Methods of Blood Pressure Predictions for Noninvasive Sensors

Morayo Adeyemi, Parker Ruth

Abstract

Accurate blood pressure monitoring is crucial for diagnosing and managing hypertension, a leading risk factor for cardiovascular disease. Traditional office-based measurements can be influenced by factors like the "white-coat effect," leading to misdiagnosis. Continuous 24-hour monitoring offers a more comprehensive assessment but is limited by the availability of practical noninvasive sensors. This study explores the use of predictive models for blood pressure estimation from tonometer data, comparing real-world measurements with simulated data from the Pulse Wave Database (PWDB). The tonometer provides continuous pulse pressure readings, while the simulated data replicates blood flow variations throughout the body. We implemented and evaluated Linear regression, K-Nearest Neighbors (KNN), and Convolutional Neural Network (CNN) models on segmented pulse data. Additionally, we applied K-means clustering to group similar pulse shapes, enhancing the performance of KNN models in particular. Our findings indicate that while CNNs excel in broad datasets, KNN models significantly benefit from clustering, achieving more precise blood pressure predictions. Notably, the comparison between tonometer and simulated data revealed a substantial gap in accuracy, with simulated data offering a simplified representation that performed better in predictive modeling. However, translating these methods to more variable real-world data remains challenging. The results underscore the potential of combining clustering techniques with KNN for improving noninvasive blood pressure monitoring and highlight the importance of refining simulation data to better match real-world variability.

Introduction

Cardiovascular disease remains a leading cause of death globally, with hypertension being one of the most common and manageable conditions.[1] Despite its treatable nature, over 580 million people worldwide remain undiagnosed.[2] Annual office visits often contribute to misdiagnosis of high blood pressure due to factors like the "white-coat effect,"[3] where patients exhibit elevated blood pressure due to anxiety in a clinical setting. This can lead to "false positives," where doctors diagnose hypertension even when the patient's blood pressure may be normal outside the clinical environment. Conversely, "false negatives" can occur when a patient's blood pressure spikes in other situations that are not captured during a one-time office visit, leading to an undetected condition.

Continuous 24-hour monitoring is recognized as a superior method for accurately diagnosing hypertension, as it provides a more comprehensive view of a patient's blood pressure throughout their daily activities.[4] However, the lack of practical, existing sensors for continuous blood pressure sensing presents a significant challenge. To address this gap, devices like the Microsoft tonometer have been developed, offering continuous measurement of pulse pressure.[5]

In their study, Microsoft conducted tests where participants wore the tonometer device alongside a traditional blood pressure cuff while engaging in various activities. These activities included both auscultatory and oscillometric testing. Participants completed 7 different auscultatory activities, with each activity being repeated multiple times during the study. Additionally, the researchers collected data on participants' medical history, including known arterial diseases, age, and weight.

Since conducting such large ambulatory blood pressure data collections is challenging, an alternative is to use computational simulations. In this work, we examine a simulated dataset called the "Pulse Wave Database (PWDB)".[6] This dataset simulates blood flow throughout the body by varying the size of arteries. The simulated data includes blood flow velocity, luminal area, volume flow rate, and pulse waves.

The goal of this project was to compare the simulated data with real data from the tonometer device, which measures blood pressure. The objective was to assess the usefulness of the simulated data and to identify insights that could be gained from comparing it to the tonometer readings.

Methods

A. Data

Tonometer recordings ranged from 10 to 50 seconds, with the pulse data from each activity needing to be downsampled for comparison with the simulation data. A Slope Sum function[7] was used to accurately segment the pulses within the time series data. To isolate the baseline drift in the tonometer signal, a low-pass Butterworth filter with a frequency of 0.5 was applied, as illustrated in Figure 1A. This filtered curve was then subtracted from the tonometer signals, shown in Figure 1B . Scipy's find_peaks function was subsequently used with a minimum distance of 80 units between each peak, ensuring the elimination of local minima and resulting in a more precise pulse representation.

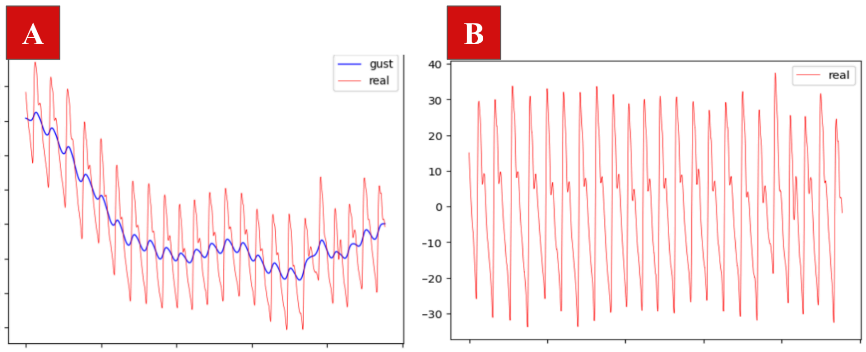

Figure 1: The left plot shows the effect of applying a low-pass Butterworth filter, where the red line represents the unfiltered signal and the blue line represents the low-pass filtered signal. The right plot displays the pulses with baseline removed, aligning them for more accurate pulse segmentation.

B. Data Cleaning

By applying a low-pass filter to tonometer readings that are excessively noisy or improperly recorded by the device, it is possible to eliminate unusable data, example Figure 2A, 2B. However, excessively noisy signals may still exhibit significant deviations even after filtering, resulting in Figure 2C, 2D. These deviations can distort the pulse waveform, making it difficult to accurately identify local minima.

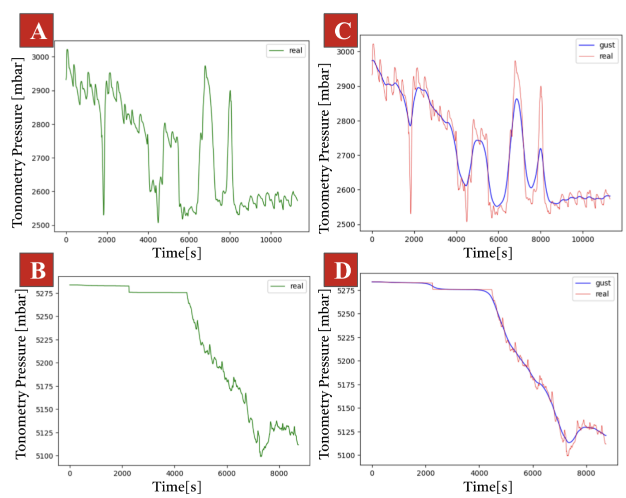

Figure 2: The A and B show a tonometer reading, the C and D show the low pass butter filter,

C. Predictive models

With the pulses correctly segmented, the average pulse was used to represent the tonometer recording and was downsampled to 50 points per pulse for predictive modeling. Three kinds of predictive models were trained to predict blood pressure from segmented tonometry pulses: Linear regression, k-nearest neighbors (KNN), and convolutional neural networks (CNN) all models trained and tested using cross-validation.

The decision to implement a CNN was inspired by prior studies that successfully applied neural networks to similar tasks, such as predicting physiological signals from photoplethysmography (PPG) sensors.[8] Specifically, a simple CNN was implemented with three convolutional layers consisting of 32, 64, and 128 neurons, along with dropout layers to prevent overfitting.

D. Grouping data

To analyze the variation in the one-dimensional pulse data, an iterative process was developed using the tonometer data. K-means clustering, based on Euclidean distances, was applied to group the pulses by similarity. For each cluster, the range of diastolic and systolic values was calculated, and clustering continued until the mean deviation within all clusters was reduced to below 5 mmHg. The average size of each cluster was also considered, as small clusters would be less reliable for predictive testing.

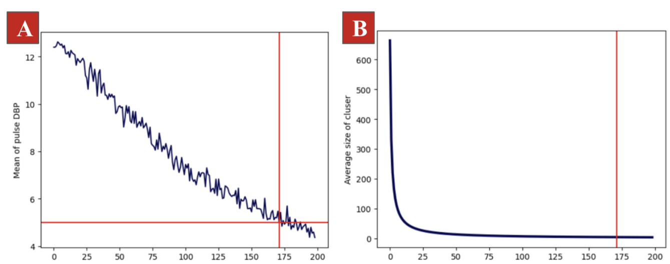

Figure 3: 3A shows iterations of testing for the range of diastolic values within the cluster's pulses. 3B displays the average group sizes within the clusters.

This approach identified unique pulse patterns across different datasets and enabled comparisons to determine whether certain datasets performed better due to higher similarity in pulse shapes. The clusters with a mean deviation within ± 5 mmHg were used as input for a secondary algorithm that iteratively applied linear regression, KNN, and CNN to each cluster. The mean absolute error for all clusters was calculated and represented the error for the entire cluster group. The process continued until the maximum number of clusters with a standard deviation of 5 mmHg was reached.

Through this process, the impact of pulse similarity, as measured by Euclidean distance, on the performance of predictive models was evaluated. Finally, four testing approaches were compared for blood pressure predictions: cross-validation within individual patient data, across task-specific data, across a combined dataset of all pulses, and using clusters based on one-dimensional pulse shapes.

Results

Predictive modeling for blood pressure aims to achieve an accuracy within 5 mmHg and a standard deviation below 8 mmHg, as per the AAMI standard.[9]

A. Testing within individual patient data

Initial tests using person-specific data focused on predicting blood pressure within the same individual. Cross-validation with linear regression revealed that 53 out of 671 participants achieved an average error under 5 mmHg, Figure 4A. In contrast, K-Nearest Neighbors (KNN) improved this result, with 152 participants meeting the same criteria of under 5 mmHg error, Figure 4B.

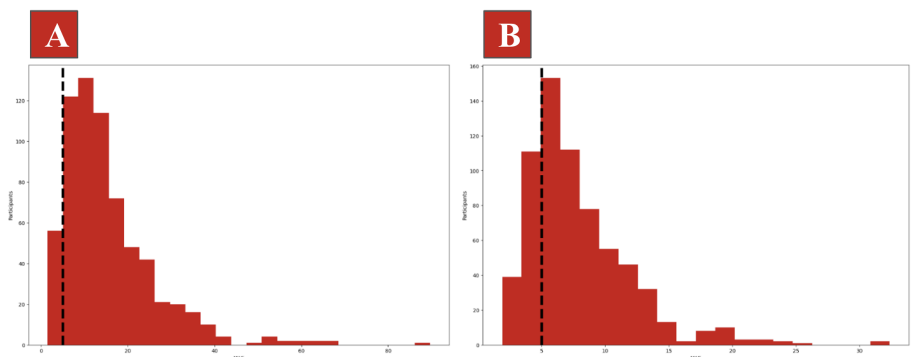

Figure 4: 4A is a histogram of participants tested with linear regression who met the industry standard. 4B shows participants tested with KNN who met the industry standard.

B. Cross-validation across a combined dataset of all pulses.

With all the average pulse samples collected, testing was conducted on all the pulses simultaneously. This approach was used to determine whether predicting blood pressure across different tasks could yield better results, Table 1.

Table 1

Auscultatory Testing	MAE	SD of Error
Linear Regression (systolic)	16.29	2.55
Linear Regression (diastolic)	9.91	1.53

KNN (systolic)	15.89	5.19
KNN (diastolic)	10.03	3.36
CNN (systolic)	16.09	3.22
CNN (diastolic)	9.96	1.69

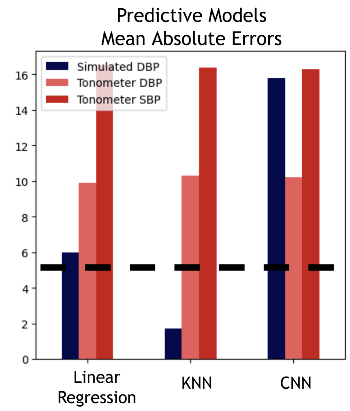

Figure 5: Linear regression, KNN, and CNN bar graph tested on all data for tonometer and all data for Simulated.

C. Cross-validation across task-specific data.

Testing within a specific task might reveal similarities within that task. The activities were cross-validated within their own categories. See Appendix A for detailed tables containing Linear regression, KNN, and CNN mean absolute error for all Auscultatory testing.

D. Simulated data all pulses

Simulated data was tested using Linear, KNN, and CNN.

Table 2

Simulated data	MAE	SD of Error
Linear	1.10	0.16
KNN	1.48	0.24
CNN	21.38	4.93

With KNN, K-means clustering was examined as a method to improve the performance of KNN, given that KNN operates by comparing one-dimensional pulse shapes to similar shapes and using the corresponding blood pressure for predictions. K-means clustering, an algorithm that groups data points based on their distance from each other, was applied to determine the number of clusters of one-dimensional pulses present in the simulated dataset. This analysis revealed a surprising result: only about 15 unique pulse clusters were identified across the 4,000 simulations.

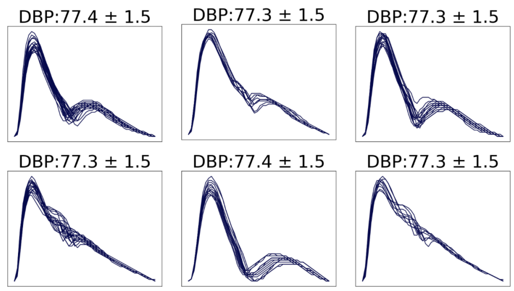

Figure 6: Each subplot represents a unique cluster of one-dimensional pulse shapes, with the corresponding average diastolic values and its standard deviation indicated.

The CNN demonstrated improved accuracy, particularly in temporal data analysis. However, KNN performed exceptionally well on the simulated dataset, achieving an error of 1.7 mmHg, which is significantly below the industry standard.

E. Cross-validation using clusters based on one-dimensional pulse shapes.

An iterative approach was used to determine whether clustering groups by their one-dimensional shape affects the accuracy of blood pressure predictions. The clustering algorithm was executed repeatedly until it identified clusters corresponding to the lowest systolic and diastolic values. The analysis revealed approximately 170 unique pulse shape groupings, with further testing across different activities yielding between 110 and over 250 unique clusters. The objective was to find the optimal number of clusters that would yield the lowest average MAE for the model, thereby indicating the most effective number of clusters for the algorithm, Figure 5B.

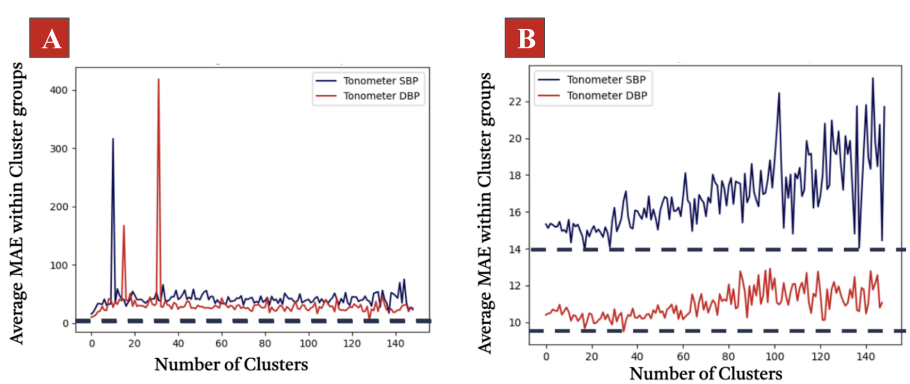

Figure 7: Comparison of the average Mean Absolute Error (MAE) within cluster groups across different numbers of clusters. Figure 5A illustrates how the MAE varies with the number of

clusters when applying linear regression to predict systolic (SBP) and diastolic (DBP) blood pressure. Figure 5B highlights the clustering performance for systolic and diastolic blood pressure predictions, showing the effectiveness of using KNN to minimize MAE with the optimal number of clusters.

Discussion

The analysis shows that Convolutional Neural Networks (CNNs) performed better when tested on the entire auscultatory dataset compared to individual tasks. When using the full dataset, the CNN model had a Mean Absolute Error (MAE) of 16.09 mmHg for systolic and 9.96 mmHg for diastolic predictions, with relatively low variation. However, when CNNs were applied to specific tasks like "Temporal_challenge_start," the errors increased, with MAEs of 19.6 mmHg for systolic and 12.28 mmHg for diastolic predictions. Other tasks, like "Seated_calibration" and "Exercise_challenge_start," showed even higher errors. This suggests that CNNs are more accurate when they can learn from the full range of data in the auscultatory dataset, rather than being limited to individual tasks, where their performance can vary more widely.

The results of this study offer valuable insights into the challenges and potential improvements in using noninvasive sensors for blood pressure prediction. The comparison between simulated and real data predictions underscored a substantial gap in accuracy. For instance, the Mean Absolute Error (MAE) for the KNN model was 1.48 mmHg on the simulated dataset but increased to 10.03 mmHg when applied to real tonometer data.

Reducing tonometer waveform data to 50 points per pulse likely missed critical features of the pulse waveforms. This reduction oversimplified the waveform by averaging out important details, such as the length of pauses between pulses and the precise shape of systolic and diastolic peaks. As a result, key variations in the waveform that are essential for accurate blood pressure predictions were lost. A more refined sampling method, one that captures pulses based on key distinctive features, could more accurately reflect the nuances of each tonometer waveform and improve predictive accuracy.

Further analysis of the diastolic values for the simulated pulses showed that they were all very similar, with an average value of 77.3 mmHg and a standard deviation of 1.5 mmHg within each group. The simulated data was found to contain only seven unique diastolic values.

The presence of a significant number of unique pulse shapes in the tonometer data, compared to the simulated data's 15 unique shapes, likely explains why KNN performed better on the simulated data. The next step was to determine whether the techniques used on the simulated data, which featured fewer pulse shapes and less variation in diastolic values, could be effectively applied to the tonometer data.

To imitate KNN methods on real data the clustering analysis revealed that while there are groups of similar tonometer pulses, achieving low blood pressure variation within a cluster required an average of only 5-10 pulses per group.

To explore these differences further, linear regression, KNN, and CNN models were tested on each cluster group. The analysis aimed to determine whether testing pulses with similar one-dimensional shapes was more effective than testing random shapes. Interestingly, applying linear regression to each cluster resulted in a significant increase in mean error, indicating that clustering was not beneficial for this method. Conversely, when KNN was applied within these clusters, the results improved considerably. Increasing the number of clusters and focusing on KNN within those clusters led to a substantial reduction in prediction error for both systolic and diastolic values, demonstrating a clear improvement in accuracy compared to testing CNNs on the entire activity.

Using the task from the study with the lowest predicted model error, Temporal_challenge_start, the analysis demonstrated that clustering significantly improved the performance of the KNN model for blood pressure prediction, particularly when compared to the original, non-clustered approach. For systolic blood pressure predictions, the Mean Absolute Error (MAE) for KNN decreased from 14.22 mmHg before clustering to 13.21 mmHg after clustering. In contrast, the CNN model, which had a higher MAE of 19.6 mmHg before clustering, was not tested within clusters, indicating that KNN was the more effective approach in this context. Similarly, for diastolic blood pressure predictions, the KNN model's MAE improved from 9.85 mmHg before clustering to 9.41 mmHg after clustering, while the CNN model had a higher MAE of 12.28 mmHg before clustering. These results clearly indicate that clustering the data into groups with similar pulse shapes led to more accurate predictions, particularly for the KNN model, which outperformed the CNN model both before and after the clustering process.

Conclusion

Overall, with their lack of variability in the simulated data, was found to be overly simplified compared to real-world data, this resulted in models that do not perform as well with complex patterns present in real human data. The application of clustering to group similar pulse shapes proved to be a promising approach, particularly for the KNN model, which showed clear improvements in accuracy following this method. These results indicate that while CNNs benefit from broad datasets, KNN models can be enhanced through strategic data clustering, offering a more precise tool for blood pressure prediction from tonometer data.

References:

[1] World Health Organization. "Cardiovascular Diseases." World Health Organization,
[2] World Health Organization. More than 700 million people with untreated hypertension, 2021
[3] Schutte, A.E., Kollias, A. & Stergiou, G.S. Blood pressure and its variability: classic and novel measurement techniques. *Nat Rev Cardiol* 19, 643–654 (2022).
[4] Giuseppe Mancia, Paolo Verdecchia. “Clinical Value of Ambulatory Blood Pressure: Evidence and Limits.” Ahajournals, 2015
[5] R.J. Mieloszyk et al. "A Comparison of Wearable Tonometry, Photoplethysmography, and Electrocardiography for Cuffless Measurement of Blood Pressure in an Ambulatory Setting," IEEE Journal of Biomedical and Health Informatics, vol. 26, no. 7, pp. 2864–2875, 2022
[6] Charlton P.H., Mariscal Harana, J., Vennin, S., Li, Y., Chowienczyk, P. & Alastruey, J., “Modeling arterial pulse waves in healthy aging: a database for in silico evaluation of haemodynamics and pulse wave indices,” AJP Heart. Circ. Physiol., [in press], 2019.
[7] W. Zong, T. Heldt, G. B. Moody, and R. G. Mark, “An open-source algorithm to detect onset of arterial blood pressure pulses,” in Computers in Cardiology, 2003, Sep. 2003, pp. 259–262.
[8] S. Yang, J. Sohn, S. Lee, J. Lee and H. C. Kim, "Estimation and Validation of Arterial Blood Pressure Using Photoplethysmogram Morphology Features in Conjunction With Pulse Arrival Time in Large Open Databases," in IEEE Journal of Biomedical and Health Informatics, vol. 25, no. 4, pp. 1018-1030, April 2021
[9]Non-invasive sphygmomanometers - Part 2: Clinical investigation of automated measurement type. American National Standards Institute; 2013. ANSI/AAMI/ISO 81060-2.

Appendix A

Table A1: Auscultatory Linear Regression Systolic testing

Auscultatory Testing	MAE (Linear Regression)	SD of Error (Linear Regression)
Calibration_start	14.14	3.64
Static_challenge_start	14.42	3.50
Seated_calibration	15.77	3.87
Static_seated_challenge	15.98	3.54
Exercise_challenge_start	24.59	6.74
Temporal_challenge_start	14.17	3.12
Temporal_seated_challenge	15.60	4.45

Table A2 - Auscultatory Linear Regression Diastolic testing

Auscultatory Testing	MAE (Linear Regression)	SD of Error (Linear Regression)
Calibration_start	9.78	1.02
Static_challenge_start	9.98	1.25
Seated_calibration	9.56	0.76
Static_seated_challenge	10.22	1.46
Exercise_challenge_start	16.39	5.75
Temporal_challenge_start	9.94	1.32
Temporal_seated_challenge	10.02	1.71

Table A3 - Auscultatory KNN Systolic testing

Auscultatory Testing	MAE (KNN)	SD of Error (KNN)
Calibration_start	14.47	1.76
Static_challenge_start	14.81	1.38
Seated_calibration	15.73	1.23
Static_seated_challenge	15.36	2.14
Exercise_challenge_start	22.20	4.75
Temporal_challenge_start	14.22	1.61
Temporal_seated_challenge	15.06	1.50

Table A4 - Auscultatory KNN Diastolic testing

Auscultatory Testing	MAE (KNN)	SD of Error (KNN)
Calibration_start	9.73	1.22
Static_challenge_start	9.82	1.45
Seated_calibration	9.47	1.25
Static_seated_challenge	10.04	1.6
Exercise_challenge_start	13.96	2.05
Temporal_challenge_start	9.85	1.11
Temporal_seated_challenge	9.81	1.38

Table A5 - Auscultatory CNN Systolic testing

Auscultatory Testing	MAE (CNN)	SD of Error (CNN)
Calibration_start	16.82	3.11
Static_challenge_start	19.37	3.84
Seated_calibration	35.26	5.08
Static_seated_challenge	38.14	5.51
Exercise_challenge_start	33.98	11.29
Temporal_challenge_start	19.6	6.35
Temporal_seated_challenge	34.68	4.4

Table A6 - Auscultatory CNN Diastolic testing

Auscultatory Testing	MAE (CNN)	SD of Error (CNN)
Calibration_start	11.79	3.5
Static_challenge_start	12.65	2.89
Seated_calibration	23.88	6.92
Static_seated_challenge	22.81	2.16
Exercise_challenge_start	20.11	4.68
Temporal_challenge_start	12.28	1.12
Temporal_seated_challenge	21.89	4.53