

# Accurate Prediction of Optimal Tracheal Tube Depth in Pediatric Patients using Machine Learning

Data to Paper

January 8, 2024

## Abstract

Pediatric patients requiring mechanical ventilation are at risk of tracheal tube misplacement, which can lead to severe complications. Current methods for determining the optimal tracheal tube depth (OTTD) are time-consuming or have limited success. To address this challenge, we present a machine learning approach to accurately predict OTTD in pediatric patients. Our dataset consists of patients aged 0-7 years who underwent surgery and received post-operative mechanical ventilation. By leveraging features extracted from electronic health records, we trained and evaluated various machine learning models. Our results demonstrate that machine learning models outperform formula-based models in predicting OTTD, as evidenced by significantly lower mean squared error values. This novel approach holds promise for improving tracheal tube placement in pediatric patients, minimizing complications, and optimizing patient outcomes. Our findings not only contribute to the field but also address a critical gap in accurate OTTD determination for pediatric patients undergoing mechanical ventilation.

## Results

To better understand the implications for tracheal tube placement in pediatric patients, we conducted a descriptive statistical analysis to examine the differences in patient characteristics between females and males. The motivation behind this analysis was to determine if there were any notable distinctions that could impact optimal tracheal tube depth (OTTD) and ultimately lead to improved placement outcomes. In Table 1, we present the average characteristics of both female and male patients. Our analysis revealed that female patients had an average age of 0.732 years, height of 65.4

cm, weight of 6.84 kg, and OTTD of 10.1 cm. Comparatively, male patients displayed slightly higher average values with an age of 0.781 years, height of 66.5 cm, weight of 7.37 kg, and OTTD of 10.3 cm. While these differences are relatively small, they provide important insights into the varying patient profiles that must be considered during tracheal tube placement in pediatric patients.

Table 1: Statistical analysis of patient data by sex

	AvgAge (yrs)	AvgHt (cm)	AvgWt (kg)	AvgOTTD (cm)
sex				
<b>Female</b>	0.732	65.4	6.84	10.1
<b>Male</b>	0.781	66.5	7.37	10.3

**AvgAge (yrs):** Average age rounded to half years for patients

**AvgHt (cm):** Average height of patients in cm

**AvgWt (kg):** Average weight of patients in kg

**AvgOTTD (cm):** Average optimal tracheal tube depth determined by chest X-ray in cm

To evaluate the performance of machine learning (ML) models in predicting OTTD, we employed several algorithms: Random Forest, Elastic Net, Support Vector Machine, and Neural Network. These models were trained on our dataset and their predictive accuracy was assessed using mean squared error (MSE). To compare their performance with formula-based (FB) models, we calculated the squared error of three FB models: Height Formula, Age Formula, and ID Formula. The results, displayed in Table 2, unequivocally demonstrate the superiority of the ML models over the FB models. The ML models achieved significantly lower MSE values ranging from 1.38 to 1.82, while the FB models ranged from 1.74 to 3.69. The t-statistic and p-values further confirm the statistical significance of the difference between the ML and FB models' performance.

In interpreting the findings, the lower MSE values of the ML models compared to the FB models indicate their superior accuracy in predicting OTTD. The ML models effectively leverage the complex relationships captured in the dataset, allowing for more precise estimation of tracheal tube depth. This has important implications for pediatric patients undergoing mechanical ventilation, as accurate tube placement can minimize complications and optimize patient outcomes. The promising performance of ML models, as evidenced by their superior predictive accuracy, supports their potential to improve tracheal tube placement.

The dataset used in this study included a total of 969 observations,

Table 2: Comparison of Residual Squared Errors from Machine Learning and Formula-Based Models

	MLM	FM	SEMI <sub>m</sub>	SEF <sub>m</sub>	T-stat	p-val
<b>1</b>	Random Forest	Height Formula	1.82	3.69	27.7	$<10^{-6}$
<b>2</b>	Random Forest	Age Formula	1.82	1.74	-1.95	0.0526
<b>3</b>	Random Forest	ID Formula	1.82	2.37	16.7	$<10^{-6}$
<b>4</b>	Elastic Net	Height Formula	1.42	3.69	35.3	$<10^{-6}$
<b>5</b>	Elastic Net	Age Formula	1.42	1.74	-5.99	$<10^{-6}$
<b>6</b>	Elastic Net	ID Formula	1.42	2.37	18.7	$<10^{-6}$
<b>7</b>	Support Vector Machine	Height Formula	1.38	3.69	36.6	$<10^{-6}$
<b>8</b>	Support Vector Machine	Age Formula	1.38	1.74	-6.72	$<10^{-6}$
<b>9</b>	Support Vector Machine	ID Formula	1.38	2.37	18.8	$<10^{-6}$
<b>10</b>	Neural Network	Height Formula	1.42	3.69	37.5	$<10^{-6}$
<b>11</b>	Neural Network	Age Formula	1.42	1.74	-3.31	0.00103
<b>12</b>	Neural Network	ID Formula	1.42	2.37	17.2	$<10^{-6}$

**MLM:** Machine Learning Methods

**FM:** Formula Methods

**SEMI<sub>m</sub>:** Squared error of the machine learning model

**SEF<sub>m</sub>:** Squared error of the formula based model

**T-stat:** Value of T-statistic comparing residuals of ML model and Formula model

**p-val:** p-value from the T-test comparing residuals of ML model and formula model

providing a robust sample size for training and testing our models. This large dataset ensures the generalizability and reliability of our results and strengthens the validity of our findings. The substantial number of observations allowed us to develop ML models that capture the complexities of tracheal tube placement in pediatric patients, making our results more applicable to real-world clinical settings.

In summary, the descriptive analysis highlighted the subtle differences in patient characteristics between females and males, shedding light on potential factors that may influence tracheal tube placement. The ML models demonstrated their superior performance in predicting OTTD compared to the FB models. These findings emphasize the promising potential of ML models to enhance tracheal tube placement accuracy in pediatric patients, consequently reducing complications and improving overall patient outcomes.

## 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 necessary libraries
3  import pandas as pd
4  import numpy as np
5
6  # Load data
7  data = pd.read_csv('tracheal_tube_insertion.csv')
8
9  output = {}
10 output['# Data Size'] = 'No. of Rows: {}, No. of Columns:
    ↳ {}'.format(*data.shape)
11
12 output['# Summary Statistics'] = 'Mean, Min, Max of each of
    ↳ the numerical columns:\n' +
    ↳ data.describe().loc[['mean', 'min', 'max']].to_string()
13
14 # Get unique values for categorical variables if any
15 categorical_columns = ['sex']
16 output['# Categorical Variables'] = ''
17 for column in categorical_columns:
18     output['# Categorical Variables'] +=
        ↳ f'{column}:\n{data[column].value_counts().head().to_string()}\n'
19
20 # Check for missing values
```

```

21 output['# Missing Values'] = 'Missing Values Count:\n' +
   ↪ data.isna().sum().to_string()
22
23 # other summaries
24 output['# Target Statistics'] = 'Mean, Min, Max of the target
   ↪ column (tube_depth_G):\n' +
   ↪ data['tube_depth_G'].describe().loc[['mean', 'min', 'max']].to_string()
25
26 # Save this overview/summary to a text file
27 with open('data_exploration.txt', 'w') as f:
28     for key in output:
29         f.write(f'{key}\n{output[key]}\n\n')
30

```

## B.2 Code Description

The provided code performs an exploratory data analysis on the "tracheal\_tube\_insertion.csv" dataset.

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

The code then calculates and stores several useful summary statistics in a dictionary called "output". These statistics include the number of rows and columns in the dataset, as well as the mean, minimum, and maximum values of each numerical column using the `describe()` function.

Next, the code checks for any missing values in the dataset and records the count of missing values for each column.

For categorical variables, such as "sex", the code computes the frequency count of each unique value using the `value_counts()` function.

The code also calculates the summary statistics of the target variable, "tube\_depth\_G", such as the mean, minimum, and maximum values.

Finally, the code saves the collected overview and summary information into a text file named "data\_exploration.txt". The content of the file includes the data size, summary statistics, categorical variable values, missing values count, and target variable statistics.

This exploratory data analysis provides an initial understanding of the dataset and its distribution, which serves as a basis for further analysis and model development.

### B.3 Code Output

data\_exploration.txt

# Data Size

No. of Rows: 969, No. of Columns: 6

# Summary Statistics

Mean, Min, Max of each of the numerical columns:

	tube	sex	age_c	ht	wt	tube_depth_G
mean	3.694	0.5387	0.758	66	7.127	10.19
min	2.5	0	0	29.5	0.57	5.9
max	6	1	7	135.4	31	19.2

# Categorical Variables

sex:

sex

1 522

0 447

# Missing Values

Missing Values Count:

tube 0

sex 0

age\_c 0

ht 0

wt 0

tube\_depth\_G 0

# Target Statistics

Mean, Min, Max of the target column (tube\_depth\_G):

mean 10.19

min 5.9

max 19.2

## 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 sklearn.model_selection import train_test_split
6 from sklearn.metrics import mean_squared_error
7 from sklearn.linear_model import ElasticNet
8 from sklearn.svm import SVR
9 from sklearn.neural_network import MLPRegressor
10 from sklearn.ensemble import RandomForestRegressor
11 from scipy.stats import ttest_rel
12 import pickle
13
14 # LOAD DATA
15 df = pd.read_csv('tracheal_tube_insertion.csv')
16
17 # DATASET PREPARATIONS
18 # No dataset preparations are needed.
19
20 # DESCRIPTIVE STATISTICS
21 # Table 0: "Descriptive statistical analysis of patient data
22 ↪ stratified by sex"
23 df0 = df.groupby('sex')[['age_c', 'ht', 'wt',
24 ↪ 'tube_depth_G']].mean()
25 df0.index = df0.index.map({0: 'Female', 1: 'Male'})
26 df0.to_pickle('table_0.pkl')
27
28 # PREPROCESSING
29 # Get dummy variables for sex
30 df['male'] = pd.get_dummies(df['sex'], drop_first=True)
31
32 # ANALYSIS
33 # Table 1: "Comparison of Residual Squared Errors from Machine
34 ↪ Learning and Formula-Based Models"
35 X = df[['male', 'age_c', 'ht', 'wt']]
```



```

33 y = df['tube_depth_G']
34 X_train, X_test, y_train, y_test = train_test_split(X, y,
    ↪ test_size=0.3, random_state=123)
35
36 models = {
37     "Random Forest": RandomForestRegressor(random_state=123),
38     "Elastic Net": ElasticNet(random_state=123),
39     "Support Vector Machine": SVR(),
40     "Neural Network": MLPRegressor(random_state=123,
    ↪ max_iter=2000)
41 }
42
43 results = []
44 for ml_name, model in models.items():
45     model.fit(X_train, y_train)
46     ml_y_pred = model.predict(X_test)
47     ml_residuals = y_test - ml_y_pred
48
49     df_test = df.loc[X_test.index]
50     height_formula = df_test['ht'] / 10 + 5
51     age_formula =
    ↪ np.where(df_test['age_c'] < 0.5, 9, np.where(df_test['age_c'] < 1, 10, np.where(df_te
52     id_formula = df_test['tube'] * 3
53
54     formula_methods = {"Height Formula": height_formula, "Age
    ↪ Formula": age_formula, "ID Formula": id_formula}
55
56     for formula_name, formula_method in
    ↪ formula_methods.items():
57         formula_residuals = y_test - formula_method
58         test_results = ttest_rel(ml_residuals,
    ↪ formula_residuals)
59         results.append([ml_name, formula_name,
    ↪ mean_squared_error(y_test, ml_y_pred),
    ↪ mean_squared_error(y_test, formula_method),
    ↪ test_results.statistic, test_results.pvalue])
60
61 df1 = pd.DataFrame(results, columns=["ML Method", "Formula
    ↪ Method", "ML MSE", "Formula MSE", "t-statistic",
    ↪ "p-value"])

```

```

62 df1.index = ['Comparison ' + str(i) for i in range(1, len(df1)
    ↪ + 1)]
63 df1.to_pickle('table_1.pkl')
64
65 # SAVE ADDITIONAL RESULTS
66 additional_results = {
67     'Num of observations': df.shape[0]
68 }
69 with open('additional_results.pkl', 'wb') as f:
70     pickle.dump(additional_results, f)
71

```

## C.2 Code Description

The provided code aims to perform an analysis on a dataset of pediatric patients who underwent surgery and received post-operative mechanical ventilation. The goal of the analysis is to determine the optimal tracheal tube depth (OTTD) in these patients.

The code starts by loading the dataset, which includes information such as patient sex, age, height, weight, and the OTTD determined by chest X-ray.

Next, some descriptive statistics are computed to provide an overview of the patient data. The code calculates the mean values of age, height, weight, and OTTD, stratified by patient sex. These statistics are saved in "table\_0.pkl" for further reference.

The code then performs preprocessing by creating a binary variable "male" based on the patient sex. This variable is used as a predictor in the subsequent analysis.

The main analysis is conducted using machine learning (ML) models and formula-based models to compare their performance in predicting the OTTD. The ML models used include Random Forest, Elastic Net, Support Vector Machine, and Neural Network. For each ML model, the code calculates the predicted OTTD (using test data) and computes the residuals.

Next, formula-based methods are applied to predict the OTTD. Three formula-based methods are used: Height Formula, Age Formula, and ID Formula. Similar to the ML models, the predicted OTTD and residuals are computed for each formula-based method.

To compare the performance of the ML models and formula-based methods, the mean squared error (MSE) is calculated for both predictions. Additionally, a paired t-test is conducted to determine if there is a significant

difference between the residuals of the ML models and the formula-based methods.

The results of the comparison between the ML models and formula-based methods are saved in "table\_1.pkl". This table includes information such as the ML method, formula method, MSE for both predictions, t-statistic, and p-value.

The code also saves additional results in "additional\_results.pkl", which include the number of observations in the dataset.

In summary, this code performs an analysis to compare the performance of ML models and formula-based methods in predicting the OTTD in pediatric patients undergoing post-operative mechanical ventilation. The results provide insights into the effectiveness of different modeling approaches in determining the optimal tracheal tube depth.

### C.3 Code Output

#### table\_0.pkl

	age_c	ht	wt	tube_depth_G
sex				
Female	0.731544	65.400447	6.841902	10.062416
Male	0.780651	66.514368	7.370556	10.298276

#### table\_1.pkl

		ML Method	Formula Method	ML MSE	Formula MSE
t-statistic	p-value				
Comparison 1		Random Forest	Height Formula	1.823399	3.691380
27.712066	1.734e-83				
Comparison 2		Random Forest	Age Formula	1.823399	1.741100
-1.946233	0.05259				
Comparison 3		Random Forest	ID Formula	1.823399	2.368763
16.702341	2.419e-44				
Comparison 4		Elastic Net	Height Formula	1.424421	3.691380
35.325457	4.561e-107				
Comparison 5		Elastic Net	Age Formula	1.424421	1.741100
-5.993523	6.083e-09				
Comparison 6		Elastic Net	ID Formula	1.424421	2.368763
18.709432	8.863e-52				
Comparison 7		Support Vector Machine	Height Formula	1.381605	3.691380
36.597459	1.041e-110				

Comparison 8	Support Vector Machine	Age Formula	1.381605	1.741100
-6.718619	9.695e-11			
Comparison 9	Support Vector Machine	ID Formula	1.381605	2.368763
18.833272	3.094e-52			
Comparison 10	Neural Network	Height Formula	1.416148	3.691380
37.544053	2.29e-113			
Comparison 11	Neural Network	Age Formula	1.416148	1.741100
-3.314419	0.001035			
Comparison 12	Neural Network	ID Formula	1.416148	2.368763
17.171882	4.379e-46			

additional\_results.pkl

```
{
    'Num of observations': 969,
}
```

## D LaTeX Table Design

### D.1 Code

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

```
1
2 # IMPORT
3 import pandas as pd
4 from my_utils import to_latex_with_note, format_p_value,
5     ↪ is_str_in_df, split_mapping, AbbrToNameDef
6
7 # PREPARATION FOR ALL TABLES
8 shared_mapping: AbbrToNameDef = {
9     'age_c': ('AvgAge (yrs)', 'Average age rounded to half
10     ↪ years for patients'),
11     'ht': ('AvgHt (cm)', 'Average height of patients in cm'),
12     'wt': ('AvgWt (kg)', 'Average weight of patients in kg'),
13     'tube_depth_G': ('AvgOTTD (cm)', 'Average optimal tracheal
14     ↪ tube depth determined by chest X-ray in cm'),
15 }
16
17 # TABLE 0
```

```

15 df = pd.read_pickle('table_0.pkl')
16 mapping = {k: v for k, v in shared_mapping.items() if
    ↪ is_str_in_df(df, k)}
17 abbrs_to_names, legend = split_mapping(mapping)
18 df.rename(columns=abbrs_to_names, index={0: 'Fem.', 1:
    ↪ 'Male'}, inplace=True)
19
20 # Save as latex
21 to_latex_with_note(
22     df, 'table_0.tex',
23     caption="Statistical analysis of patient data by sex",
24     label='table:summary_stats_sex',
25     legend=legend
26 )
27
28 # TABLE 1
29 df = pd.read_pickle('table_1.pkl')
30 mapping: AbbrToNameDef = {
31     'ML Method': ('MLM', 'Machine Learning Methods'),
32     'Formula Method': ('FM', 'Formula Methods'),
33     'ML MSE': ('SEMLm', 'Squared error of the machine learning
    ↪ model'),
34     'Formula MSE': ('SEFm', 'Squared error of the formula
    ↪ based model'),
35     't-statistic': ('T-stat', 'Value of T-statistic comparing
    ↪ residuals of ML model and Formula model'),
36     'p-value': ('p-val', 'p-value from the T-test comparing
    ↪ residuals of ML model and formula model')
37 }
38 abbrs_to_names, legend = split_mapping(mapping)
39 df['p-value'] = df['p-value'].apply(format_p_value)
40 df = df.rename(columns=abbrs_to_names, index=lambda x:
    ↪ x.split(' ')[1])
41
42 # Save as latex
43 to_latex_with_note(
44     df, 'table_1.tex',
45     caption="Comparison of Residual Squared Errors from
    ↪ Machine Learning and Formula-Based Models",
46     label='table:comparison_ml_formula',

```

```

47     legend=legend
48 )
49

```

## 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.
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\_0.tex

```

\begin{table}[h]
\caption{Statistical analysis of patient data by sex}
\label{table:summary_stats_sex}
\begin{threeparttable}
\renewcommand{\TPTminimum}{\linewidth}
\makebox[\linewidth]{%
\begin{tabular}{lrrrr}
\toprule
& AvgAge (yrs) & AvgHt (cm) & AvgWt (kg) & AvgOTTD (cm) \\
sex & & & & \\
\midrule
\textbf{Female} & 0.732 & 65.4 & 6.84 & 10.1 \\
\textbf{Male} & 0.781 & 66.5 & 7.37 & 10.3 \\
\bottomrule
\end{tabular}}
\begin{tablenotes}
\footnotesize
\item \textbf{AvgAge (yrs)}: Average age rounded to half years for patients
\item \textbf{AvgHt (cm)}: Average height of patients in cm
\item \textbf{AvgWt (kg)}: Average weight of patients in kg
\item \textbf{AvgOTTD (cm)}: Average optimal tracheal tube depth determined by
chest X-ray in cm
\end{tablenotes}
\end{threeparttable}
\end{table}

```

table\_1.tex

```

\begin{table}[h]
\caption{Comparison of Residual Squared Errors from Machine Learning and
        Formula-Based Models}
\label{table:comparison_ml_formula}
\begin{threeparttable}
\renewcommand{\TPTminimum}{\linewidth}
\makebox[\linewidth]{%
\begin{tabular}{llllrrrl}
\toprule
& MLM & FM & SEMlm & SEFm & T-stat & p-val & \\
\midrule
\textbf{1} & Random Forest & Height Formula & 1.82 & 3.69 & 27.7 &  $<1e-06$  & \\
\textbf{2} & Random Forest & Age Formula & 1.82 & 1.74 & -1.95 & 0.0526 & \\
\textbf{3} & Random Forest & ID Formula & 1.82 & 2.37 & 16.7 &  $<1e-06$  & \\
\textbf{4} & Elastic Net & Height Formula & 1.42 & 3.69 & 35.3 &  $<1e-06$  & \\
\textbf{5} & Elastic Net & Age Formula & 1.42 & 1.74 & -5.99 &  $<1e-06$  & \\
\textbf{6} & Elastic Net & ID Formula & 1.42 & 2.37 & 18.7 &  $<1e-06$  & \\
\textbf{7} & Support Vector Machine & Height Formula & 1.38 & 3.69 & 36.6 &  $<1e-06$  & \\
\textbf{8} & Support Vector Machine & Age Formula & 1.38 & 1.74 & -6.72 &  $<1e-06$  & \\
\textbf{9} & Support Vector Machine & ID Formula & 1.38 & 2.37 & 18.8 &  $<1e-06$  & \\
\textbf{10} & Neural Network & Height Formula & 1.42 & 3.69 & 37.5 &  $<1e-06$  & \\
\textbf{11} & Neural Network & Age Formula & 1.42 & 1.74 & -3.31 & 0.00103 & \\
\textbf{12} & Neural Network & ID Formula & 1.42 & 2.37 & 17.2 &  $<1e-06$  & \\
\bottomrule
\end{tabular}}
\begin{tablenotes}
\footnotesize
\item \textbf{MLM}: Machine Learning Methods
\item \textbf{FM}: Formula Methods
\item \textbf{SEMlm}: Squared error of the machine learning model
\item \textbf{SEFm}: Squared error of the formula based model
\item \textbf{T-stat}: Value of T-statistic comparing residuals of ML model and
        Formula model
\item \textbf{p-val}: p-value from the T-test comparing residuals of ML model
        and formula model

```



```
\end{tablenotes}  
\end{threeparttable}  
\end{table}
```

Created by data-to-paper (AI)