European Journal of Molecular & Clinical Medicine
ISSN 2515-8260 Volume 7, Issue 06, 2020

Leveraging Digital Twin Technology in the Healthcare Industry – A Machine Learning Based Approach

Aashish Bende[1], Dr.Saikat Gochhait[2]

Symbiosis Institute of Digital and Telecom Management, constituent of Symbiosis International (Deemed University)

Abstract: This paper deals with the concept of digital twin technology and leveraging the same in the healthcare domain. Digital twin technology is adding value to the healthcare industry by personalizing the diagnosis and therapy selection procedure. Finite Element Analysis (Finite Element Analysis, 2001) is a simulation method used to create a digital replica or a digital instance of the human organs such as heart, kidney, or brain. IoT devices or sensors such as implantable cardioverter defibrillatoror heart pacemakers collect the medical data of patients, which is analyzed to create a virtual instance using simulation software. The virtual instance is continuously updated and is used in generating reports for further diagnosis.

Keywords: Machine Learning, Jupyter Notebook, Python, Decision Tree, Simulation, Digital Twin, IoT, HIPPA, Predictive Analytics, Prescriptive Analytics, and Finite Element Analysis

Introduction –

A Digital Twin is a virtual replica of physical objects or digital representation of devices, which are created through the process of simulation using finite element analysis. NASA used this technology for space exploration in the year 2002. The technology was later adopted in different sectors such as manufacturing, oil & gas, telecommunication, and aerospace. The advent of Industry 4.0, AI, analytics software, big data analytics, and machine learning algorithms have eased the process of data collection and analysis (Gochhait and Rimal,2019). Software such as MATLAB, Simio Simulation, Abaqus, and many others are used for creating a virtual copy of the physical entity. The process involves three major steps, first, in the pre-processor stage, the image of the real object is created based on the input parameters, the second step divides the virtual image into different parts and adopts an agile-based approach to analyze each part of the virtual replica, and the last step involves assembling of the image to gather insights from the analysis. Simulation is the process of designing a model of a real system (could be machine or human organs) to conduct experiments on the model for understanding behavior or evaluating various strategies for the operation of the system. These techniques are most widely used for operations research and management science. The finite element method (FEM) or finite element analysis is the most common method used in the simulation. This method helps in predicting how a product will react to real-world forces. FEM can predict the future behavior of the real object by analyzing the virtual image. Abaqus is a common software for finite element analysis. With the advent of technologies such as AI, IoT, and other sensor-based devices have bolstered demand analytics. It has various end-users such as automotive, healthcare, aerospace, defense, and manufacturing. Digital Twin has a wide variety of applications in the health care domain. It is the future of personalized health care, which provides every individual the right type of

care at the right time. The paper is focused on leveraging the digital twin technology in the health care domain to improve personalized diagnosis. Machine learning algorithm is used for classifying the types of heart diseases.

Literature Review–

The current paper focuses on creating the digital twin of human heart using simulation software predict the chances of heart disease. The model prepared is original and explains the process flow of data transfer from implantable devices to devising a strategy for diagnosis. The reference for creating the model is taken from Philips Healthcare Model of digital twin (Houten, 2018) model; machine learning algorithm is added to increase its analytics capabilities. Machine-learning algorithm (decision tree) is used to predict the maximum probability that affects the target variable (target variable explains the parameters causing heart diseases). The target variable is used by the doctors to assess the accuracy of the model and then the diagnosis is designed accordingly. Several papers have been published in the area of digital twin technology such as (Madni, Madni, and Lucero, 2019), which leverages digital twin technology in model-based system engineering and include it as an integral part of MBSE methodology and experimentation test beds. The paper also presents the benefits of integrating digital twins with systems simulation and IoT devices. (Cunbo, Jianhua, Hui, Xiaoyu, Shaoli, and Gang, 2017) discuss the architecture, connotations, and trends in digital twin products, the paper has proposed architecture of the product digital twin, which is based on a systemic analysis of its connotation. (Fei Tao, Jiangfeng Cheng, Qinglin Qi, Meng Zhang, He Zhang &Fangyuan Sui)have leveraged big data analytics to derive product design, manufacturing, and services to be more efficient. The paper is focused on highlighting the limitations of the virtual replica and what problems occur while performing the analysis. (Zhuang, Liu, and Xiong, 2018)the research focuses on a digital twin based framework in smart production management. It also suggests a measure to control the approach for complex products. Some core techniques embodied in the proposed framework are real-time management, organization, and acquisition of physical assembly data related to the shop floor. (Boschert, and Rosen, 2016) covers the simulation aspect of the digital twin technology and prepare a model to support design tasks or validate systems. (Schleich, Anwer, Mathieu, and Wartzack, 2017) paper uses Skin Model Shapes and proposes a comprehensive reference model. (Tao, Sui, Liu, Qi, Zhang, Song, Guo, Lu, and Nee, 2018) has prepared the framework of digital twin-driven product design (DTPD). (Sivalingam, Sepulveda, Spring,and Davis, 2018) have proposed a methodology, which predicts the remaining useful life of an offshore wind turbine power converter using a digital twin framework. (Schroeder, Steinmetz, Pereira, and Espindola, 2016) used automation ML model attributes related to the Digital twin. (Brosinsky, Westermann, and Krebs, 2018) has enhanced the application in power control system centers by adopting digital twin, and (Uhlemann, Lehmann, and Steinhilper, 2017)have done the research for automated data acquisition and selection using digital twin technology. It explains the concept of microprocessor controlled pacemakers to record, store, and transmit information regarding the patient (Sanders, Martin, Frumin, and Goldberg, 1984).

Research Methodology –

The paper describes the concept of a digital twin using a framework. The data is obtained from devices implanted in the patient's heart such as pacemaker and implantable cardioverter-defibrillator to gather information related to various parameters such as chest pain, resting blood pressure, etc. The parameter is used by simulation software to create a virtual instance of the human heart. The simulation technique will create a biophysical model of the patient, which looks and behaves similarly to the real one. Data analytics techniques

such as big data analytics and cloud computing are used to access real-time data; this connects the digital twin of the patient with similar cases and selects the one with the optimal computed result (Gochhait, Shou & Fazalbhoy, 2020). Further machine learning algorithms such as decision trees can be used to categorize patient's conditions based on the target variable accuracy. The chosen ideal scenarios form the basis of real-time intervention guidance. During the procedure, all actions and unforeseen situations are processed in real-time into the patient's digital twin, which is then used for following the best strategy on the real object. Once the model is created using the above-mentioned technologies and processes, the digital twin of the heart can be linked to physical creation for real-time monitoring and detecting future hearth diseases using predictive and prescriptive analytics. The accuracy achieved during the procedure is 79%; the decision tree divides the parameters based on the accuracy. The decision tree algorithm is chosen for its low complexity and as it is easy to be implemented in wider scenarios compared to other algorithms.

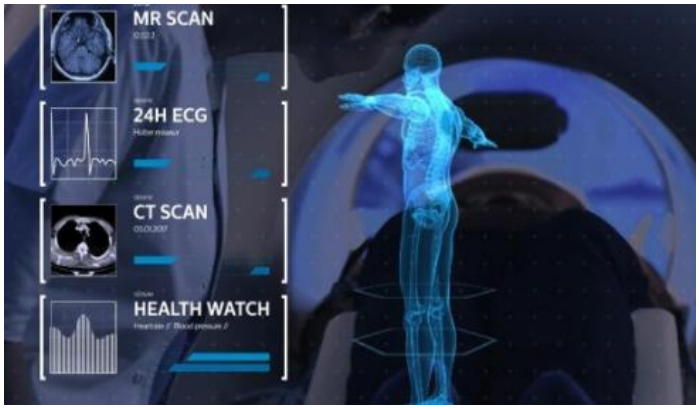
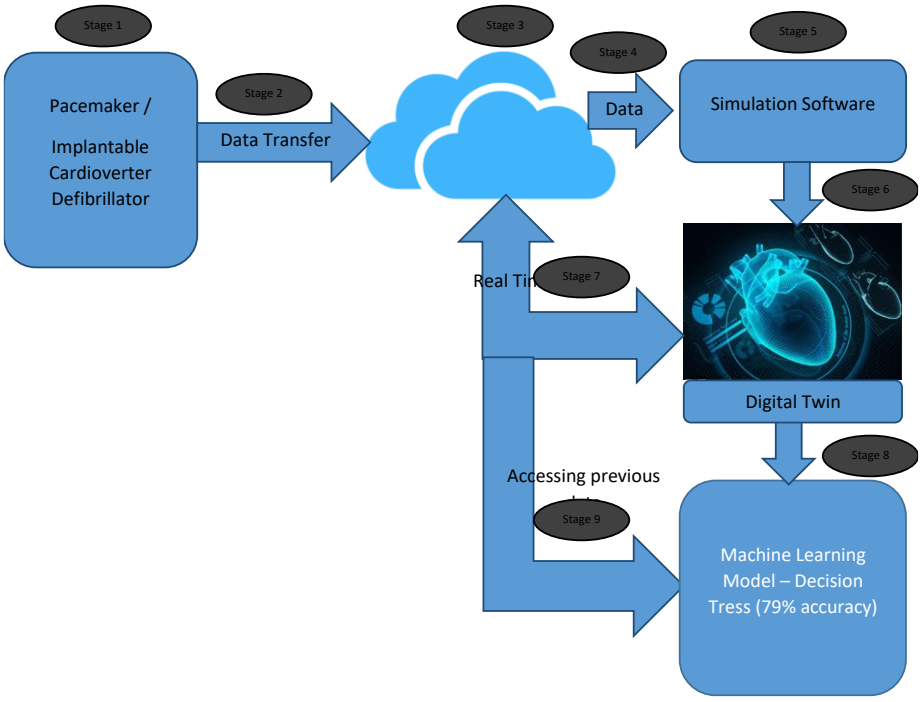


Fig. Biophysical Model – Simulation

Digital Twin Model and Framework



Different stages in the process flow–

Stage 1) – The pacemaker and implantable cardioverter defibrillator implanted in the patient's heart to gather data related to heart conditions.

Stage 2) Data is transferred to the cloud. The data gathered include parameters such as blood pressure, chest pain, fasting blood sugar, and others.

Stage 3) Cloud stores the data and analysis it using simulation software to create virtual twin. SimScale is a cloud-based simulation software used for creating a virtual instance.

Stage 4) Data is transferred from cloud to simulation software. (This step is not required if Cloud-based simulation software are used).

Stage 5) The simulation software inputs the parameter received and performs finite element analysis to create a digital twin.

Stage 6) Digital twin behaves and reacts similarly to the real heart.

Stage 7) Digital twins can access real-time data from cloud and any changes in the parameters will be updated in the virtual instance.

Stage 8) Use of Machine Learning algorithm - decision tree to devise the strategy based on the risk parameters and deviation from the standards to detect the possibility of the disease.

Stage 9) The algorithm can also access previous health data of the patients and can devise the diagnosis strategy accordingly, which is first performed in the virtual heart to reduce the error rate.

Testing and Analysis using Decision Tree Algorithm in Python -

Data is taken from Kaggleand coding is done in Python using Jupyter Notebook.

Importing Libraries –

##Importing all the necessary libraries –

```
In [1]:
import pandas as pd
import matplotlib.pyplot as plt
from matplotlib import rcParams
from matplotlib.cm import rainbow
%matplotlib inline
import warnings
warnings.filterwarnings('ignore')
```

```
In [2]:
from sklearn.model_selection import train_test_split from sklearn.preprocessing import Stan
dardScaler
```

1) numpy and pandas are used for data manipulation and processing.
2) matlplotlib for data visualization
3) rcParams of matplotlib are used for coloring and styling plots
4) sklearn library is used for implementing machine learning algorithm

##Importing decision tree classifier –

```
In [3]:
fromsklearn.treeimportDecisionTreeClassifier
```

Importing Data –

##After importing required libraries, we will import the dataset; the data is taken from kaggle. It is stored by the name of dataset.csv

##For using the dataset, we have used pd.read_csv('dataset.csv')

```
In [4]:
dataset=pd.read_csv('dataset.csv')
```

Data Information and Cleaning –

##This step is performed to validate the data before processing

```
In [4]:
dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 303 entries, 0 to 302
Data columns (total 14 columns):
age         303 non-null int64
sex         303 non-null int64
cp          303 non-null int64
trestbps    303 non-null int64
chol        303 non-null int64
fbs         303 non-null int64
restecg     303 non-null int64
thalach     303 non-null int64
exang       303 non-null int64
oldpeak     303 non-null float64
slope       303 non-null int64
ca          303 non-null int64
thal        303 non-null int64
target      303 non-null int64
dtypes: float64(1), int64(13)
memory usage: 33.2 KB
```

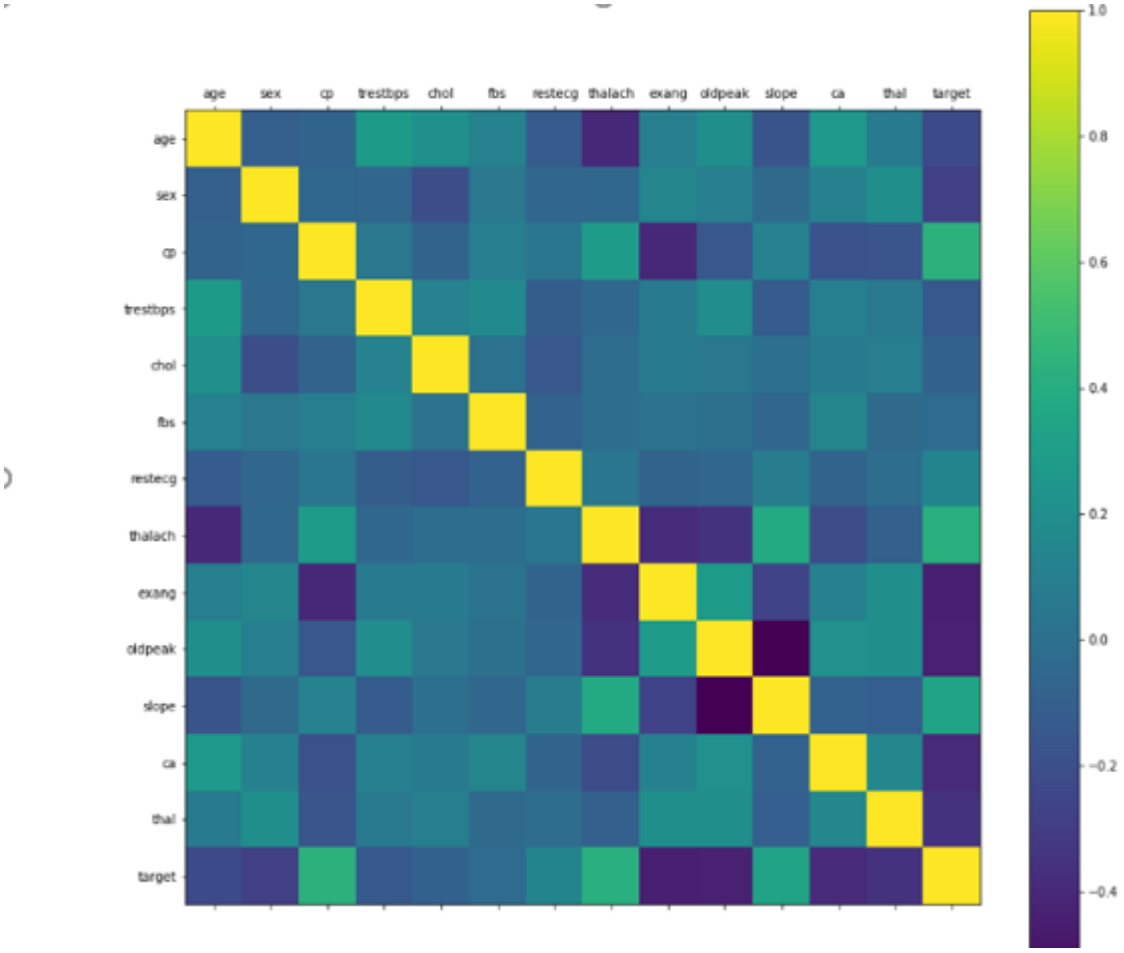
The data has total 303 rows with no missing values. In addition, it has 13 features and 1 target value that needs to be find out.

Data Visualization–

##Before final processing, data needs to be visualized to understand the parameters

```
In [5]:
rcParams['figure.figsize']=20,14
plt.matshow(dataset.corr())
plt.yticks(np.arange(dataset.shape[1]),dataset.columns)
plt.xticks(np.arange(dataset.shape[1]),dataset.columns)
plt.colorbar()
```

Output -



The output is a correlation matrix, which indicates that a few features have positive correlation with the target value and other have negative.

Check Size of Target Variable –

```
In [6]:

rcParams['figure.figsize']=20,14
plt.matshow(dataset.corr())
plt.yticks(np.arange(dataset.shape[1]),dataset.columns)
plt.xticks(np.arange(dataset.shape[1]),dataset.columns)
plt.colorbar()
```

Out[6]:

Text(0.5, 1.0, 'Count of each Target Class')

It is observed that the target classes are of different size but are good enough to continue processing the data.

Data Processing –

##Before implementing the machine learning algorithms, some categorical variables need to be convertedinto dummy variables.

```
In [7]:

dataset=pd.get_dummies(dataset,columns=['cp', 'sex','ca', 'restecg','exang','slope','fbs', 'thal'])
```

##After creating dummy variables, data needs to be scaled. StandardScaler form sklearncan be used for the same

Machine Learning –

##Before importing ML algorithm decision tree, the data is spiltinto training and testing. train_test_split is used for training and testing.

```
In [8]:
y=dataset['target']
X=dataset.drop(['target'],axis=1)
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.33,random_state=0)
```

Decision Tree –

##Decision tree algorithm is implemented to find the feature with best accuracy.

```
In [8]:
dt_scores=[]
foriinrange(1,len(X.columns)+1):
```

```
dt_classifier=DecisionTreeClassifier(max_features=i,random_state=0)
dt_classifier.fit(X_train,y_train)
dt_scores.append(dt_classifier.score(X_test,y_test))
```

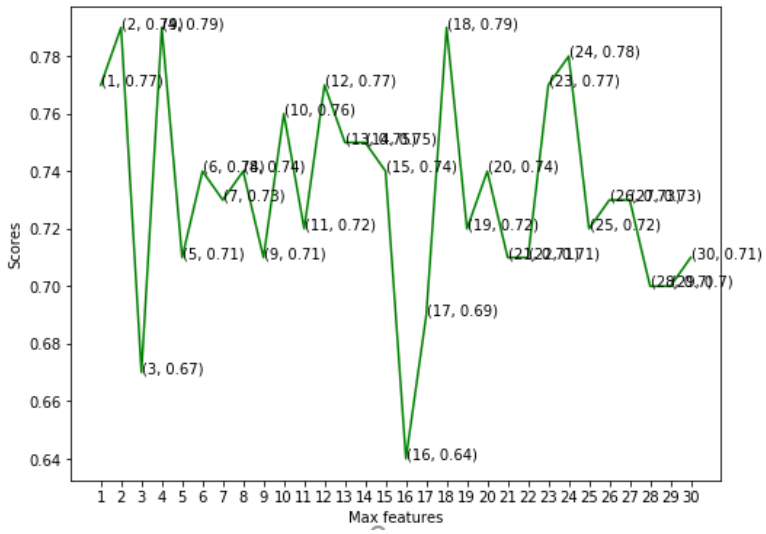
##After selecting maximum features (1 to 30) for splitting the data, the scores of each needs to be find out.

```
In [9]:
plt.plot([i for i in range(1, len(X.columns) + 1)], dt_scores, color =
'green')
for i in range(1, len(X.columns) + 1):
    plt.text(i, dt_scores[i-1], (i, dt_scores[i-1]))
plt.xticks([i for i in range(1, len(X.columns) + 1)])
plt.xlabel('Max features')
plt.ylabel('Scores')
plt.title(' Scores of Decision Tree Classifier for different number of
maximum features)
```

#Output

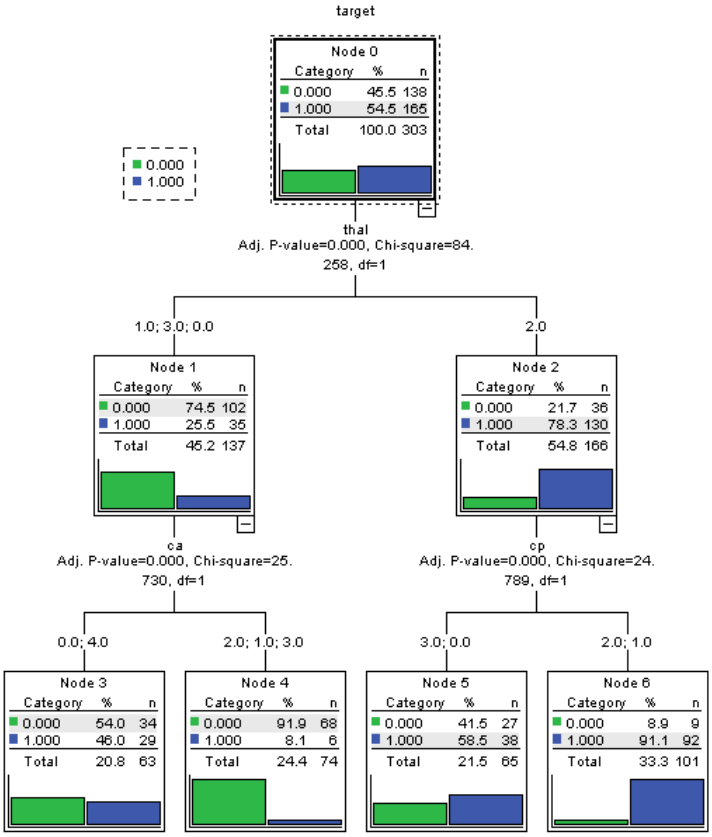
Out [9]: Text (5.0, 1.0, "Scores of Decision Tree Classifier for different number of maximum features")

Scores obtained for different number of maximum features




```
In [9]:
In [10]:
print("The score for Decision Tree Classifier is {}% with {} maximum
features.".format(dt_scores[17]*100,[2,4,18]))
```

Output of Decsion Tree Algorithm (SPSS)



The accuracy achieved for decision tree classifier is 79% [19]. The data parameters are varied to improve the scores. Doctors and Physician scan use these parameters to match the similarity of different models already available on the cloud databaseand can prepare personalized diagnosis. The target value explains the chances of person diseases, and analyzes if the patient is suffering from heart disease or not, if the result is 0, it indicates the absence of disease and if the result is 1,2,3,4, it indicates presence based on the type. Following parameters are considered while performing the analysis

Limitationsand Way Forward–

The major challenge one can face after implementing digital twin technology is related to cybersecurity. The increased cybercrimes have resulted in the loss of data. The massive amount of data that is being collected from various sources, each of which represents potential areas of weakness. In order to tackle these issues, cloud-based simulation software and storage can be used to prevent data leak and data loss. Data governance can also be implemented using policies such as HIPPA [20]. These methods if adopted properly could reduce the risk of data loss.

Conclusion –

Digital twin technology helps in tailoring medical treatment to the individual using a machine-learning algorithm (decision tree). The model is developed to capture both real-time and historical data of the patients. The observations will guide doctors, healthcare organizations, nurses, and patients in using simulation technologies to predict, manage heart disease, and device a model that could be used for further diagnosis. Effective and tailored

medical treatment can be developed using these technologies and more personalized treatment will be available to the individuals.

Acknowledgment –

This paper of Aashish Bende and the research behind it would not have been possible without (Madni, Madni, and Lucero, 2019)exacting attention to detail have been an inspiration and kept my work on track from my first encounter with the topic and the several analysis and resource collection to the final draft of this paper. (Houten, 2018)

References -

Boschert, S. and Rosen, R. (2016). Digital Twin—The Simulation Aspect. *Mechatronic Futures* , 59-74.

Brosinsky, C., Westermann, D., and Krebs, R. (2018). Recent and prospective developments in power system control centers: Adapting the digital twin technology for application in power system control centers.

Cunbo,Z., Jianhua, L., Hui, X., Xiaoyu, D., Shaoli, and Gang, W. (2017). Connotation,architecture and trends of product digital twin. *Computer Integrated Manufacturing Systems* .

Gochhait, S., Shou, D. T., and Fazalbhoy, S. (2020). *Cloud Computing Applications and Techniques for E-Commerce*. IGI Global. http://doi:10.4018/978-1-7998-1294-4

Houten, H. (2018). The rise of the digital twin: how healthcare can benefit.

Madni, A., Madni, C., and Lucero, S. (2019). Leveraging Digital Twin Technology in Model-Based Systems Engineering. *MDPI* , 13.

Roylance, D. (2001). Finite Element Analysis. 16.

Rimal, Y. ,and Gochhait,S. (2019). ”Machine Learning Neural Analysis Noisy Data”, International Journal of Engineering and Advanced Technology , ISSN: 2249-8958, 8(6),08/2019

Sanders, R., Martin, R., Frumin, H., and Goldberg, M. (1984). Data Storage and Retrieval by Implantable Pacemakers for Diagnostic Purposes. *Pacing Clin Electrophysiol* , 1228-33.

Schleich, B., Anwer, N., Mathieu, L., and Wartzack, S. (2017). Shaping the digital twin for design and production engineering. *CIRP Annals* , 141-144.

Schroeder, G., Steinmetz, C., Pereira, C., and Espindola D. (2016). Digital Twin Data Modeling with AutomationML and a Communication Methodology for Data Exchange. *IFAC-PapersOnLine* , 12-17.

Sivalingam,K., Sepulveda,M., Spring,M., and Davis,P. (2018). A Review and Methodology Development for Remaining Useful Life Prediction of Offshore Fixed and Floating Wind turbine Power Converter with Digital Twin Technology Perspective.

Tao, F., Cheng, J., Qi, Q., Zhang, M., Zhang, H., and Sui, F. (2017). Digital twin-driven product design, manufacturing and service with big data. *The International Journal of Advanced Manufacturing Technology* , 94.

Tao, F., Sui, F., Liu, A., Qi, Q., Zhang, M., Song, B., Guo, Z., Lu, C., and Nee, A. (2018). Digital twin-driven product design framework. *International Journal of Production Research* , 3935-3953.

Uhlemann, T., Lehmann, C., and Steinhilper, R. (2017). The Digital Twin: Realizing the Cyber-Physical Production System for Industry 4.0. *Procedia CIRP* , 335-340.

Zhuang, C., Liu, J., and Xiong, H. (2018). Digital twin-based smart production management and control framework for the complex product assembly shop-floor. *The International Journal of Advanced Manufacturing Technology* , 94.