new_col = dic_encoders[’action_result’].transform(
    sim_results[’action_result’])

sim_results[’action_result’] = new_col

new_cols = dic_encoders[’command_type’].fit_transform(
sim_results[’command_type’].values.reshape(-1, 1)).T

for k in range(new_cols.shape[0]):
    sim_results[f’command_type_{k}’] = new_cols[k]
sim_results.drop(columns=[’command_type’], inplace=True)

# Metadata of simulation results after the encoding stage.
sim_results.info()
Int64Index: 10000 entries, 0 to 9999
Data columns (total 27 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   P_Z0_before               10000 non-null  float64
 1   P_Z0_after                10000 non-null  float64
 2   P_Z1_before               10000 non-null  float64
 3   P_Z1_after                10000 non-null  float64
 4   P_WH_before               10000 non-null  float64
 5   P_WH_after                10000 non-null  float64
 6   T_Z0_before               10000 non-null  float64
 7   T_Z0_after                10000 non-null  float64
 8   T_Z1_before               10000 non-null  float64
 9   T_Z1_after                10000 non-null  float64
 10  top_icv_status_before     10000 non-null  bool
 11  top_icv_status_after      10000 non-null  bool
 12  bottom_icv_status_before  10000 non-null  bool
 13  bottom_icv_status_after   10000 non-null  bool
 14  P_bottom_before           10000 non-null  float64
 15  P_bottom_after            10000 non-null  float64
 16  T_WH_before               10000 non-null  float64
 17  T_WH_after                10000 non-null  float64
 18  command_type              10000 non-null  float64
 19  action_result             10000 non-null  int32
 20  P_Z1_delta                10000 non-null  float64
 21  P_Z0_delta                10000 non-null  float64
 22  T_WH_delta                10000 non-null  float64
 23  T_Z1_delta                10000 non-null  float64
 24  P_bottom_delta            10000 non-null  float64
 25  P_WH_delta                10000 non-null  float64
 26  T_Z0_delta                10000 non-null  float64
```

E.4 Machine Learning Models Parameters

```
{'Logistic_Reg_clf':
    LogisticRegression(penalty='l2', *,
    dual=False, tol=0.0001, C=1.0,
    fit_intercept=True, intercept_scaling=1, class_weight=None,
    random_state=42, solver='lbfgs', max_iter=100,
    multi_class='auto', verbose=0, warm_start=False,
    n_jobs=None, l1_ratio=None),

 'DT_clf': DecisionTreeClassifier(*, criterion='gini',
    splitter='best', max_depth=None, min_samples_split=2,
    min_samples_leaf=1, min_weight_fraction_leaf=0.0,
    max_features=None, random_state=42,
    max_leaf_nodes=None, min_impurity_decrease=0.0,
    class_weight=None, ccp_alpha=0.0),

’RF_clf’: RandomForestClassifier(n_estimators=100, *,
    criterion=’gini’, max_depth=None,
    min_samples_split=2, min_samples_leaf=1,
    min_weight_fraction_leaf=0.0,
    max_features=’sqrt’, max_leaf_nodes=None,
    min_impurity_decrease=0.0, bootstrap=True,
    oob_score=False, n_jobs=None, random_state=42,
    verbose=0, warm_start=False, class_weight=None,
    ccp_alpha=0.0, max_samples=None),

’KNN_clf’: KNeighborsClassifier(n_neighbors=5, *,
    weights=’uniform’, algorithm=’auto’, leaf_size=30,
    p=2, metric=’minkowski’, metric_params=None,
    n_jobs=None),

’NB_clf’: GaussianNB(*, priors=None, var_smoothing=1e-09),

’SV_clf’: SVC(*, C=1.0, kernel=’rbf’,
    degree=3, gamma=’scale’, coef0=0.0,
    shrinking=True, probability=False, tol=0.001,
    cache_size=200, class_weight=None, verbose=False,
    max_iter=-1, decision_function_shape=’ovr’,
    break_ties=False, random_state=42),

’Adaboost_clf’: AdaBoostClassifier(estimator=None, *,
     n_estimators=50, learning_rate=1.0,
    algorithm=’SAMME.R’, random_state=42,
     base_estimator=’deprecated’),

’XGBoost_RF_clf’: XGBRFClassifier(base_score=0.5,
    booster=’gbtree’, callbacks=None,
    colsample_bylevel=1, colsample_bytree=1,
    early_stopping_rounds=None, enable_categorical=False,
    eval_metric=None, gamma=0, gpu_id=-1,
    grow_policy='depthwise', importance_type=None,
    interaction_constraints='', max_bin=256,
    max_cat_to_onehot=4, max_delta_step=0, max_depth=6,
    max_leaves=0, min_child_weight=1, missing=nan,
    monotone_constraints='()', n_estimators=100, n_jobs=0,
    num_parallel_tree=100, objective='binary:logistic',
    predictor='auto', random_state=42, reg_alpha=0,
    sampling_method='uniform', scale_pos_weight=1, ...),

 'MLP_clf': MLPClassifier(alpha=1e-05,
    hidden_layer_sizes=(5, 2), random_state=42,
    solver='sgd')}
```

E.5 Classification Metrics Reports

```
Evaluating model Logistic_Reg_clf...
Making predictions to the test set...
Evaluating the metrics:
              precision    recall  f1-score   support

           0      0.000     0.000     0.000        23
           1      0.902     1.000     0.949       212

    accuracy                          0.902       235
   macro avg      0.451     0.500     0.474       235
weighted avg      0.814     0.902     0.856       235

Evaluating model DT_clf...
Making predictions to the test set...
Evaluating the metrics:
```

```
              precision    recall  f1-score   support

           0      0.655     0.826     0.731        23
           1      0.981     0.953     0.967       212

    accuracy                          0.940       235
   macro avg      0.818     0.889     0.849       235
weighted avg      0.949     0.940     0.943       235

Evaluating model RF_clf...
Making predictions to the test set...
Evaluating the metrics:
              precision    recall  f1-score   support

           0      0.667     0.783     0.720        23
           1      0.976     0.958     0.967       212

    accuracy                          0.940       235
   macro avg      0.821     0.870     0.843       235
weighted avg      0.946     0.940     0.943       235

Evaluating model KNN_clf...
Making predictions to the test set...
Evaluating the metrics:
              precision    recall  f1-score   support

           0      0.095     0.957     0.173        23
           1      0.667     0.009     0.019       212

    accuracy                          0.102       235
   macro avg      0.381     0.483     0.096       235
weighted avg      0.611     0.102     0.034       235

Evaluating model NB_clf...
```

```
Making predictions to the test set...
Evaluating the metrics:
              precision    recall  f1-score   support

           0      0.000     0.000     0.000        23
           1      0.902     1.000     0.949       212

    accuracy                          0.902       235
   macro avg      0.451     0.500     0.474       235
weighted avg      0.814     0.902     0.856       235

Evaluating model SV_clf...
Making predictions to the test set...
Evaluating the metrics:
              precision    recall  f1-score   support

           0      0.000     0.000     0.000        23
           1      0.902     1.000     0.949       212

    accuracy                          0.902       235
   macro avg      0.451     0.500     0.474       235
weighted avg      0.814     0.902     0.856       235

Evaluating model Adaboost_clf...
Making predictions to the test set...
Evaluating the metrics:
              precision    recall  f1-score   support

           0      0.676     1.000     0.807        23
           1      1.000     0.948     0.973       212

    accuracy                          0.953       235
   macro avg      0.838     0.974     0.890       235
weighted avg      0.968     0.953     0.957       235
```

```
Evaluating model XGBoost_RF_clf...
Making predictions to the test set...
Evaluating the metrics:
              precision    recall  f1-score   support

           0      0.676     1.000     0.807        23
           1      1.000     0.948     0.973       212

    accuracy                          0.953       235
   macro avg      0.838     0.974     0.890       235
weighted avg      0.968     0.953     0.957       235

Evaluating model MLP_clf...
Making predictions to the test set...
Evaluating the metrics:
              precision    recall  f1-score   support

           0      0.058     0.478     0.103        23
           1      0.727     0.151     0.250       212

    accuracy                          0.183       235
   macro avg      0.392     0.315     0.176       235
weighted avg      0.662     0.183     0.236       235
```