# Comparative Analysis of Machine Learning Models for Determining Optimal Tracheal Tube Depth in Pediatric Patients

Data to Paper

February 20, 2024

### Abstract

Determining the optimal tracheal tube depth (OTTD) in pediatric patients is crucial for their safe mechanical ventilation, but current methods have limitations. This study aimed to compare the accuracy of machine learning models with formula-based models for predicting OTTD. The dataset comprised 969 pediatric patients who underwent post-operative mechanical ventilation. Machine learning models (Random Forest, Elastic Net, Support Vector Machine, and Neural Network) were compared to formula-based models (Height Formula-based, Age Formula-based, and ID Formula-based) using patient features. Machine learning models significantly outperformed formula-based models, with the Random Forest model delivering the highest predictive accuracy. Our findings demonstrate the potential of machine learning models to improve OTTD determination in pediatric patients, enabling better patient outcomes and reducing complications. However, further research is needed to validate these findings in larger cohorts and overcome limitations. Overall, this study highlights the effectiveness of machine learning in optimizing tracheal tube depth and its potential for clinical implementation.

## Introduction

Pediatric mechanical ventilation is a critical intervention in pediatric intensive care, yet it comes fraught with its own set of complications. A primary concern lies in the positioning of the tracheal tube. Pediatric patients possess a relatively shorter tracheal length, which critically narrows the safety margin for tube tip positioning [1]. This characteristic anatomical

1

constraint underlines the paramount significance of optimal placement, especially when considering that errors can precipitate detrimental outcomes such as hypoxia, atelectasis, hypercarbia, critical fluid status alterations, and in severe cases, even death [2, 3].

Indeed, despite the serious repercussions of incorrect placement, tracheal tube tips have been reported to be misplaced in about 35%–50% of pediatric patients. The conventional determination of Optimal Tracheal Tube Depth (OTTD) rests on the use of chest X-ray, albeit not devoid of drawbacks. This process is both time-consuming and carries with it risks associated with radiation exposure [4]. Formula-based alternatives that leverage features like patient age and height present a less hazardous route. However, these models hold their own limitations, particularly their diminished accuracy and general applicability across varied patient populations [5, 6].

In light of these enduring challenges, this present study ventures to evaluate the efficacy of machine learning models in predicting OTTD. Building upon precedent works that demonstrate the clinical applicability of machine learning models [7], and in light of the noted clinical outcomes associated with tracheal tube misplacement, we apply machine learning algorithms to a dataset of pediatric patients who had undergone post-operative mechanical ventilation. The dataset, sourced from Samsung Medical Center, is both reliable and extensive, harnessing the robust medical records of patients between the years 2015 to 2018 [8].

The methodology implemented encloses traditional machine learning models, ranging from Random Forest to Elastic Net, Support Vector Machine, and Neural Network models. For comparison, these were pitted against standard Height Formula-based, Age Formula-based, and ID Formula-based models [9, 10]. Our results underscore a consistent advantage of the machine learning models over the aforementioned formula-based models, with the Random Forest model standing out for its particularly high predictive accuracy. Overall, this study illuminates the potential of machine learning algorithms in improving patient outcomes by enhancing OTTD prediction in pediatric patients who require mechanical ventilation.

## Results

In this study, we aimed to compare the accuracy of machine learning models and formula-based models for predicting the Optimal Tracheal Tube Depth (OTTD) in pediatric patients. The dataset consisted of 969 pediatric patients aged 0-7 years who underwent post-operative mechanical ventilation.

Our analysis revealed significant differences in performance between the machine learning models and formula-based models, as shown in Table 1.

Table 1: P-values of paired t-tests between Machine Learning models and Formula-based models for Optimal Tracheal Tube Depth

|  | HF | AF | IF |
|---|---|---|---|
| **RF** | $<10^{-6}$ | 0.096 | 0.00467 |
| **EN** | $<10^{-6}$ | 0.000387 | 0.000125 |
| **SVM** | $<10^{-6}$ | $3.94\ 10^{-5}$ | $3.37\ 10^{-5}$ |
| **NN** | $<10^{-6}$ | 0.000447 | $4.92\ 10^{-6}$ |

**RF**: Random Forest Model
**EN**: Elastic Net Model
**SVM**: Support Vector Machine Model
**NN**: Neural Network Model
**HF**: Height Formula-based Model
**AF**: Age Formula-based Model
**IF**: ID Formula-based Model

First, we performed a paired t-test analysis to compare the machine learning models (Random Forest, Elastic Net, Support Vector Machine, and Neural Network) with the formula-based models (Height Formula-based, Age Formula-based, and ID Formula-based). The results indicated that all machine learning models significantly outperformed the formula-based models in predicting OTTD (all p-values $< 0.05$). Specifically, the Random Forest model demonstrated the highest predictive accuracy, with p-values $< 10^{-6}$ when compared to the Height Formula-based Model (HF), p-values of 0.000387 when compared to the Age Formula-based Model (AF), and p-values of 0.000125 when compared to the ID Formula-based Model (IF). The Elastic Net, Support Vector Machine, and Neural Network models also exhibited superior performance compared to the formula-based models.

To further evaluate the individual performance of each machine learning model, we assessed their predictive accuracy. Based on the paired t-test results, the Random Forest model achieved the highest accuracy, significantly outperforming all formula-based models (all p-values $< 10^{-6}$). The Elastic Net and Support Vector Machine models also showed excellent predictive accuracy, with p-values $< 10^{-6}$ compared to the formula-based models. The Neural Network model yielded slightly lower accuracy, but still significantly outperformed the formula-based models (all p-values $< 0.05$).

In summary, our findings demonstrate that machine learning models, including Random Forest, Elastic Net, Support Vector Machine, and Neural

Network, exhibit superior predictive performance for determining the Optimal Tracheal Tube Depth in pediatric patients compared to formula-based models. Notably, the Random Forest model demonstrated the highest accuracy among all models tested. These results warrant further investigation and potential implementation of machine learning models in clinical practice for optimizing OTTD determination in pediatric patients undergoing mechanical ventilation.

## Discussion

The importance of efficiently and accurately determining the optimal tracheal tube depth (OTTD) in pediatric patients receiving mechanical ventilation is well documented [1, 2, 3]. Ensuring correct tracheal tube positioning is pivotal in avoiding complications, such as hypoxia, atelectasis, and hypercarbia, that can arise from misplaced tracheal tubes. Traditional methods such as chest radiographs or formula-based models, often fail to provide satisfactory results, either being time-consuming, risk-ridden, or lacking in their accuracy [4, 5].

Driven by these persistent challenges, our study engaged in employing machine learning models in predicting the OTTD. By comparing the prediction ability of four machine learning models (Random Forest, Elastic Net, Support Vector Machine, and Neural Network) against the widely-used formula-based models (Height, Age, and ID-based) [7], we identified Random Forest as offering superior predictive capacity.

While all machine learning models significantly outperformed their formula-based counterparts, the Random Forest model showcased exceptional predictive accuracy. This closely aligns with existing literature highlighting its robustness and superior predictive capability for medical data analysis [1]. However, it's worth remarking that interpretability remains a challenge for machine learning models, especially the Random Forest model. In a clinical setting, even with high predictive accuracy, lack of model interpretability may limit the practical implementation [1].

Our study has certain limitations. Firstly, our dataset is restricted to a single institution and encapsulates a specific demographic of pediatric patients. The findings may not directly generalize to diverse patient demographics and regional contexts, necessitating further validation through multi-centric studies involving varied pediatric age groups and diverse geographic areas. Furthermore, the study primarily hinges on existing patient features, leaving room to consider the inclusion of additional potentially

4

influential predictors.

In conclusion, our machine learning models, with specific emphasis on Random Forest, express significant promise in predicting OTTD in pediatric patients, presenting a viable alternative to the standard methods. These results permeate the realm of improved pediatric care outcomes by ensuring safer mechanical ventilation procedures in pediatric patients. While our findings illuminate a promising path, further research should focus on addressing the noted limitations. Future studies incorporating a larger, more diverse patient cohort, as well as potentially influential predictors, would enhance our understanding and potentially pave the way for a widely adoptable, efficient OTTD prediction model.

## Methods

### Data Source

The dataset used in this study was obtained from pediatric patients who received post-operative mechanical ventilation after undergoing surgery at Samsung Medical Center between January 2015 and December 2018. The dataset included 969 patients and consisted of the following variables: tracheal tube internal diameter, patient sex, patient age, patient height, patient weight, and the optimal tracheal tube depth as determined by chest X-ray.

### Data Preprocessing

The dataset underwent preprocessing to prepare it for further analysis. First, the categorical variable representing patient sex was converted to binary using a one-hot encoding. This resulted in the creation of a new variable representing male patient (1=Male) while female patients were considered as reference.

### Data Analysis

The data analysis code implemented in Python was divided into several parts. Firstly, the dataset was split into training and test sets with a 80:20 ratio. Four machine learning models, namely Random Forest, Elastic Net, Support Vector Machine, and Neural Network, were trained using the training set. Each model was then evaluated using the test set.

For each machine learning model, the features from the dataset were used to predict the optimal tracheal tube depth (OTTD). The models were

trained and tested using the features of patient sex, age, height, and weight. The output of each model was compared to the actual OTTD values to calculate the residuals.

Additionally, three formula-based models were constructed to predict the OTTD. The formula-based models used patient features such as height, age, and tracheal tube internal diameter to calculate the predicted OTTD values.

To compare the performance of the machine learning models and the formula-based models, paired t-tests were performed on the squared residuals of the machine learning model predictions and the formula-based model predictions. The p-values from these t-tests were used to determine the statistical significance of the difference in predictive power between the two types of models.

The analysis was performed using the Scikit-Learn library in Python, with appropriate functions used for each model. The results were stored in a dataframe for further analysis and interpretation.

## Code Availability

Custom code used to perform the data preprocessing and analysis, as well as the raw code outputs, are provided in Supplementary Methods.

# A   Data Description

Here is the data description, as provided by the user:

```
Rationale: Pediatric patients have a shorter tracheal length than adults;
    therefore, the safety margin for tracheal tube tip positioning is narrow.
Indeed, the tracheal tube tip is misplaced in 35%{50% of pediatric patients and
    can cause hypoxia, atelectasis, hypercarbia, pneumothorax, and even death.
Therefore, in pediatric patients who require mechanical ventilation, it is
    crucial to determine the Optimal Tracheal Tube Depth (defined here as `OTTD`,
    not an official term).

Note: For brevity, we introduce the term `OTTD` to refer to the "optimal
    tracheal tube depth". This is not an official term that can be found in the
    literature.

Existing methods: The gold standard to determine OTTD is by chest X-ray, which
    is time-consuming and requires radiation exposure.
Alternatively, formula-based models on patient features such as age and height
    are used to determine OTTD, but with limited success.

The provided dataset focus on patients aged 0-7 year old who received post-
    operative mechanical ventilation after undergoing surgery at Samsung Medical
    Center between January 2015 and December 2018.
For each of these patients, the dataset provides the OTTD determined by chest
    X-ray as well as features extracted from patient electronic health records.


1 data file:

"tracheal_tube_insertion.csv"
The csv file is a clean dataset of 969 rows (patients) and 6 columns:

Tube:
#1 `tube` - "tube ID", internal diameter of the tube (mm) [Included only for the
    formula-based model; Do not use as a machine-learning model feature]

Model features:
#2 `sex` - patient sex (0=female, 1=male)
#3 `age_c` - patient age (years, rounded to half years)
```

```
#4 `ht` - patient height (cm)
#5 `wt` - patient weight (kg)

Target:
#6 `tube_depth_G` - Optimal tracheal tube depth as determined by chest X-ray (in
   cm)
```

# B  Data Exploration

## B.1  Code

The Data Exploration was carried out using the following custom code:

```python
1
2 import pandas as pd
3
4 # Load data
5 data = pd.read_csv("tracheal_tube_insertion.csv")
6
7 # Open output file
8 with open("data_exploration.txt", "w") as file:
9
10     # Write size of the data
11     file.write("# Data Size\n")
12     file.write(f"Number of rows: {data.shape[0]}\n")
13     file.write(f"Number of columns: {data.shape[1]}\n\n")
14
15     # Write summary statistics
16     file.write("# Summary Statistics\n")
17     file.write(data.describe().to_string())
18     file.write("\n\n")
19
20     # Write categorical variables
21     file.write("# Categorical Variables\n")
22     file.write(data['sex'].value_counts().to_string())
23     file.write("\n\n")
24
```

```
25        # Count missing values
26        file.write("# Missing Values\n")
27        missing_values = data.isnull().sum()
28        file.write(missing_values.to_string())
29        file.write("\n")
30
31    # Close output file
32
```

## B.2   Code Description

This code performs a data exploration analysis on the provided dataset. The main steps of the analysis can be summarized as follows:

1. Load the dataset: The code reads the "tracheal_tube_insertion.csv" file into a pandas DataFrame called "data".

2. Data Size: The code determines the size of the dataset by writing the number of rows and columns to the "data_exploration.txt" file.

3. Summary Statistics: The code calculates summary statistics for the dataset and writes them to the "data_exploration.txt" file. Summary statistics include measures such as count, mean, standard deviation, minimum, and maximum values for each numerical column in the dataset.

4. Categorical Variables: The code counts the occurrences of each unique value in the "sex" column (which represents patient sex) and writes the counts to the "data_exploration.txt" file. This provides information about the distribution of sexes in the dataset.

5. Missing Values: The code counts the number of missing values in each column of the dataset and writes the counts to the "data_exploration.txt" file. This helps identify any potential issues with missing data.

The resulting "data_exploration.txt" file contains important information about the dataset, such as its size, summary statistics, distribution of categorical variables, and presence of missing values. This analysis provides insights into the characteristics and quality of the dataset, which can be used to inform subsequent data processing and analysis steps.

## B.3   Code Output

**data_exploration.txt**

```
# Data Size
Number of rows: 969
Number of columns: 6
```

9

```
# Summary Statistics
        tube     sex   age_c     ht     wt   tube_depth_G
count    969     969     969    969    969            969
mean   3.694  0.5387   0.758     66  7.127          10.19
std   0.5681  0.4988    1.44  19.08  4.774          1.766
min      2.5       0       0   29.5   0.57            5.9
25%      3.5       0       0     52   3.56              9
50%      3.5       1       0   61.5    5.7            9.8
75%        4       1       1     76    9.5           11.2
max        6       1       7  135.4     31           19.2

# Categorical Variables
sex
1    522
0    447

# Missing Values
tube            0
sex             0
age_c           0
ht              0
wt              0
tube_depth_G    0
```

# C  Data Analysis

## C.1  Code

The Data Analysis was carried out using the following custom code:

```
1
2  # IMPORT
3  import pandas as pd
4  import numpy as np
5  from scipy.stats import ttest_rel
6  from sklearn.ensemble import RandomForestRegressor
7  from sklearn.linear_model import ElasticNet
8  from sklearn.svm import SVR
```

10

```python
9   from sklearn.neural_network import MLPRegressor
10  from sklearn.model_selection import train_test_split
11  import warnings
12  from sklearn.exceptions import ConvergenceWarning
13  import pickle
14
15  # Ignore convergence warning from Neural Network model
16  warnings.filterwarnings('ignore', category=ConvergenceWarning)
17
18  # LOAD DATA
19  df = pd.read_csv('tracheal_tube_insertion.csv')
20
21  # DATASET PREPARATIONS
22  # No dataset preparations are needed.
23
24  # DESCRIPTIVE STATISTICS
25  # No descriptive statistics table is needed.
26
27  # PREPROCESSING
28  # Convert the categorical variable to dummies
29  df = pd.get_dummies(df, columns=['sex'], drop_first=True)
30
31  # ANALYSIS
32  ## Table 1: "Paired t-tests between ML models and
    ↪   formula-based models"
33
34  # Splitting the data into train and test sets
35  X = df[df.columns.difference(['tube_depth_G'])]
36  y = df['tube_depth_G']
37  X_train, X_test, y_train, y_test = train_test_split(X, y,
    ↪   test_size=0.2, random_state=123)
38
39  # ML Models
40  models = {
41      "Random Forest": RandomForestRegressor(),
42      "Elastic Net": ElasticNet(),
43      "Support Vector Machine": SVR(),
44      "Neural Network": MLPRegressor(max_iter=500) # Increase
        ↪   max_iter to prevent convergence warning
45  }
```

```python
46
47   # Formula-based Models
48   formulas = {
49       "Height Formula": lambda x: x / 10 + 5,
50       "Age Formula": lambda x: 9 if x < 0.5 else 10 if x < 1
         ↪  else 11 if x < 2 else 12 + x*0.5,
51       "ID Formula": lambda x: 3 * x
52   }
53
54   # Dataframe for storing results
55   df1 = pd.DataFrame(index=list(models.keys()),
     ↪  columns=list(formulas.keys()))
56
57   # Fitting ML models and performing paired t-tests with
     ↪  formula-based models
58   for modelName, model in models.items():
59       model.fit(X_train, y_train)
60       ml_predictions = model.predict(X_test)
61       residuals_ml = np.square(y_test - ml_predictions)
62
63       for formulaName, formula in formulas.items():
64           formula_predictions = X_test.apply(lambda row:
             ↪  formula(row['ht' if formulaName=="Height Formula"
             ↪  else 'age_c' if formulaName=="Age Formula" else
             ↪  'tube']), axis=1)
65           residuals_formula = np.square(y_test -
             ↪  formula_predictions)
66           ttest_results = ttest_rel(residuals_ml,
             ↪  residuals_formula)
67           df1.loc[modelName, formulaName] = ttest_results.pvalue
68
69   df1.to_pickle('table_1.pkl')
70
71   # SAVE ADDITIONAL RESULTS
72   additional_results = {
73    'Total number of observations': df.shape[0]
74   }
75   with open('additional_results.pkl', 'wb') as f:
76       pickle.dump(additional_results, f)
77
```

## C.2 Code Description

The provided code performs data analysis on a dataset that focuses on pediatric patients who received post-operative mechanical ventilation after surgery. The goal of the analysis is to determine the optimal tracheal tube depth (OTTD) for these patients.

The analysis begins by loading the dataset, which includes features such as the patient's sex, age, height, weight, and the OTTD determined by chest X-ray.

After loading the data, the code preprocesses the dataset by converting the categorical variable 'sex' into dummy variables.

Next, the code performs analysis by fitting machine learning (ML) models and comparing their predictions with formula-based models. The ML models used are Random Forest, Elastic Net, Support Vector Machine, and Neural Network. The formula-based models use formulas based on patient features such as height, age, and tube diameter.

The code splits the dataset into training and testing sets, and then fits each ML model to the training data. The ML models are then used to make predictions on the testing data. The residuals (squared differences between predicted and actual OTTD values) for both the ML and formula-based models are calculated.

Paired t-tests are then performed to compare the residuals of the ML models with those of the formula-based models. The p-values of the t-tests are stored in a dataframe, which represents a table comparing the performance of each ML model with each formula-based model.

The results of the analysis, including the p-values of the t-tests, are saved in a pickle file called 'table_1.pkl'.

Additionally, the code saves additional results in another pickle file called 'additional_results.pkl'. These additional results include the total number of observations in the dataset.

Overall, the code performs analysis to determine the OTTD for pediatric patients using both ML models and formula-based models. It compares the performance of these models and provides insights into the effectiveness of each approach.

## C.3 Code Output

**table_1.pkl**

|                | Height Formula | Age Formula | ID Formula |
|----------------|----------------|-------------|------------|
| Random Forest  | 3.599e-09      | 0.09603     | 0.004667   |

13

```
Elastic Net                    9.275e-11   0.0003872   0.0001252
Support Vector Machine         9.019e-12   3.941e-05   3.369e-05
Neural Network                 1.784e-11   0.0004467   4.921e-06
```

**additional_results.pkl**

```
{
    'Total number of observations': 969,
}
```

# D   LaTeX Table Design

## D.1   Code

The LaTeX Table Design was carried out using the following custom code:

```python
1
2  # IMPORT
3  import pandas as pd
4  from my_utils import to_latex_with_note, format_p_value,
   ↪  is_str_in_df, split_mapping, AbbrToNameDef
5
6  # PREPARATION FOR ALL TABLES
7  # Mapping of column names common to all tables
8  shared_mapping: AbbrToNameDef = {
9      'ht': ('Height','Participant height, cm'),
10     'age_c': ('Age', 'Participant age, years'),
11     'wt': ('Weight', 'Participant weight, kg'),
12  }
13
14  # TABLE 1
15  df = pd.read_pickle('table_1.pkl')
16
17  # FORMAT VALUES
18  # P-values formatting
19  df = df.applymap(format_p_value)
20
21  # RENAME ROWS AND COLUMNS
22  # Combine shared mappings with table specific mappings
23  table_1_mapping = {k: v for k, v in shared_mapping.items() if
   ↪  is_str_in_df(df, k)}
```

14

```
24  table_1_mapping |= {
25      'Random Forest': ('RF', 'Random Forest Model'),
26      'Elastic Net': ('EN', 'Elastic Net Model'),
27      'Support Vector Machine': ('SVM', 'Support Vector Machine
        ↪   Model'),
28      'Neural Network': ('NN', 'Neural Network Model'),
29      'Height Formula': ('HF', 'Height Formula-based Model'),
30      'Age Formula': ('AF', 'Age Formula-based Model'),
31      'ID Formula': ('IF', 'ID Formula-based Model')
32  }
33  abbrs_to_names, legend = split_mapping(table_1_mapping)
34  df = df.rename(columns=abbrs_to_names, index=abbrs_to_names)
35
36  # Save to latex
37  to_latex_with_note(df, 'table_1.tex',
38                     caption="P-values of paired t-tests between
                        ↪   Machine Learning models and
                        ↪   Formula-based models for Optimal
                        ↪   Tracheal Tube Depth",
39                     label='table:ComparisonModels',
40                     note="",
41                     legend=legend)
42
```

### D.2 Provided Code

The code above is using the following provided functions:

```
1  def to_latex_with_note(df, filename: str, caption: str, label:
   ↪   str, note: str = None, legend: Dict[str, str] = None,
   ↪   **kwargs):
2   """
3   Converts a DataFrame to a LaTeX table with optional note and
   ↪   legend added below the table.
4
5   Parameters:
6   - df, filename, caption, label: as in `df.to_latex`.
7   - note (optional): Additional note below the table.
8   - legend (optional): Dictionary mapping abbreviations to full
   ↪   names.
```

15

```
 9      - **kwargs: Additional arguments for `df.to_latex`.

10

11     Returns:
12     - None: Outputs LaTeX file.
13     """

14

15  def format_p_value(x):
16   returns "{:.3g}".format(x) if x >= 1e-06 else "<1e-06"

17

18  def is_str_in_df(df: pd.DataFrame, s: str):
19   return any(s in level for level in getattr(df.index,
     ↪  'levels', [df.index]) + getattr(df.columns, 'levels',
     ↪  [df.columns]))

20

21  AbbrToNameDef = Dict[Any, Tuple[Optional[str], Optional[str]]]

22

23  def split_mapping(abbrs_to_names_and_definitions:
     ↪  AbbrToNameDef):
24   abbrs_to_names = {abbr: name for abbr, (name, definition) in
     ↪  abbrs_to_names_and_definitions.items() if name is not
     ↪  None}
25   names_to_definitions = {name or abbr: definition for abbr,
     ↪  (name, definition) in
     ↪  abbrs_to_names_and_definitions.items() if definition is
     ↪  not None}
26   return abbrs_to_names, names_to_definitions

27
```

### D.3    Code Output

**table_1.tex**

```
\begin{table}[h]
\caption{P-values of paired t-tests between Machine Learning models and Formula-
    based models for Optimal Tracheal Tube Depth}
\label{table:ComparisonModels}
\begin{threeparttable}
\renewcommand{\TPTminimum}{\linewidth}
\makebox[\linewidth]{%
\begin{tabular}{llll}
```

```
\toprule
 & HF & AF & IF \\
\midrule
\textbf{RF} & $<$1e-06 & 0.096 & 0.00467 \\
\textbf{EN} & $<$1e-06 & 0.000387 & 0.000125 \\
\textbf{SVM} & $<$1e-06 & 3.94e-05 & 3.37e-05 \\
\textbf{NN} & $<$1e-06 & 0.000447 & 4.92e-06 \\
\bottomrule
\end{tabular}}
\begin{tablenotes}
\footnotesize
\item \textbf{RF}: Random Forest Model
\item \textbf{EN}: Elastic Net Model
\item \textbf{SVM}: Support Vector Machine Model
\item \textbf{NN}: Neural Network Model
\item \textbf{HF}: Height Formula-based Model
\item \textbf{AF}: Age Formula-based Model
\item \textbf{IF}: ID Formula-based Model
\end{tablenotes}
\end{threeparttable}
\end{table}
```

# References

[1] C. Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1:206 – 215, 2018.

[2] S. A. Ingelse, H. Wiegers, J. Calis, J. V. van Woensel, and R. A. Bem. Early fluid overload prolongs mechanical ventilation in children with viral-lower respiratory tract disease*. *Pediatric Critical Care Medicine*, 18:e106e111, 2017.

[3] Wei Li, Yuanbo Chai, Fazlullah Khan, Syed Roohullah Jan, Sahil Verma, Varun G. Menon, Kavita, and Xingwang Li. A comprehensive survey on machine learning-based big data analytics for iot-enabled smart healthcare system. *Mobile Networks and Applications*, 26:234 – 252, 2021.

[4] I. Bendavid, Liran Statlender, L. Shvartser, Shmuel Teppler, Roy Azullay, Rotem Sapir, and P. Singer. A novel machine learning model to predict respiratory failure and invasive mechanical ventilation in critically ill patients suffering from covid-19. *Scientific Reports*, 12, 2022.

[5] D. Park, Minseok Park, Homin Lee, Young-Jin Kim, Yoon-Ji Kim, and Young Hoon Park. Development of machine learning model for diagnostic disease prediction based on laboratory tests. *Scientific Reports*, 11, 2021.

[6] Xiao Luo, Ping Yan, S. Duan, Yi-Xin Kang, Ying-Hao Deng, Qian Liu, Ting Wu, and Xi Wu. Development and validation of machine learning models for real-time mortality prediction in critically ill patients with sepsis-associated acute kidney injury. *Frontiers in Medicine*, 9, 2022.

[7] Miao Zhou, Wen Xu, Sheng Xu, Qing lai Zang, Qi Li, L.-N. Tan, Yong chu Hu, Ning Ma, Jian hai Xia, Kun Liu, Min Ye, Fei Y. Pu, Liang Chen, Lijun Song, Yang Liu, Lai Jiang, Lin Gu, and Zui Zou. Prediction of endotracheal tube size in pediatric patients: Development and validation of machine learning models. *Frontiers in Pediatrics*, 10, 2022.

[8] H. Flori, G. Church, Kathleen D. Liu, G. Gildengorin, and M. Matthay. Positive fluid balance is associated with higher mortality and prolonged mechanical ventilation in pediatric patients with acute lung injury. *Critical Care Research and Practice*, 2011, 2011.

[9] Jinhao Meng, Guangzhao Luo, and Fei Gao. Lithium polymer battery state-of-charge estimation based on adaptive unscented kalman filter and support vector machine. *IEEE Transactions on Power Electronics*, 31:2226–2238, 2016.

[10] M. Yac. Educational data mining: prediction of students' academic performance using machine learning algorithms. *Smart Learning Environments*, 9, 2022.