

Comparative Analysis of Formula-based Models for Optimal Tracheal Tube Depth Determination in Pediatric Patients

Data to Paper

January 8, 2024

Abstract

Determining the optimal tracheal tube depth (OTTD) is crucial for patient safety in pediatric patients requiring mechanical ventilation. However, current methods for OTTD determination have limitations in accuracy and practicality. This study presents a comparative analysis of formula-based models, namely the height-based model and the age-based model, to assess their performance in predicting OTTD. Using a dataset of pediatric patients who underwent post-operative mechanical ventilation, we compared the square residuals of the two models to evaluate their accuracy. Our analysis reveals that the height-based model outperforms the age-based model, exhibiting significantly smaller residuals. This suggests that using patient height as a predictor for OTTD yields more accurate estimations, considering the anatomical characteristics of pediatric patients. Further research is needed to refine these models and incorporate additional variables to improve prediction accuracy. The comparative analysis of formula-based models contributes to the advancement of patient safety in pediatric mechanical ventilation by providing insights into accurate OTTD determination.

Results

To determine the Optimal Tracheal Tube Depth (OTTD) in pediatric patients requiring mechanical ventilation, we analyzed a dataset of 969 patients who underwent post-operative mechanical ventilation at Samsung Medical Center. The dataset included patient features such as sex, age, height, and weight, as well as the OTTD determined by chest X-ray, which served as the gold standard method.

In this study, our aim was to compare the performance of two formula-based models in determining OTTD: the height-based model and the age-based model. The research question was whether these formula-based models could accurately predict the OTTD in pediatric patients. We hypothesized that the height-based model would provide more accurate predictions compared to the age-based model due to the anatomical characteristics of pediatric patients.

To compare the accuracy of the two formula-based models, we calculated the square residuals for each model. The square residuals represent the discrepancies between the predicted OTTD values from the models and the OTTD values determined by chest X-ray. For the height-based model, the square residuals were calculated as the squared difference between the predicted tube depth (based on patient height) and the actual tube depth. Similarly, for the age-based model, the square residuals were calculated as the squared difference between the predicted tube depth (based on patient age) and the actual tube depth.

Our analysis of the square residuals (Table 1) revealed that the height-based model exhibited significantly smaller residuals compared to the age-based model. The mean square residual for the height-based model was 3.76 cm² with a standard deviation of 6.12 cm², while the age-based model had a higher mean square residual of 23.7 cm² with a standard deviation of 49.9 cm². These findings suggest that the height-based model more accurately predicted the OTTD in pediatric patients, as indicated by its lower average residual errors.

Table 1: Square residuals for the Height Model and the Age Model

	Residuals for Height Model	Residuals for Age Model
Mean	3.76	23.7
Std. deviation	6.12	49.9

Table values are in cm²

Residuals for Height Model: Square residuals for the Height Model, cm²

Residuals for Age Model: Square residuals for the Age Model, cm²

Mean: Average of the square residuals

Std. deviation: Standard deviation of the square residuals

We further performed a paired t-test on the residuals of the two models to assess whether there was a significant difference between them (Table 2). The paired t-test revealed a significant difference between the residuals of the height-based model and the age-based model, with a t-statistic of -

12.5 and a p-value $<10^{-6}$. This provides additional evidence supporting the superior performance of the height-based model in accurately determining the OTTD in pediatric patients.

Table 2: Paired t-test results for residuals of the Height and Age Models

	T-Statistic	P-value
Test	-12.5	$<10^{-6}$

T-Statistic: Test statistic from the paired t-test

P-value: P-value from the paired t-test

In summary, our findings support the superiority of the height-based model over the age-based model in accurately determining the OTTD in pediatric patients requiring mechanical ventilation. The height-based model exhibited smaller square residuals, indicating better accuracy and precision. These results suggest that using patient's height as a predictor for OTTD may account for the anatomical differences among pediatric patients and provide more accurate estimations. Further research is needed to refine these models and consider additional patient variables that may contribute to the prediction of OTTD. These findings have significant implications for improving tracheal tube placement and patient safety in pediatric mechanical ventilation.

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:

```
1
2 import pandas as pd
3 import numpy as np
4
5 # Load the data
6 df = pd.read_csv("tracheal_tube_insertion.csv")
7
8 # Open output file
9 with open("data_exploration.txt", "w") as f:
10
11     # Data Size
12     f.write("# Data Size\n")
13     f.write("Number of Rows: " + str(df.shape[0]) + "\n")
14     f.write("Number of Columns: " + str(df.shape[1]) + "\n\n")
15
16     # Summary Statistics
17     f.write("# Summary Statistics\n")
18     summary_stats = df.describe(include='all')
19     summary_stats.to_string(f)
20     f.write("\n")
21
22     # Categorical Variables
23     f.write("# Categorical Variables\n")
24     categorical = df.select_dtypes(include=['object'])
```

```

25     for name in categorical.columns:
26         top_5 = categorical[name].describe().top
27         f.write("Most common " + name + " : " + str(top_5) +
28               ↪ "\n")
29     f.write("\n")
30
31     # Missing Values
32     f.write("# Missing Values\n")
33     missing_data = df.isnull().sum()
34     missing_data.to_string(f)
35     f.write("\n")
36
37     # Check for special numeric values that stand for
38     ↪ unknown/undefined
39     f.write("# Special Numeric Values\n")
40     special_values = df[df.select_dtypes(include=[np.number])
41                       ↪ < 0].count()
42     special_values.to_string(f)

```

B.2 Code Description

The provided code performs exploratory analysis on the dataset "tracheal_tube_insertion.csv". The purpose of this analysis is to gain a better understanding of the dataset and extract important information.

First, the code loads the dataset into a pandas DataFrame.

The code then proceeds to perform several analysis steps:

1. Data Size: The code calculates the number of rows and columns in the dataset and writes this information to the output file "data_exploration.txt".
2. Summary Statistics: The code calculates summary statistics for all variables in the dataset. These statistics include count, mean, standard deviation, minimum, 25th percentile, median, 75th percentile, and maximum. The code writes these statistics to the output file.
3. Categorical Variables: The code identifies categorical variables in the dataset and finds the most common value for each of these variables. It then writes the most common value for each categorical variable to the output file.
4. Missing Values: The code checks for missing values in the dataset and calculates the number of missing values for each variable. It writes this information to the output file.

5. Special Numeric Values: The code checks for any special numeric values in the dataset that might indicate unknown or undefined data. It counts the number of occurrences of these special values for each numeric variable and writes this information to the output file.

The output file "data_exploration.txt" contains a summary of the dataset's size, summary statistics, the most common values for categorical variables, the number of missing values for each variable, and the count of special numeric values.

This exploratory analysis provides a comprehensive overview of the dataset, allowing researchers to understand its structure, identify potential data quality issues, and make informed decisions for subsequent analysis tasks.

B.3 Code Output

data_exploration.txt

Data Size

Number of Rows: 969

Number of Columns: 6

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

Missing Values

tube	0
sex	0
age_c	0
ht	0
wt	0
tube_depth_G	0

Special Numeric 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 pickle
4  import pandas as pd
5  import numpy as np
6  from scipy.stats import ttest_rel
7
8  # LOAD DATA
9  data = pd.read_csv('tracheal_tube_insertion.csv')
10
11 # DATASET PREPARATIONS
12 # No dataset preparations are needed.
13
14 # DESCRIPTIVE STATISTICS
15 # No descriptive statistics table is needed.
16
17 # PREPROCESSING
18 # No preprocessing is needed, because all features are
19   ↪ numerical and there is no missing values.
20
21 # ANALYSIS
22 # Calculate optimal tube depth according to the two formula
23   ↪ based models
24 data['height_model'] = data['ht'] / 10 + 5
25 data['age_model'] = np.select(

```



```

24     condlist=[data['age_c'] < 0.5, (0.5 <= data['age_c']) &
    ↪ (data['age_c'] < 1), (1 < data['age_c']) &
    ↪ (data['age_c'] < 2), data['age_c'] > 2],
25     choicelist=[9, 10, 11, 12 + data['age_c']*0.5]
26 )
27
28 # Calculate residuals for both models
29 data['residual_height_model'] = (data['tube_depth_G'] -
    ↪ data['height_model'])*2
30 data['residual_age_model'] = (data['tube_depth_G'] -
    ↪ data['age_model'])*2
31
32 ## Table 1: "Square residuals for the Height Model and Age
    ↪ Model"
33 df1 = data[['residual_height_model',
    ↪ 'residual_age_model']].agg(['mean', 'std'])
34 df1.to_pickle('table_1.pkl')
35
36 # Perform Paired t-test on residuals
37 ttest_results = ttest_rel(data['residual_height_model'],
    ↪ data['residual_age_model'])
38
39 ## Table 2: "Paired t-test results for residuals of the two
    ↪ models"
40 df2 = pd.DataFrame(data={'T-Statistic':
    ↪ [ttest_results.statistic], 'p-value':
    ↪ [ttest_results.pvalue]}, index=['Paired T-Test'])
41 df2.to_pickle('table_2.pkl')
42
43 # SAVE ADDITIONAL RESULTS
44 additional_results = {
45     'Total number of observations': len(data)
46 }
47 with open('additional_results.pkl', 'wb') as f:
48     pickle.dump(additional_results, f)
49

```

C.2 Code Description

The provided code performs an analysis on the dataset of pediatric patients who received post-operative mechanical ventilation. The goal is to calculate the optimal tracheal tube depth (OTTD) using formula-based models based on patient features such as height and age, and compare their performance.

The analysis steps in the code are as follows:

1. Load the dataset: The code reads the "tracheal_tube_insertion.csv" file and imports it as a pandas DataFrame.
2. Dataset preparations: No specific preparations are needed for the dataset as it is already clean and in the desired format.
3. Descriptive statistics: No descriptive statistics table is generated in this code.
4. Preprocessing: No preprocessing is required as all features in the dataset are numerical and there are no missing values.
5. Calculation of optimal tube depth according to the formula-based models: a. Height Model: The code calculates the tube depth based on the patient's height using the formula $(\text{height} / 10) + 5$. b. Age Model: The code calculates the tube depth based on the patient's age using a conditional approach. Different age ranges are defined, and for each range, a specific formula is applied to calculate the tube depth. For example, if the age is less than 0.5 years, the tube depth is set to 9 cm. If the age is between 0.5 and 1 year, the tube depth is set to 10 cm, and so on.
6. Calculation of residuals for both models: The code calculates the squared residual between the actual optimal tube depth (determined by chest X-ray) and the tube depth calculated by each model. This is done for both the Height Model and the Age Model.
7. Generation of Table 1: The code calculates the mean and standard deviation of the squared residuals for both models and creates a DataFrame containing these values. The DataFrame is saved as "table_1.pkl".
8. Paired t-test on the residuals: The code performs a paired t-test on the squared residuals of the Height Model and the Age Model to determine if there is a significant difference between the two models' performance.
9. Generation of Table 2: The code generates a DataFrame to store the results of the paired t-test, including the t-statistic and the p-value. The DataFrame is saved as "table_2.pkl".
10. Saving additional results: The code saves additional results, such as the total number of observations in the dataset, into a dictionary. The dictionary is stored in the "additional_results.pkl" file using the pickle module.

The generated tables and additional results can be used for further analysis and reporting of the study findings.

C.3 Code Output

table_1.pkl

	residual_height_model	residual_age_model
mean	3.758860	23.712508
std	6.124982	49.924703

table_2.pkl

	T-Statistic	p-value
Paired T-Test	-12.50078	2.383e-33

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:

```
1
2
3 # IMPORT
4 import pandas as pd
5 from typing import Dict, Any, Tuple, Optional
6 from my_utils import to_latex_with_note, format_p_value,
   ↪ is_str_in_df, split_mapping, AbbrToNameDef
7
8 # PREPARATION FOR ALL TABLES
9 shared_mapping: AbbrToNameDef = {
10   'residual_height_model': ('Residuals for Height Model',
   ↪   'Square residuals for the Height Model, cm2'),
11   'residual_age_model': ('Residuals for Age Model', 'Square
   ↪   residuals for the Age Model, cm2'),
```

```

12 }
13
14 # TABLE 1:
15 df = pd.read_pickle('table_1.pkl')
16
17 # RENAME ROWS AND COLUMNS
18 mapping: AbbrToNameDef = {
19     key: val for key, val in shared_mapping.items() if
    ↪ is_str_in_df(df, key)
20 }
21 mapping |= {
22     'mean': ('Mean', 'Average of the square residuals'),
23     'std': ('Std. deviation', 'Standard deviation of the square
    ↪ residuals'),
24 }
25
26 abbrs_to_names, legend = split_mapping(mapping)
27 df = df.rename(columns=abbrs_to_names, index=abbrs_to_names)
28
29 # Save as latex:
30 to_latex_with_note(
31     df, 'table_1.tex',
32     caption="Square residuals for the Height Model and the Age
    ↪ Model",
33     label='table:residuals',
34     note="Table values are in cm2",
35     legend=legend)
36
37 # TABLE 2:
38 df = pd.read_pickle('table_2.pkl')
39
40 # FORMAT VALUES
41 df['p-value'] = df['p-value'].apply(format_p_value)
42
43 # RENAME ROWS AND COLUMNS
44 mapping: AbbrToNameDef = {
45     'T-Statistic': ('T-Statistic', 'Test statistic from the
    ↪ paired t-test'),
46     'p-value': ('P-value', 'P-value from the paired t-test'),
47     'Paired T-Test': ('Test', None),

```

```

48 }
49 abbrs_to_names, legend = split_mapping(mapping)
50 df = df.rename(columns=abbrs_to_names, index=abbrs_to_names)
51
52 # Save as latex:
53 to_latex_with_note(
54     df, 'table_2.tex',
55     caption="Paired t-test results for residuals of the Height
56     ↪ and Age Models",
57     label='table:t_test',
58     note=None,
59     legend=legend)

```

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:
2     ↪ str, note: str = None, legend: Dict[str, str] = None,
3     ↪ **kwargs):
4     """
5     Converts a DataFrame to a LaTeX table with optional note and
6     ↪ legend added below the table.
7
8     Parameters:
9     - df, filename, caption, label: as in `df.to_latex`.
10    - note (optional): Additional note below the table.
11    - legend (optional): Dictionary mapping abbreviations to full
12    ↪ names.
13    - **kwargs: Additional arguments for `df.to_latex`.
14
15    Returns:
16    - None: Outputs LaTeX file.
17    """
18
19 def format_p_value(x):
20     returns "{:.3g}".format(x) if x >= 1e-06 else "<1e-06"
21
22 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{Square residuals for the Height Model and the Age Model}
\label{table:residuals}
\begin{threeparttable}
\renewcommand{\TPTminimum}{\linewidth}
\makebox[\linewidth]{%
\begin{tabular}{lrr}
\toprule
& Residuals for Height Model & Residuals for Age Model \\
\midrule
\textbf{Mean} & 3.76 & 23.7 \\
\textbf{Std. deviation} & 6.12 & 49.9 \\
\bottomrule
\end{tabular}}
\begin{tablenotes}
\footnotesize
\item Table values are in cm2
\item \textbf{Residuals for Height Model}: Square residuals for the Height

```

Model, cm^2

```

\item \textbf{Residuals for Age Model}: Square residuals for the Age Model,  $\text{cm}^2$ 
\item \textbf{Mean}: Average of the square residuals
\item \textbf{Std. deviation}: Standard deviation of the square residuals
\end{tablenotes}
\end{threeparttable}
\end{table}

```

table_2.tex

```

\begin{table}[h]
\caption{Paired t-test results for residuals of the Height and Age Models}
\label{table:t_test}
\begin{threeparttable}
\renewcommand{\TPTminimum}{\linewidth}
\makebox[\linewidth]{%
\begin{tabular}{lrl}
\toprule
& T-Statistic & P-value \\
\midrule
\textbf{Test} & -12.5 &  $< 1e-06$  \\
\bottomrule
\end{tabular}}
\begin{tablenotes}
\footnotesize
\item \textbf{T-Statistic}: Test statistic from the paired t-test
\item \textbf{P-value}: P-value from the paired t-test
\end{tablenotes}
\end{threeparttable}
\end{table}

```