

# Using Sensor Data and ML to Estimate Room Occupancy

## Final Results and Project Report

Jeffrey Fitzpatrick

500728133

Supervisor: Ceni Babaoglu

December 2, 2024



## Table of Contents

---

<b>Introduction .....</b>	<b>3</b>
Problem.....	3
Data .....	3
Techniques and Tools.....	3
<b>Research Questions.....</b>	<b>4</b>
<b>Contribution of Work Compared to Past Research.....</b>	<b>4</b>
<b>GitHub Repository.....</b>	<b>4</b>
<b>Applied Methodology and Study Design.....</b>	<b>5</b>
<b>Initial Analysis .....</b>	<b>6</b>
Univariate Analysis .....	6
Data Dictionary .....	6
Dependent (target) Variable.....	7
Summary Statistics .....	7
Frequency of Categorical Variables .....	8
Bivariate Analysis.....	9
Pairwise Visualizations.....	9
Time Series Analysis .....	10
Correlation Analysis .....	12
<b>Exploratory Analysis.....</b>	<b>13</b>
<b>Dimensionality Reduction .....</b>	<b>15</b>
<b>Experimental Design.....</b>	<b>17</b>
<b>Multinomial Logistic Regression .....</b>	<b>17</b>
All Features.....	17
PCA Features .....	19
<b>Random Forest.....</b>	<b>19</b>

All Features.....	19
PCA Features .....	21
<b>Support Vector Machine .....</b>	<b>22</b>
All Features.....	22
PCA Features .....	23
<b>Findings .....</b>	<b>24</b>
Summary of Model Results .....	24
Summary of Sensor Data.....	24
<b>Limitations and Future Work.....</b>	<b>25</b>
<b>References .....</b>	<b>26</b>

# Introduction

---

## Problem

To provide comfort for occupants, commercial buildings rely on heating, ventilation, and air conditioning (HVAC) systems and lighting systems. Large amounts of energy are wasted, especially during non-working hours (Masoso and Grobler, 2010).

To optimize these systems, a research paper (Singh et al., 2018) described an experiment for accurately estimating the number of occupants in a room using “non-intrusive” environment sensors and machine learning (ML) models. Multiple sensor nodes were placed throughout a 6m by 4.6m test room in a wireless sensor network (WSN). Low-cost sensors were deployed at each of the four desks to measure temperature, light, and sound. A carbon dioxide (CO<sub>2</sub>) sensor was deployed in the middle of the room to provide the most accurate reading. Two motion detection sensors were deployed on the ceiling, above the door and large window. Because of privacy concerns, video-based systems are not considered appropriate for detecting occupancy.

Some previous studies focused on using occupancy detection (i.e., determining whether a room is occupied or not) to save energy. On the other hand, the goal of ML occupancy estimation research is to design adaptive systems that can detect the exact number of occupants, resulting in additional energy savings and improved comfort for occupants.

## Data

For my project, I chose the Room Occupancy Estimation dataset, available from the [UC Irvine Machine Learning Repository](https://archive.ics.uci.edu/dataset/864/room+occupancy+estimation). The Room Occupancy Estimation dataset can be downloaded from <https://archive.ics.uci.edu/dataset/864/room+occupancy+estimation>.

The dataset contains over 10000 data points and 16 features. Each feature represents data (temperature, light, sound, motion, or CO<sub>2</sub>) from a particular sensor. Measurements were recorded over several days in 30 second intervals. The actual occupancy was established by having participants register and record the exact time each time they entered or left the room.

## Techniques and Tools

To solve the stated problem, I implemented supervised learning algorithms including Multinomial Logistic Regression, Random Forest, and Support Vector Machine (SVM). In addition, I investigated how an unsupervised learning algorithm named Principal Component Analysis (PCA) can be used for dimensionality reduction.

For this project, I used R for the initial data analysis and Python for exploratory analysis, dimensionality reduction, and implementation of the machine learning algorithms. All the supervised and unsupervised learning algorithms were implemented using the scikit-learn open-source library. To evaluate the models, I used multiple performance metrics including accuracy, confusion matrices, and F1-score.

## Research Questions

---

For my project, I chose the theme of Classification for building predictive models.

To reduce energy consumption in buildings, I investigated these research questions:

- Which of the implemented supervised learning techniques perform the best in predicting occupancy?
- Which types of sensor data (temperature, light, sound, motion, CO<sub>2</sub>) show the most promising results?
- Based on the research, what alternative types of sensor data could be used for ML-based occupancy estimation?

## Contribution of Work Compared to Past Research

---

The introductory paper for my chosen dataset (Singh et al., 2018) is the basis for this project. In that study, the authors deployed multiple light, temperature, sound, and CO<sub>2</sub> sensors in a test room as described in the [Introduction](#). To estimate the number of occupants in the room, four classification models were used: Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis (QDA), Support Vector Machine (SVM), and Random Forest. For SVM, the results were evaluated with both a linear and radial basis function (RBF) kernel. With all features included, SVM with RBF kernel performed the best, with an accuracy of 98% and an F1-score of 95%. After employing Principal Component Analysis (PCA), the authors concluded that an accuracy of 92% and F1-score of 72% was achievable with only four components.

In (Mao et al., 2023), the authors adapted a predictive framework for room occupancy using the same dataset. Several classification methods were used: Logistic Regression, Linear Discriminant Analysis (LDA), Support Vector Machine (SVM), Multi-layer Perceptron (MLP), LightGBM, XGBoost, and Random Forest. The models were evaluated using balanced accuracy, F1-score, and Area Under ROC Curve (AUC). Among the methods, Random Forest performed the best in all three metrics. In the Random Forest model, the light values from sensor 1 and sensor 2 had the largest impact in predicting room occupancy. The authors claimed their results improved on the performance of the original paper, with a balanced accuracy and AUC above 99% and F1-score above 98%.

The aim of this project is to compare and, in some cases, replicate the results of these earlier studies. For classification models, I've chosen to focus on Logistic Regression, Random Forest, and Support Vector Machine (SVM).

## GitHub Repository

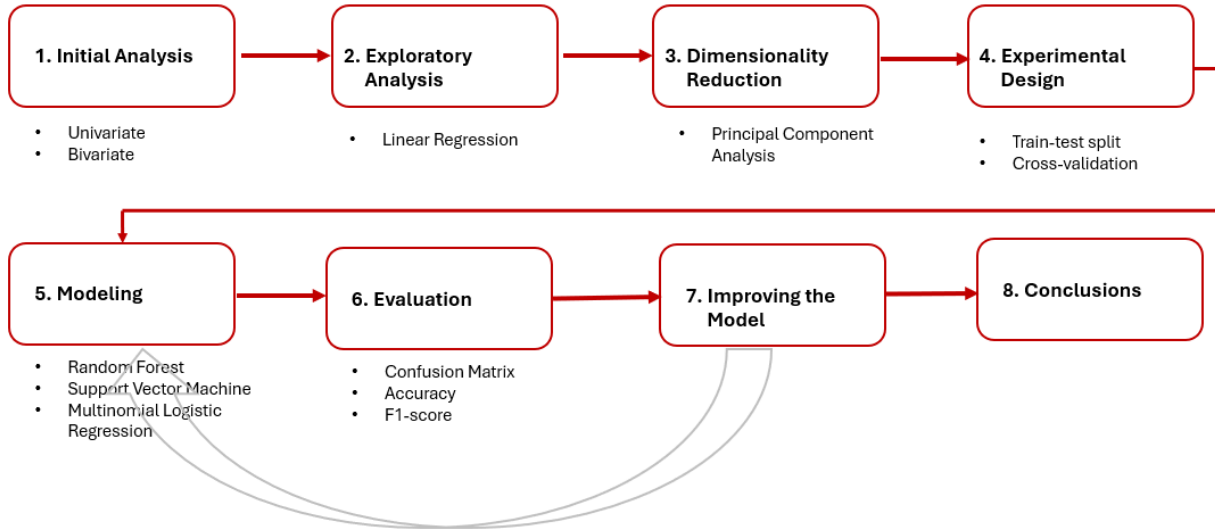
---

For this project, I'm using the following repository in GitHub:

<https://github.com/jeffreyfitzpatrick/Big-Data-Analytics-Capstone-Project>

## Applied Methodology and Study Design

This flowchart shows the steps of the data analysis process I followed for this project:



### 1. Initial Analysis

In the univariate analysis, I included a detailed data dictionary, summary statistics for the numeric attributes, and bar plots showing the frequency of the categorical variables.

In the bivariate analysis, I included a scatter plot matrix of the continuous variables, time series analysis, and correlation analysis.

### 2. Exploratory Analysis

As part of exploratory analysis, I used linear regression as a baseline model.

### 3. Dimensionality Reduction

I investigated how an unsupervised learning algorithm named Principal Component Analysis (PCA) can be used for dimensionality reduction.

### 4. Experimental Design

I split the data into training and test sets and investigated k-fold cross validation to evaluate the stability of the models.

### 5. Modeling

I implemented three classification algorithms: Multinomial Logistic Regression, Random Forest, and Support Vector Machine (SVM).

### 6. Evaluation

I evaluated the models using accuracy, confusion matrices, and F1-score.

## 7. Improving the Model

Improving the model is an iterative process.

## 8. Conclusions

In this final report, I compared my results with the two studies that used the same dataset.

# Initial Analysis

The data set contains 10129 data points and 16 features. Each feature represents data (temperature, light, sound, motion, or CO<sub>2</sub>) from a particular sensor. Measurements were recorded over several days in 30 second intervals.

There are no missing values in the dataset.

## Univariate Analysis

### Data Dictionary

This table describes the variables in the Room Occupancy Estimation data set:

Field Name	Data Type	Description
Date	Date	Date of observation in YYYY/MM/DD
Time	Date	Time of observation in HH:MM:SS
S1_Temp	Continuous	Temperature reading from sensor 1 in degrees Celsius
S2_Temp	Continuous	Temperature reading from sensor 2 in degrees Celsius
S3_Temp	Continuous	Temperature reading from sensor 3 in degrees Celsius
S4_Temp	Continuous	Temperature reading from sensor 4 in degrees Celsius
S1_Light	Integer	Light reading from sensor 1 in lux
S2_Light	Integer	Light reading from sensor 2 in lux
S3_Light	Integer	Light reading from sensor 3 in lux
S4_Light	Integer	Light reading from sensor 4 in lux
S1_Sound	Continuous	Sound reading from sensor 1 in volts (amplifier output read by ADC)

Field Name	Data Type	Description
S2_Sound	Continuous	Sound reading from sensor 2 in volts (amplifier output read by ADC)
S3_Sound	Continuous	Sound reading from sensor 3 in volts (amplifier output read by ADC)
S4_Sound	Continuous	Sound reading from sensor 4 in volts (amplifier output read by ADC)
S5_CO2	Integer	CO <sub>2</sub> reading from sensor 5 in ppm
S5_CO2_Slope	Continuous	Derived slope of CO <sub>2</sub> values taken in a sliding window <b>Note:</b> The slope was estimated using linear regression.
S6_PIR	Binary	Binary value conveying motion detection from passive infrared (PIR) sensor 6: <b>0:</b> No motion events detected in 30 second frame <b>1:</b> At least one motion event detected in 30 second frame
S7_PIR	Binary	Binary value conveying motion detection from passive infrared (PIR) sensor 7: <b>0:</b> No motion events detected in 30 second frame <b>1:</b> At least one motion event detected in 30 second frame
Room_Occupancy_Count	Integer	Number of occupants in the room at one time (ground truth)

### Dependent (target) Variable

The dependent (target) variable is Room\_Occupancy\_Count.

### Summary Statistics

This table shows the summary statistics of the numeric attributes in the data set:

Variable	Minimum	Q1	Median	Mean	Q3	Maximum
S1_Temp	24.94	25.19	25.38	25.45	25.63	26.38
S2_Temp	24.75	25.19	25.38	25.55	25.63	29.00

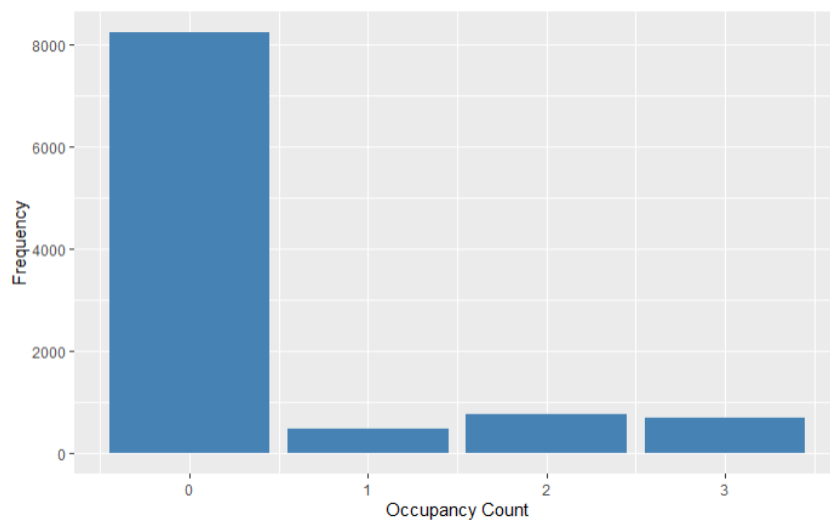


Variable	Minimum	Q1	Median	Mean	Q3	Maximum
S3_Temp	24.44	24.69	24.94	25.06	25.38	26.19
S4_Temp	24.94	25.44	25.75	25.75	26.00	26.56
S1_Light	0.00	0.00	0.00	25.45	12.00	165.00
S2_Light	0.00	0.00	0.00	26.02	14.00	258.00
S3_Light	0.00	0.00	0.00	34.25	50.00	280.00
S4_Light	0.00	0.00	0.00	13.22	22.00	74.00
S1_Sound	0.06	0.07	0.08	0.1682	0.08	3.88
S2_Sound	0.04	0.05	0.05	0.1201	0.06	3.44
S3_Sound	0.04	0.06	0.06	0.1581	0.07	3.67
S4_Sound	0.05	0.06	0.08	0.1038	0.1	3.4
S5_CO2	345	355	360	460.9	465	1270
S5_CO2_Slope	-6.29615	-0.04615	0	-0.00483	0	8.98077

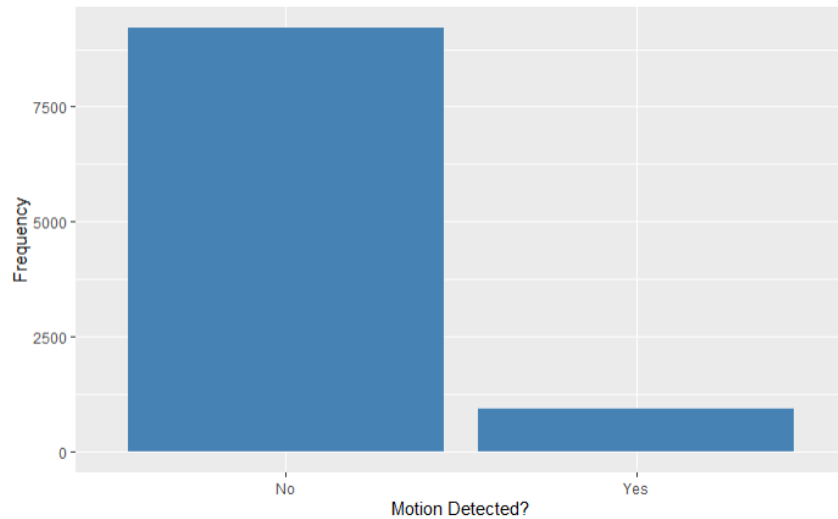
### Frequency of Categorical Variables

In about 80% of the data points, the room was unoccupied.

This chart shows the frequency of the Room\_Occupancy\_Count (target) variable:



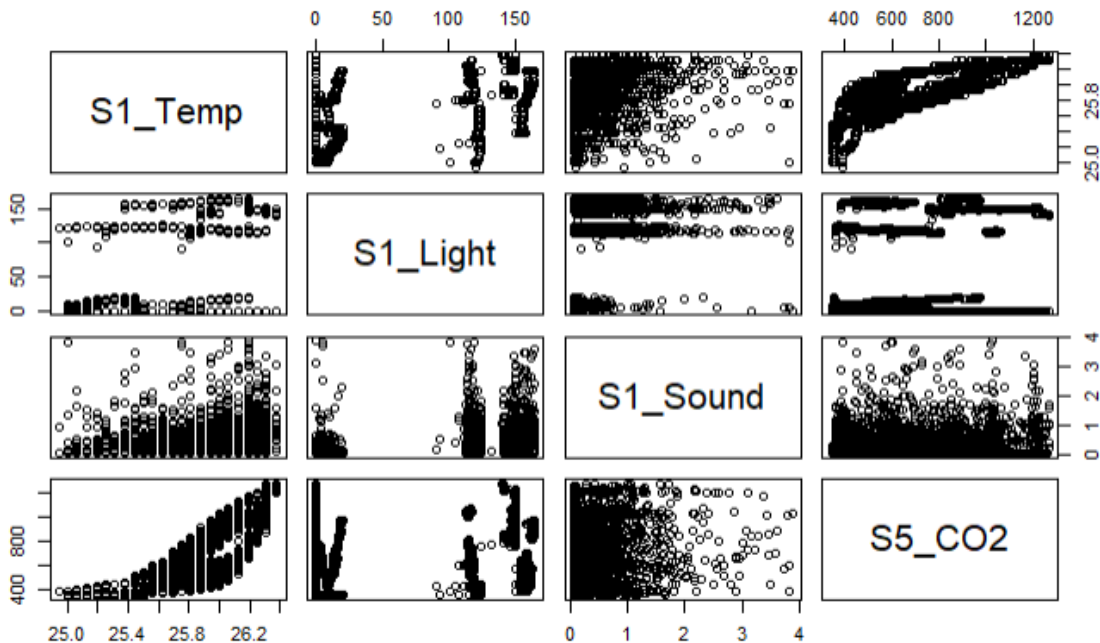
This bar chart shows the frequency of the S6\_PIR variable used to detect motion:



## Bivariate Analysis

### Pairwise Visualizations

This chart shows the scatter plot matrix of the temperature (S1), light (S1), sound (S1), and CO<sub>2</sub> (S5) values:

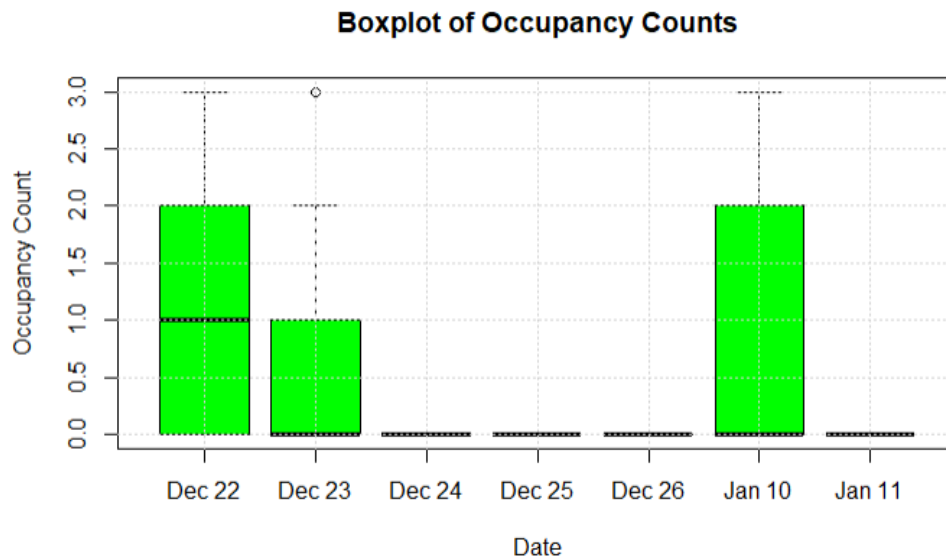


## Time Series Analysis

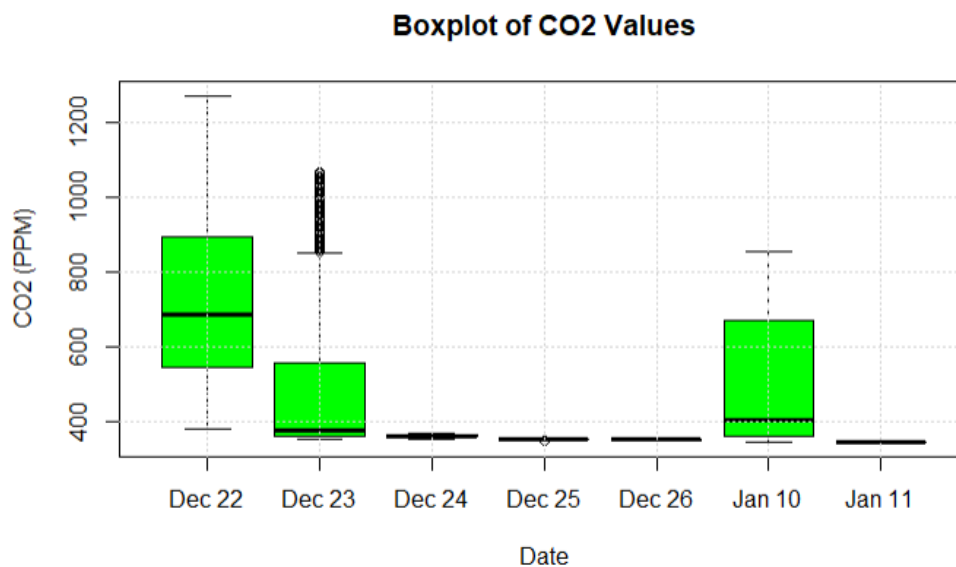
Based on the time series analysis, I observed the following:

- There were occupants only on December 22, December 23, and January 10.
- Significant CO<sub>2</sub> values were only detected on days with occupants.
- The highest temperatures occurred on the days with occupants.
- The highest light values occurred on days with occupants. Notably, light was detected on Christmas, indicating that lights were turned on during part of that day.
- The highest sound values occurred on days with occupants.

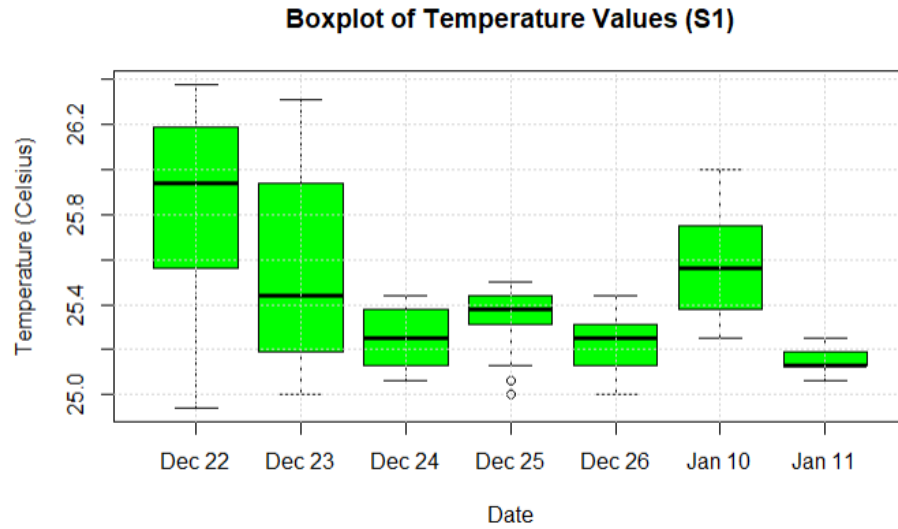
This boxplot shows the distribution of room occupancy for each day:



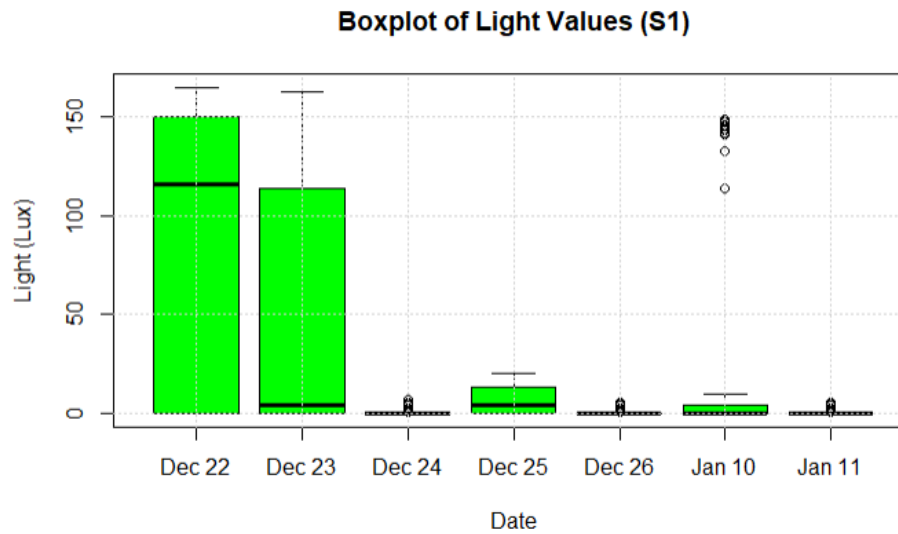
This boxplot shows the distribution of CO<sub>2</sub> values for each day:



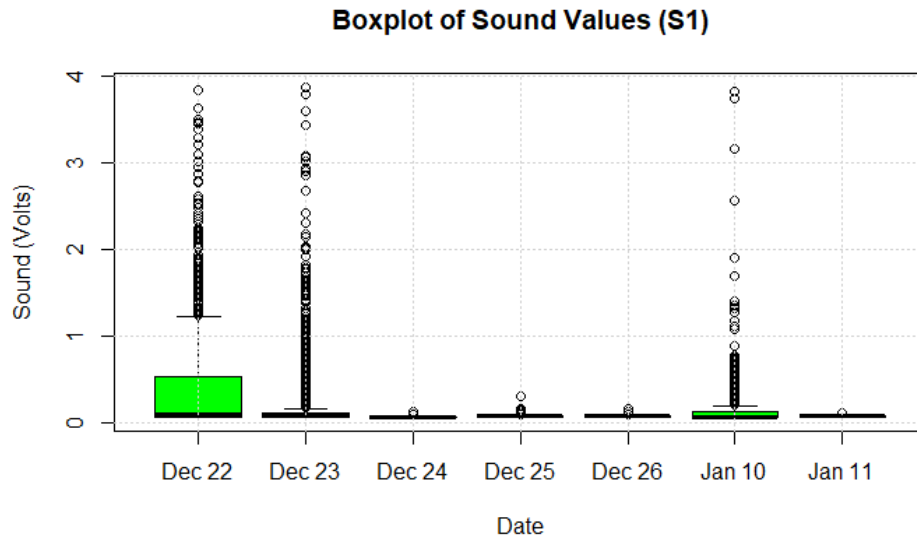
This boxplot shows the distribution of temperature values (sensor 1) for each day:



This boxplot shows the distribution of light values (sensor 1) for each day:



This boxplot shows the distribution of sound values (sensor 1) for each day:



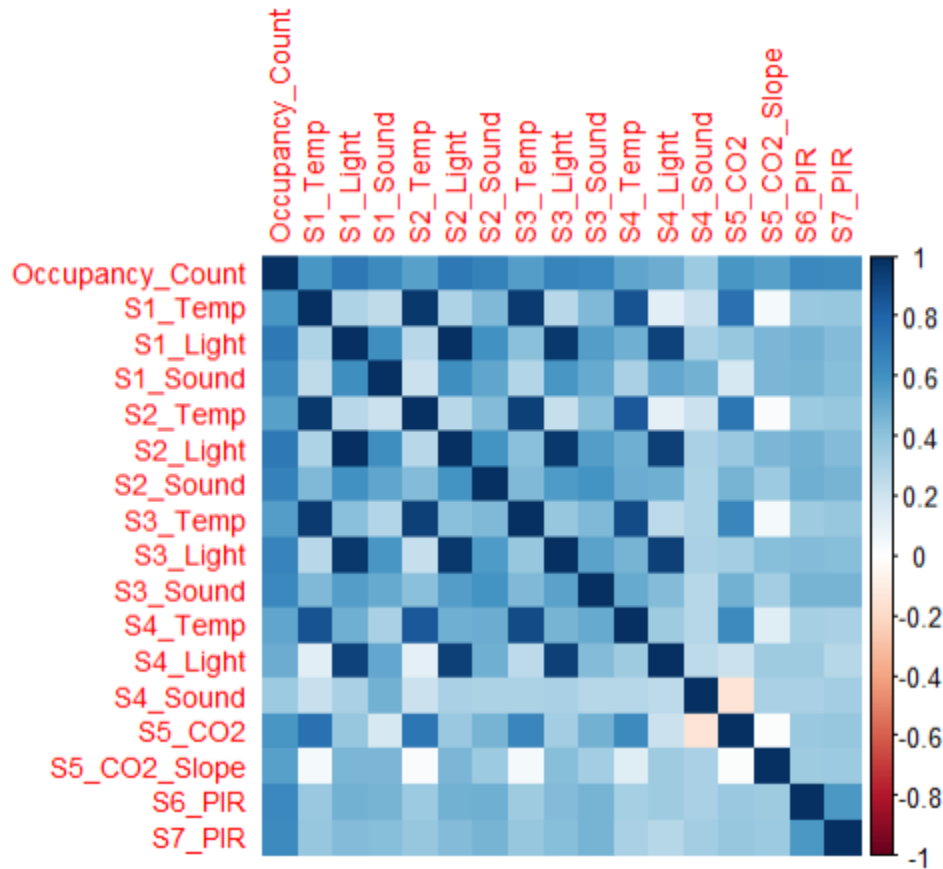
### Correlation Analysis

In this analysis, Spearman correlation was used as the attributes do not have a normal distribution.

Based on the correlation analysis, I observed the following:

- There is a moderate correlation between room occupancy and temperature variables.
- There is a strong correlation of temperature values between sensors.
- There is a moderate correlation between room occupancy and light variables.
- There is a strong correlation of light values between sensors, indicating that the lights were generally turned on at the same time or turned off.
- There is a moderate correlation between room occupancy and sound variables, except for sensor 4.
- There is a relatively weak correlation of sound values between sensors, indicating that the sensors could distinguish sounds of occupants in different parts of the room.
- There is a moderate correlation between room occupancy and CO<sub>2</sub> variables.
- There is a moderate correlation between room occupancy and motion variables.
- There is no correlation between CO<sub>2</sub> and the derived CO<sub>2</sub> slope values.

This chart shows a plot of the correlation matrix:



## Exploratory Analysis

For exploratory analysis, I fitted various linear regression models to predict room occupancy based on the sensor data.

In the following table, the R-squared value represents the proportion of variance in the dependent variable that is explained by the independent variables. The higher the R-squared value, the better the fit of the model. The intercept term represents the predicted value of the dependent variable when all independent variables are zero. P-values associated with each coefficient indicate the significance of the corresponding variable. P-values below 0.05 suggest that the corresponding variable is statistically significant in predicting the target variable (Room\_Occupancy\_Count).

Model to predict room occupancy	R-squared	Intercept	Coefficients (P-value)
Temperature data (S1_Temp, S2_Temp, S3_Temp, S4_Temp)	0.548	-37.4029	S1_Temp: 1.6245 (0.000) S2_Temp: 0.4826 (0.000) S3_Temp: 0.1581 (0.001) S4_Temp: -0.7703 (0.000)
Light data (S1_Light, S2_Light, S3_Light, S4_Light)	0.792	0.0410	S1_Light: 0.0076 (0.000) S2_Light: 0.0032 (0.000) S3_Light: 0.0058 (0.000) S4_Light: -0.0091 (0.000)
Sound data (S1_Sound, S2_Sound, S3_Sound, S4_Sound)	0.444	0.0900	S1_Sound: 0.8700 (0.000) S2_Sound: 0.9411 (0.000) S3_Sound: 0.5396 (0.000) S4_Sound: -0.3559 (0.000)
CO <sub>2</sub> data (S5_CO2, S5_CO2_Slope)	0.746	-0.8794	S5_CO2: 0.0028 (0.000) S5_CO2_Slope: 0.4281 (0.000)
CO <sub>2</sub> slope data (S5_CO2_Slope)	0.361	0.4008	S5_CO2_Slope: 0.4611 (0.000)
Motion data (S6_PIR, S7_PIR)	0.401	0.2181	SS_PIR: 6.559e+11 ( <b>0.265</b> ) S6_PIR: -6.559e+11 ( <b>0.265</b> )
All features	0.894	-8.0051	S1_Temp: 0.1335 (0.003) S2_Temp: 0.1166 (0.000) S3_Temp: 0.7861 (0.000) S4_Temp: -0.6959 (0.000) S1_Light: 0.0054 (0.000) S2_Light: 0.0009 (0.000) S3_Light: 0.0026 (0.000) S4_Light: -0.0040 (0.000) S1_Sound: 0.1009 (0.000) S2_Sound: 0.1957 (0.000) S3_Sound: -0.0763 (0.000) S4_Sound: -0.3485 (0.000) S5_CO2: 3.42e-05 ( <b>0.364</b> ) S5_CO2_Slope: 0.1896 (0.000) S6_PIR: 0.1800 (0.000) S7_PIR: 0.4139 (0.000)

Regarding p-values and coefficient values, see [Limitations and Future Work](#).

Using all sensor data provided the best model fit, with an R-squared value of 89.4%. The results suggest that the S5\_CO2 value can be dropped since it's p-value is greater than 0.05.

Based on the remaining models, light data had the highest R-squared value at 79.2% followed by CO2 data at 74.6%. This is consistent with the results of the introductory paper (Singh et al., 2018), which found that light features performed the best overall and that the CO<sub>2</sub> features had promising results.

For the linear regression model containing only S5\_CO2\_Slope, I was able to duplicate the results of the Mao et al. (2023) study, which found:

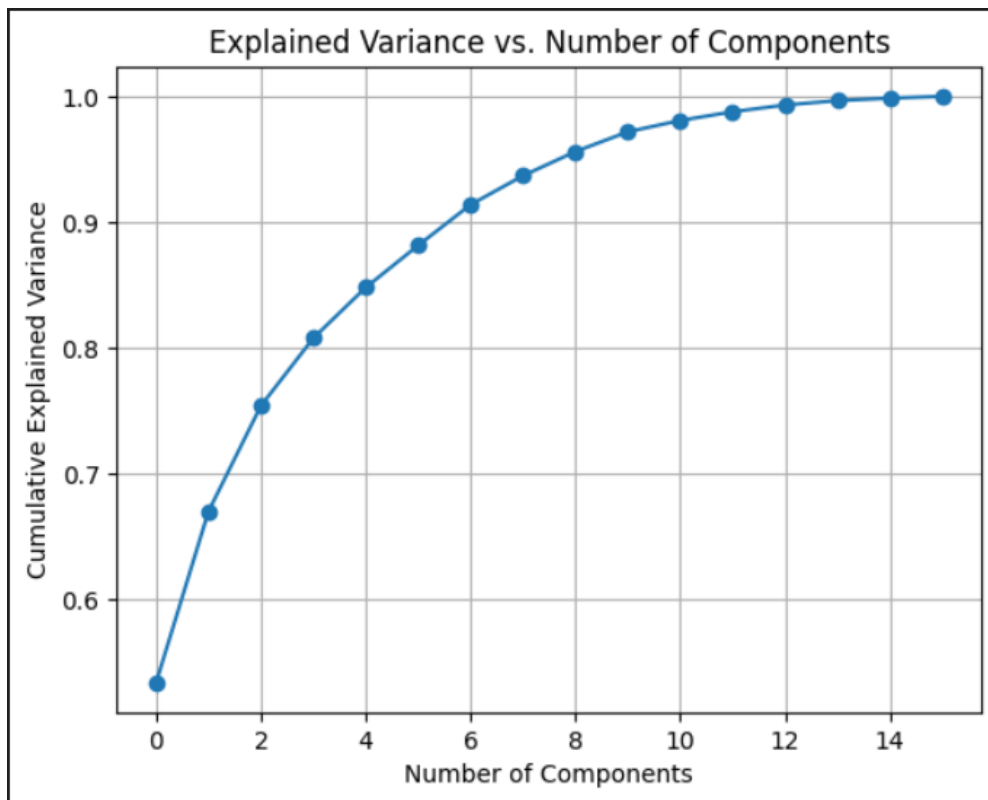
- An intercept value of approximately 0.4008, indicating the expected value of Room\_Occupancy\_Count when S5\_CO2\_Slope is zero.
- Positive slope of approximately 0.4611, indicating that for each unit increase in S5\_CO2\_Slope we can anticipate an increase of approximately 0.4611 in Room\_Occupancy\_Count.

For the linear regression model containing only motion data, p-values of 0.265 indicate that the motion data by itself is not statistically significant in predicting the target variable (Room\_Occupancy\_Count).

## Dimensionality Reduction

For dimensionality reduction, I used Principal Component Analysis (PCA). Since PCA is sensitive to the scale of the features, I first standardized the dataset so that all the numerical features have a mean of 0 and a variance of 1. This step ensures that features that have a larger variance because of scale do not dominate the principal components.

To determine the optimal number of components, I created a cumulative explained variance graph:



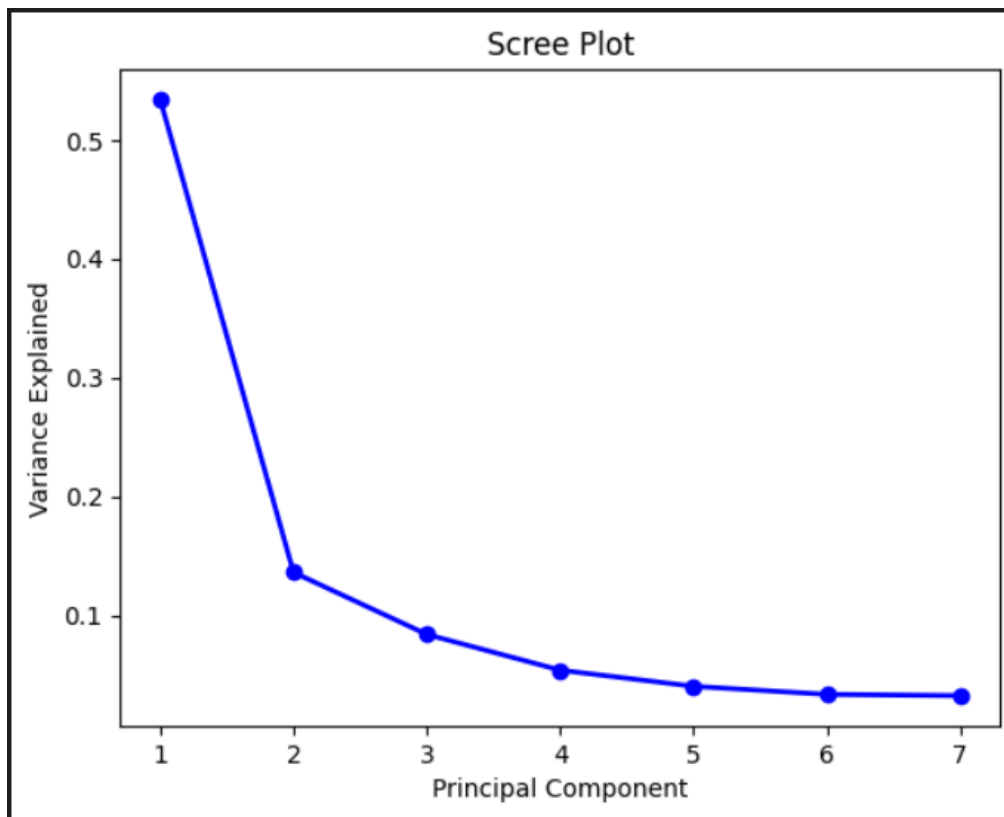


To capture at least 90% of the variance, I chose to select 7 components. This reduced the number of dimensions from 16 in the original dataset to 7 in the PCA dataset.

After applying PCA with 7 components, I calculated the percentage of total variance explained by each principal component:

- The first principal component explained **53.42%** of the total variation in the dataset.
- The second principal component explained **13.61%** of the total variation in the dataset.
- The third principal component explained **8.39%** of the total variation in the dataset.
- The fourth principal component explained **5.38%** of the total variation in the dataset.
- The fifth principal component explained **4.02%** of the total variation in the dataset.
- The sixth principal component explained **3.34%** of the total variation in the dataset.
- The seventh principal component explained **3.23%** of the total variation in the dataset.

This can be visualized using a scree plot:



Finally, I analyzed the feature contributions (loadings) for the first two principal components.

For the first component, S1\_Light, S3\_Light, S1\_Temp, S2\_Temp, and S3\_Temp had the largest contributions.

For the second component, S5\_CO2\_Slope, S4\_Sound, S2\_Sound, S1\_Sound, and S3\_Sound had the largest contributions.

## Experimental Design

---

To prepare for logistic regression and Support Vector Machine, I scaled the numeric features so that the mean would be approximately 0 and the standard deviation approximately 1. All features except for motion data were scaled. Random Forest does not require the scaling of numeric features.

For all models, I split the data into training and testing sets, with 80% of the data points devoted to training and 20% of the datapoints devoted to testing. To deal with the unbalanced dataset, I undersampled from the majority class in the training set. After rebalancing, the distribution of the target variable (Room\_Occupancy\_Count) was as follows:

- Class 0: 31.6%
- Class 1: 29.9%
- Class 2: 19.2%
- Class 3: 19.2%

To test the stability of the model, I applied 10-fold cross-validation on the training set. I then trained the model on the full training set and evaluated the model on the test set.

## Multinomial Logistic Regression

---

### All Features

For Multinomial Logistic Regression, the average cross-validation accuracy on the training set was 97.9%.

To implement Multinomial Logistic Regression, I chose a logistic regression classifier (newton-cg) that supports multiclass classification.

This table summarizes the evaluation metrics on the test set using all features:

Class	Accuracy	Precision	Recall	F1-score	Support
0		100.0%	99.2%	99.6%	1619
1		100.0%	100.0%	100.0%	103
2		97.0%	97.0%	97.0%	164
3		88.2%	96.4%	92.2%	140
Average	98.9%	96.3%	98.1%	97.2%	

**Accuracy:** The average accuracy was 98.9%.

**Precision:**

- Out of the total predictions for class 0, 100.0% belong to class 0 in the actual dataset.
- Out of the total predictions for class 1, 100.0% belong to class 1 in the actual dataset.
- Out of the total predictions for class 2, 97.0% belong to class 2 in the actual dataset.
- Out of the total predictions for class 3, 88.2% belong to class 3 in the actual dataset.

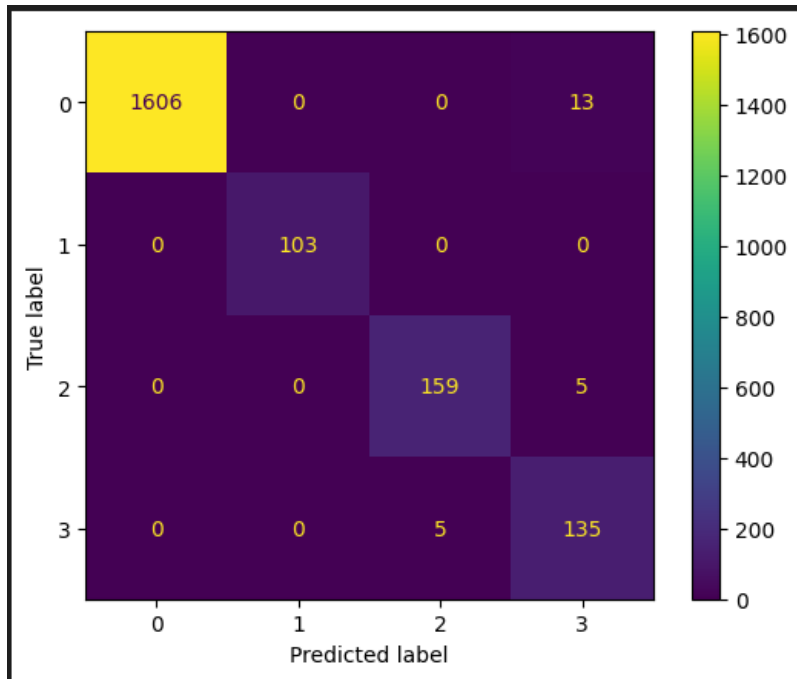
**Recall:**

- Out of the total entries in the dataset for class 0, 99.2% were classified in class 0 by the model.
- Out of the total entries in the dataset for class 1, 100.0% were classified in class 1 by the model.
- Out of the total entries in the dataset for class 2, 97.0% were classified in class 2 by the model.
- Out of the total entries in the dataset for class 3, 96.4% were classified in class 3 by the model.

**F1-score:** The average F1-score was 97.2%. Since this value is close to 1, the model did an excellent job of predicting the number of occupants in a room. The model did a better job of predicting 0-2 occupants then it did at predicting 3 occupants.

**Support:** In the actual test dataset, there are 1619 entries in class 0, 103 entries in class 1, 164 entries in class 2, and 140 entries in class 3.

From the confusion matrix, I observed that most of the wrong predictions were in predicting class 3:



## PCA Features

I reran the model with PCA using seven components.

This table summarizes the evaluation metrics on the test set using PCA:

Class	Accuracy	Precision	Recall	F1-score	Support
0		99.2%	99.4%	99.3%	1619
1		89.7%	93.2%	91.4%	103
2		71.5%	68.9%	70.2%	164
3		64.0%	63.6%	63.8%	140
Average	94.1%	81.1%	81.3%	81.2%	

**Accuracy:** The average accuracy was 94.1% with PCA compared to 98.9% without PCA.

**F1-score:** The average F1-score was 81.2.% with PCA compared to 97.2% without PCA. A significant drop in performance can be observed in both precision and recall for class 2 and class 3.

## Random Forest

### All Features

For Random Forest, the average cross-validation accuracy on the training set was 99.0%.

This table summarizes the evaluation metrics on the test set using all features:

Class	Accuracy	Precision	Recall	F1-score	Support
0		100.0%	99.8%	99.9%	1619
1		99.0%	100.0%	99.5%	103
2		99.4%	99.4%	99.4%	164
3		97.9%	99.3%	98.6%	140
Average	99.8%	99.1%	99.6%	99.3%	

**Accuracy:** The average accuracy was 99.8%.

**Precision:**

- Out of the total predictions for class 0, 100.0% belong to class 0 in the actual dataset.

- Out of the total predictions for class 1, 99.0% belong to class 1 in the actual dataset.
- Out of the total predictions for class 2, 99.4% belong to class 2 in the actual dataset.
- Out of the total predictions for class 3, 97.9% belong to class 3 in the actual dataset.

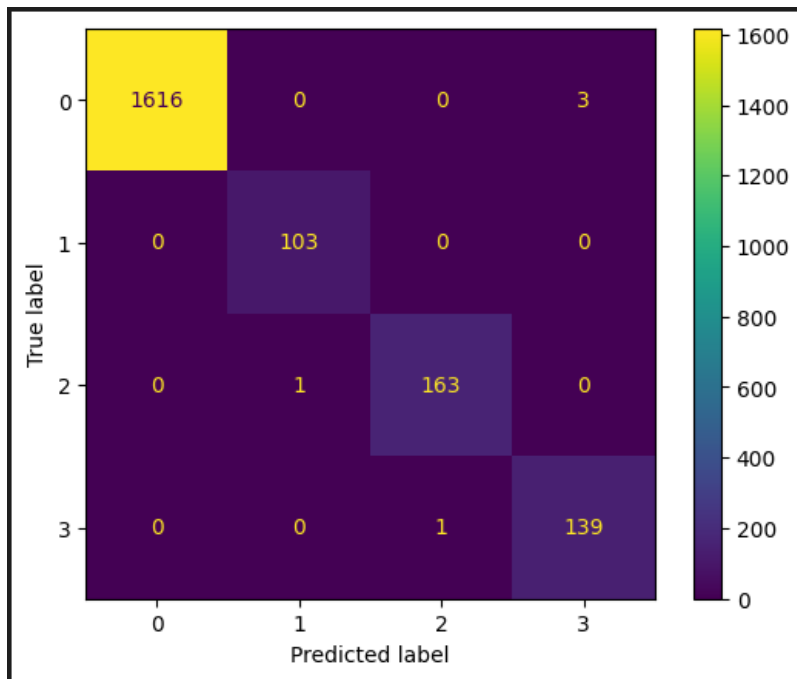
### Recall:

- Out of the total entries in the dataset for class 0, 99.8% were classified in class 0 by the model.
- Out of the total entries in the dataset for class 1, 100.0% were classified in class 1 by the model.
- Out of the total entries in the dataset for class 2, 99.4% were classified in class 2 by the model.
- Out of the total entries in the dataset for class 3, 99.3% were classified in class 3 by the model.

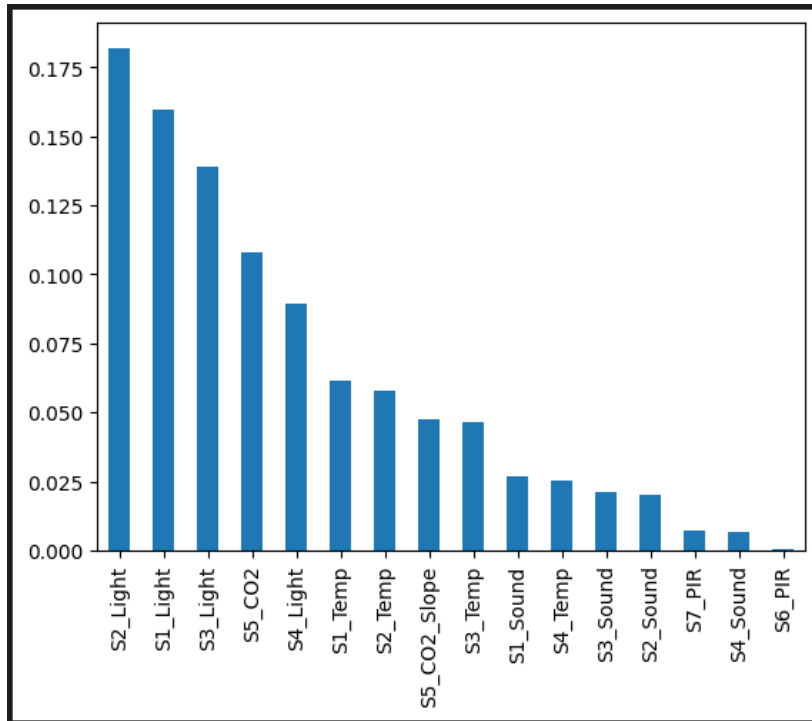
**F1-score:** The average F1-score was 99.3%. Since this value is close to 1, the model did an excellent job of predicting the number of occupants in a room.

**Support:** In the actual test dataset, there are 1619 entries in class 0, 103 entries in class 1, 164 entries in class 2, and 140 entries in class 3.

From the confusion matrix, I observed that there were only a few wrong predictions:



For Random Forest, I plotted the importance of each feature. As shown in the following graph, the most important features were light. The least important features were sound and motion.



## PCA Features

I reran the model with PCA using seven components.

This table summarizes the evaluation metrics on the test set using PCA:

Class	Accuracy	Precision	Recall	F1-score	Support
0		99.8%	99.8%	99.8%	1619
1		97.1%	97.1%	97.1%	103
2		97.5%	94.5%	96.0%	164
3		92.5%	96.4%	94.4%	140
Average	99.0%	96.7%	96.9%	96.8%	

**Accuracy:** The average accuracy was 99.0% with PCA compared to 99.8% without PCA.

**F1-score:** The average F1-score was 96.8.% with PCA compared to 99.3% without PCA. The results suggest that good performance could be achieved with fewer than 7 components.

# Support Vector Machine

---

## All Features

For SVM, the average cross-validation accuracy on the training set was 97.8%.

To implement SVM, I chose a non-linear kernel (rbf).

This table summarizes the evaluation metrics on the test set using all features:

Class	Accuracy	Precision	Recall	F1-score	Support
0		100.0%	99.4%	99.7%	1619
1		100.0%	100.0%	100.0%	103
2		97.0%	97.0%	97.0%	164
3		90.6%	96.4%	93.4%	140
Average	99.1%	96.9%	98.2%	97.5%	

**Accuracy:** The average accuracy was 99.1%.

### Precision:

- Out of the total predictions for class 0, 100.0% belong to class 0 in the actual dataset.
- Out of the total predictions for class 1, 100.0% belong to class 1 in the actual dataset.
- Out of the total predictions for class 2, 97.0% belong to class 2 in the actual dataset.
- Out of the total predictions for class 3, 90.6% belong to class 3 in the actual dataset.

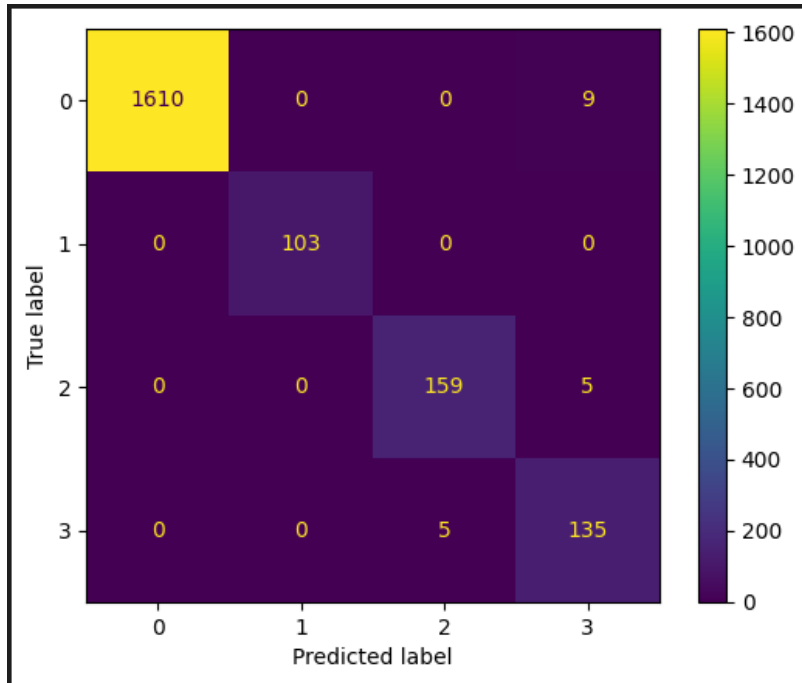
### Recall:

- Out of the total entries in the dataset for class 0, 99.4% were classified in class 0 by the model.
- Out of the total entries in the dataset for class 1, 100.0% were classified in class 1 by the model.
- Out of the total entries in the dataset for class 2, 97.0% were classified in class 2 by the model.
- Out of the total entries in the dataset for class 3, 96.4% were classified in class 3 by the model.

**F1-score:** The average F1-score was 97.5%. Since this value is close to 1, the model did an excellent job of predicting the number of occupants in a room. The model did a better job of predicting 0-2 occupants than it did at predicting 3 occupants.

**Support:** In the actual test dataset, there are 1619 entries in class 0, 103 entries in class 1, 164 entries in class 2, and 140 entries in class 3.

From the confusion matrix, I observed that most of the wrong predictions were in predicting class 3:



## PCA Features

I reran the model with PCA using seven components.

This table summarizes the evaluation metrics on the test set using PCA:

Class	Accuracy	Precision	Recall	F1-score	Support
0		99.9%	99.8%	99.9%	1619
1		100%	100%	100%	103
2		95.8%	96.3%	96.0%	164
3		95.0%	95.7%	95.4%	140
Average	99.3%	97.7%	98.0%	97.8%	

**Accuracy:** The average accuracy was 99.3% with PCA, which exceeds the model without PCA (99.1%).

**F1-score:** The average F1-score was 97.8.% with PCA, which exceeds the model without PCA (97.5%). The results suggest that good performance could be achieved with fewer than 7 components.



## Findings

### Summary of Model Results

This table summarizes the performance metrics of the models:

	All Features			PCA	
Model	Accuracy Training Set	Accuracy Test Set	F1-Score Test Set	Accuracy Test Set	F1-Score Test Set
Multinomial Logistic Regression	97.9%	98.9%	97.2%	94.1%	81.2%
Random Forest	99.0%	99.8%	99.3%	99.0%	96.8%
Support Vector Machine	97.8%	99.1%	97.5%	99.3%	97.8%

When using all features, Random Forest performed the best in terms of both accuracy and F1-score. These results are consistent with Mao et al. (2023), which also found that Random Forest performed the best in all three of their metrics.

When using PCA, Support Vector Machine performed the best in terms of both accuracy and F1-score. The results suggest that good results could be achieved with fewer than 7 components. Indeed, Singh et al. (2018) concluded that an accuracy of 92% and F1-score of 72% was achievable with only four components.

Good results were also achieved when using PCA with Random Forest. However, using PCA with Multinomial Logistic Regression is not recommended based on the reduction in F1-score.

### Summary of Sensor Data

Using linear regression, light data had the highest R-squared value at 79.2% followed by CO<sub>2</sub> data at 74.6%. Less promising features were temperature data (54.8%), sound data (44.4%), and motion data (40.1%). Using all sensor data provided the best model fit, with an R-squared value of 89.4%.

For the first component of PCA, light data from sensor 1 had the largest contribution, followed by light data from sensor 3, and temperature data from the first three sensors.

In the Random Forest model, the most important features were light data from sensor 2, sensor 1, and sensor 3. Similarly, Mao et. al (2023) found that light values from sensor 1 and sensor 2 had the largest impact in predicting room occupancy. The fourth most important feature was CO<sub>2</sub> data. The least important features were motion and sound data.

Based on these results, it could be concluded that light data performed the best overall in predicting room occupancy. However, in Singh, et al. (2018), the authors rejected using light data since the results relied on occupants turning on desk lights when they arrived and turning them off again when they left. Indeed, lights may be best controlled by the environmental system itself based on whether the room is occupied.

Although CO<sub>2</sub> data showed promising results, the best approach is to use a combination of different types of sensor data, including CO<sub>2</sub>.

With the advent of Internet of Things (IoT) technologies, alternative types of sensor data such as Bluetooth signals, Wi-Fi, camera images, GPS data, UWB radar, and electric meters could also be considered. While these technologies have their advantages, limitations such as high cost, limited range, false results, and privacy issues may prevent them from becoming widely adopted (Khan et al., 2024).

## **Limitations and Future Work**

---

Linear regression and logistic regression models assume a linear relationship between the independent variables and the dependent variable. In addition, linear regression and logistic regression models assume that the independent variables are not correlated with each other. However, in the Room Occupancy Estimation dataset, the sensor data are moderately to highly correlated with each other. This is known as multicollinearity. As a result, it can be difficult to isolate the impact of an individual feature on the dependent variable. For example, if the temperature value of a sensor increases, the temperature value of nearby sensor will also increase.

Because of multicollinearity, the reported p-values and coefficients in some of the linear regression models may not be reliable. To fix multicollinearity, PCA can be used to create uncorrelated features.

The Logistic Regression Model was rerun using uncorrelated features created by PCA. However, a significant drop in performance was noted in both precision and recall for class 2 and class 3.

Linear regression models are also sensitive to outliers. Data points that are far away from the cluster of points can have a major impact on the calculated error. In the Room Occupancy Estimation dataset, outliers were detected in all the numerical sensor data (temperature, light, sound, and CO<sub>2</sub>).

This paper concludes that Random Forest performed the best when using all features and SVM performed the best when using PCA. However, the difference in model results may not be statistically significant. Hypothesis testing of the model results, for example using the Friedman test, could be considered for future work.

In addition, future work could involve feature selection (for example, removal of light features), using PCA with fewer components, and the evaluation of more classification algorithms.

## References

---

1. Masoso, O.T., & Grobler, L.J. (2010). The dark side of occupants' behaviour on building energy use. *Energy and Buildings*, 42(2), 173-177.  
<https://doi.org/10.1016/j.enbuild.2009.08.009>
2. Singh, A.P., Jain, V., Chaudhari, S., Kraemer, F.A., Werner, S., & Garg, V. (2018). Machine learning-based occupancy estimation using multivariate sensor nodes. *IEEE Globecom Workshops (GC Wkshps)*, 1-6.  
<https://doi.org/10.1109/GLOCOMW.2018.8644432>
3. Mao, S., Yuan, Y., Li, Y., Wang, Z., Yao, Y., & Kang, Y. (2023). Room occupancy prediction: Exploring the power of machine learning and temporal insights. *American Journal of Applied Mathematics and Statistics*.  
<https://doi.org/10.48550/arXiv.2312.14426>
4. Khan, I., Zedadra, O., Guerrieri, A., & Spezzano, G. (2024). Occupancy prediction in IoT-enabled smart buildings: technologies, methods, and future directions. *Sensors*, 24(11). [doi.org/10.3390/s24113276](https://doi.org/10.3390/s24113276)