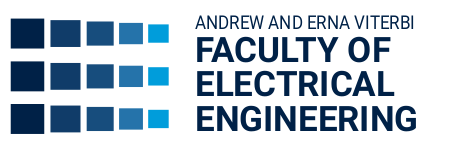
****

**Blood Pressure estimation based on PPG using neural network**

Project Submitted By: Dorin Alon & Shirili Shelef

Project Supervisors: Ron Teichner & Dr. Danny Eitan

**Contents:**

1. Abstract //shirili
2. Introduction:
   1. Motivation //shirili
   2. Project goal //dorin
   3. Medical background //dorin - *(write something about the database?)*
3. Theoretical background- neural network
   1. Artificial neural network
4. Project flow
   1. Synthetic signals //(in short) //dorin
   2. Choosing the input signals to the net //shirili – choosing PPG because it’s similar to BP
   3. Block diagram//shirili
5. Results:
   1. Creation and optimization of the net //dorin

Took an existing lstm and changed it to our needs -from where?

Talk about the nn structure-

Adam optimizer

Activation function- tanh

Loss function- MSE

What happens in every iteration

Add mean loss graph

Decide the nn characters with graphs to show

Without batch and then with batch

* 1. Final Architecture//shirili -only explenation on LSTM
  2. Training on a single patient and predicting BP of the same patient//shirili
  3. Training on a single patient and predicting BP of different patients//shirili

1. Conclusion:// dorin
   1. Discussion *//algorithm and problems-*

* Sum up good results for single patient for minimal time of couple of hours
* We didn’t find a pattern for the times in which the prediction succeeded and for which it didn’t.
* We can see that we need diversity in the train set, so we recommend the following algorithm:
* ALGORITHM
* It has flaws: *normalization data , clips position, various train*
  1. Future works- investigate the connection between PPG wave structure and BP changes- because we see it is connected, and our nn can't handle it.
* Data augmentation- create a various input to our nn by manipulation on the BP, PPG signals
* Train on various patients. Use nn that already trained on some patients and it's weight is already initialized.

1. Appendix:
   1. results from synthetic signals //Dorin
   2. results from train and test on the same patient//shirili
   3. results from train on a single patient and test on another patient//shirili
   4. Code and user instructions // dorin- fix code
2. References //shirili

**1. Abstract**

Blood pressure (BP) is a significant vital that is monitored for every patient in intensive care. BP could indicate patients' cardiovascular status. Nowadays, BP is measured throughout an invasive manner - catheter directly into an artery.

This form of measurement is not convenient for the patient and could lead to an infection. Photopletysmogram (PPG) is a signal measured in a non-invasive manner – a clips on the patient's finger. In this project, we will suggest a way to estimate BP using neural network (NN) type LSTM with PPG as it's only input.

**2. Introduction**

**2.1 - motivation:**

Critically ill patients are monitored continuously with vital physiologic signals. The medical team need to be in control over the patient signals, specifically blood pressure (BP). The way to measure this vital is a catheter into an artery, this way the signal can be presented continuously on a monitor. Another way to measure BP is with a cuff on a patient's limb, but the measurement is taken periodically, and the result isn't accurate enough and not continuous. Therefore, patient BP is monitored in an invasive way, which can lead to an infection and thrombosis (clot).

There is no direct formula which connect BP values to other vitals that are measured in non-invasive ways, such as electrocardiogram (ECG), respiratory impedance (RI), and PPG. Nevertheless, the connection between the signals exists, so artificial intelligence was required to learn a patient and figure out the connection. In this project we will present the neural network we used to estimate BP using PPG.

*הסבר על חשיבותו של לחץ הדם עבור מטופלים, למה אין דרך אחרת חוץ מהקטטר, ולמה בחרנו ברשת נוירונים שתלמד את המטופל.*

**2.2 – project goal:**

As mentioned in the motivation above, a patient's blood pressure is measured in a very invasive way. The goal of this project is to estimate a patient's blood pressure by means of deep learning methods, functioning on non-invasive continuously measured vitals, such as ECG, RI and PPG. By that, it is possible to avoid BP measuring at all, or to measure it for a short period of time. The estimation is required to be quite accurate, and predict BP elevation or descent, since it can indicate a patient's status, and alert the medical crew before an emergency.

**2.3 – medical background**

Our database is composed of 500 critically ill patients, from the children ICU department of Hospital A in Toronto.   
For every patient, the vitals PPG, ECG, RI and BP were sampled for a consistent period of 40 minutes. For every patient we got several 40 minutes sampling segments. We received raw, noisy and not normalized signals. Usually their Y axis values were meaningless, while their shape and wave pattern are the meaningful values.

BP, stands for blood pressure, is the pressure of circulating blood on the walls of blood vessels. It is usually expressed in terms of systolic pressure (maximum during heartbeat), over diastolic pressure (minimum in between two heartbeats). Blood pressure is measured in a very invasive way, by inserting a catheter to an artery. The blood pressure signals from our database are sampled at 125 Hz, and the transformation equation: we converted the signal to units of mmHg.

PPG, stands for photoplethysmogram, is an optically obtained [plethysmogram](https://en.wikipedia.org/wiki/Plethysmograph) that can be used to detect blood volume changes in the microvascular bed of tissue. A PPG is often obtained by using a [pulse oximeter](https://en.wikipedia.org/wiki/Pulse_oximeter) which illuminates the [skin](https://en.wikipedia.org/wiki/Skin) and measures changes in light absorption. The PPG signals from our database are sampled in 125 Hz, with no consideration to values in Y axis nor to measuring units.

ECG, stands for electrocardiogram, represents the electrical activity of the heart using electrodes placed on the skin. Changes in the normal ECG pattern occur in numerous cardiac abnormalities, including cardiac rhythm disturbances. The ECG signals in our database are sampled in 500 Hz, with no consideration to Y axis values nor to measuring units. The valuable information from this signal is the wave's shape and pattern.

RI, stands for respiratory impedance, represents the analysis of pressure, flow or volume of the patient's respiration. It is measured in the same way as ECG, using the electrodes on the patient's skin to measure the chest volume. This signal is sampled in 62.5 Hz, also with no consideration to Y axis values.

Until these days, no one has ever found a strong empirical equational connection between the four signals. Yet, just by looking at the signals' shapes, we are able to see some connection. In this project we are hoping that the neural network can find a connection based on the given examples.

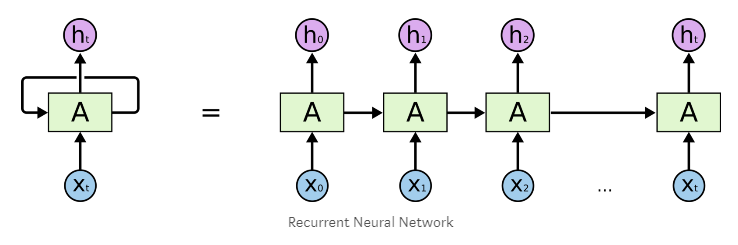
**3. theoretical background**

**3.1 neural networks**

Neural networks are a set of algorithms, trying to imitate the human brain, and therefore are built from individual parts approximating neurons. The neural network is designed to learn from various examples, and by that to be able to recognize patterns. It can be used for a classification problem, in which the network task will be to identify or recognize certain things from a given dataset, based on labeled examples. It can also be used for a clustering problem, in which the task will be detection of similarities. The main use of neural networks in this project is answering a regression problem. Deep learning with nn is able to establish correlations between features in a dataset, or between present and future events.

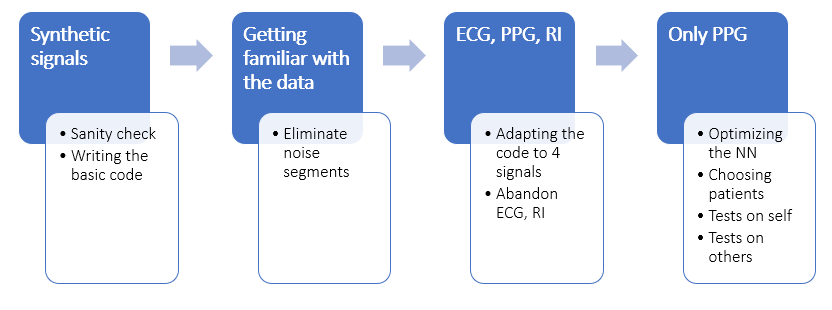
**3.2 LSTM**

Long Short-term memory (LSTM) is an artificial Recurrent neural network (RNN) architecture that can process an entire sequence of data, and not a single point. This quality of memory of the past is necessary for learning a patient's behavior. The downside of LSTM is the need for a powerful processing unit, so we used the GPU of the lab.



**3.1 - Architecture:**

**4. Project flow**



**4.1 – synthetic signals:**

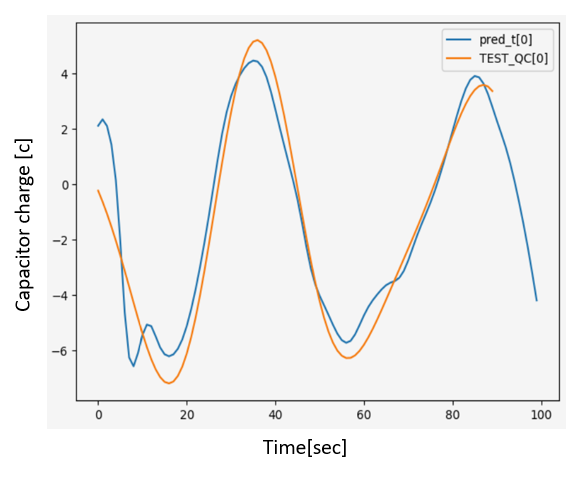
In order to ensure that our neural network is working, we performed a sanity check. We created four types of synthetic signals that simulate~~s~~ an electric circuit. We created an equation that has a mathematical solution, and by that we examined whether the neural network is able to learn a proven existing connection. The four signals are source voltage, resistor voltage, inductor current and capacitor charge. We especially chose the synthetic signals so that they will be somehow equivalent to our medical signals, and this experiment will simulate the medical situation.

The capacitor simulates a blood vessel, its capacity simulates the volume of blood that the vessel can contain.

The equation is:

We divided the signals to train and test parts, and used our LSTM network.

the results are shown below:



this network has the following characters: hidden layers = 30, num layers = 1, tau = 10, iterations number = 200, where tau is the time difference between the train segment and the test segment. With the tau variable, we examine the NN's ability to predict future outcomes. We can see that the network succeded in the current evaluation of and also in future evaluation (the descent in the end).

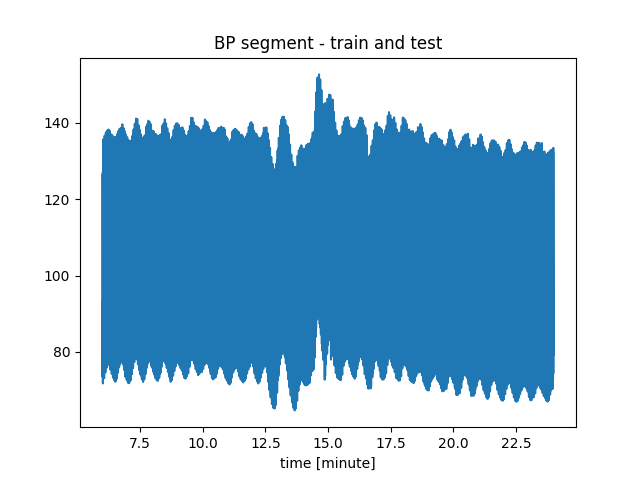
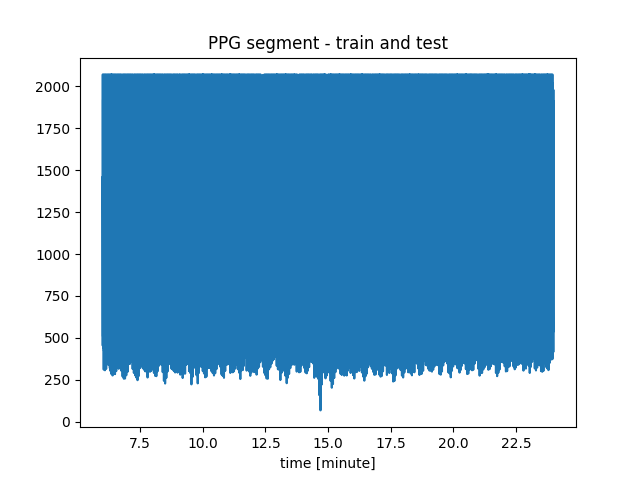
After the sanity check, we were confident with our LSTM network, and ready to test it on the medical signals.

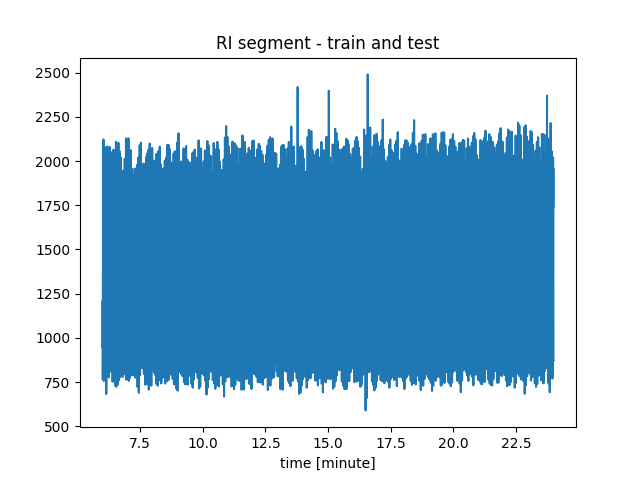
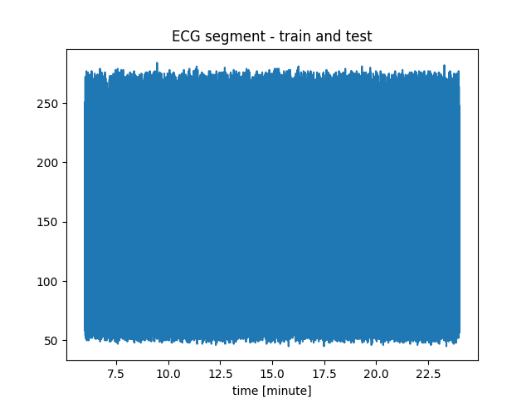
**4.2 - Choosing the input signals to the net:**

By looking at our data, the most similar vital to BP is PPG – they are sampled in the same ratio, they seem to have similar cyclic *(periodic מחזורי)* behavior, and medically it makes sense to research their correlation – every heart beat delivers blood saturated with oxygen, so a change in blood pressure is correlated with change in oxygen level in it (which is exactly what PPG indicates).

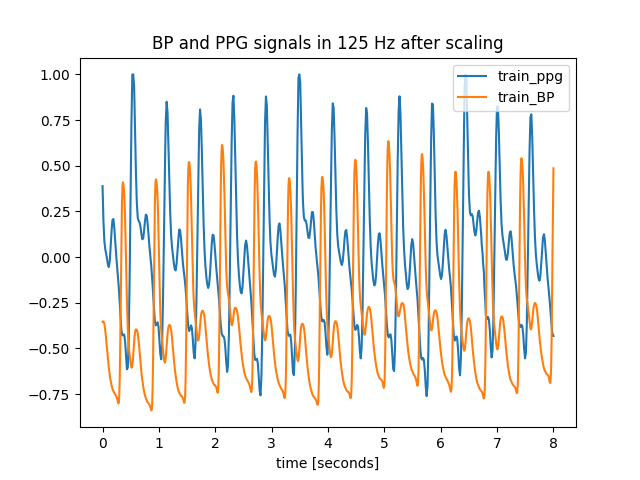
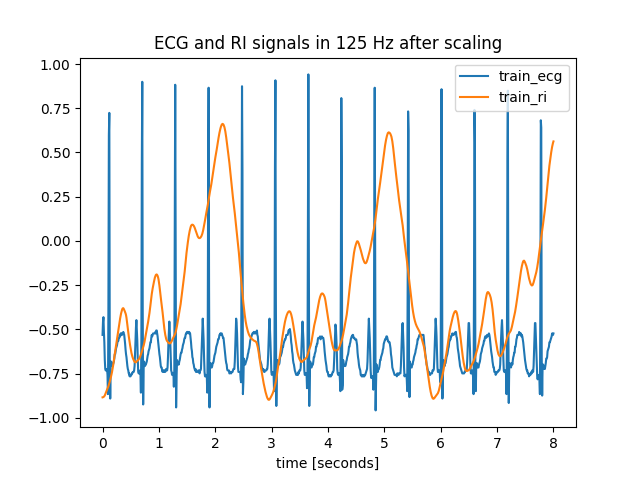
Moreover, we performed several trials for estimation BP based on all the signals in the database: PPG, RI, ECG. Those trials were made before the batches optimization that was done for the NN, which means 3 input to the NN, each contains a long vector of train data. As a result, the NN could not estimate BP in a good way – the estimation was pretty bad. For comparison, the same NN with the same features that got only PPG, estimated pretty well the BP of the patient.

All of the above led us to use the PPG vital as the only input to the NN.

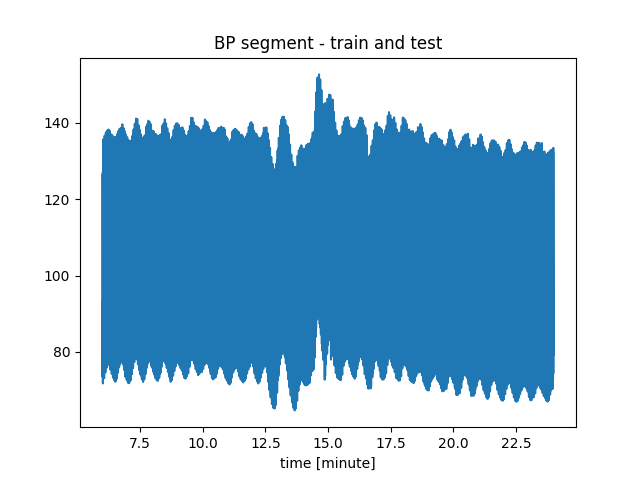
We took patient 2728529-6532 (which has various BP and doesn’t have noisy segments) to perform this experiment.



In order of use all 4 signals, we converted RFG and RI to the same sample rate as BP, PPG- 125 Hz. The scaling in the figure below is essential for NN inputs, and we discuss about scaling and normalizing our data in chapter \_\_\_\_\_



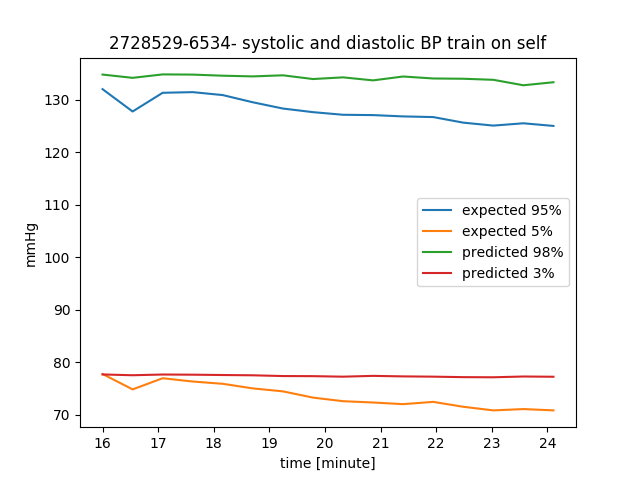
Division for train and test segments:



**test**

**train**

Throughout a 9 minutes of train that included all the signals above as inputs to the NN, we performed a test over the next 9 minutes:



At this point, comparing the estimation we got based only on PPG signal, we realized that ECG and RI don’t contribute to the learning process of the NN.

We tried several trainings with PPG, which led us to optimize the NN and realize it's limitations. The main limitation was choosing a patient from the data base. We chose a patient's signals that stood in several criteria:

- BP segments have to be diverse enough (include increasing and decreasing of BP).

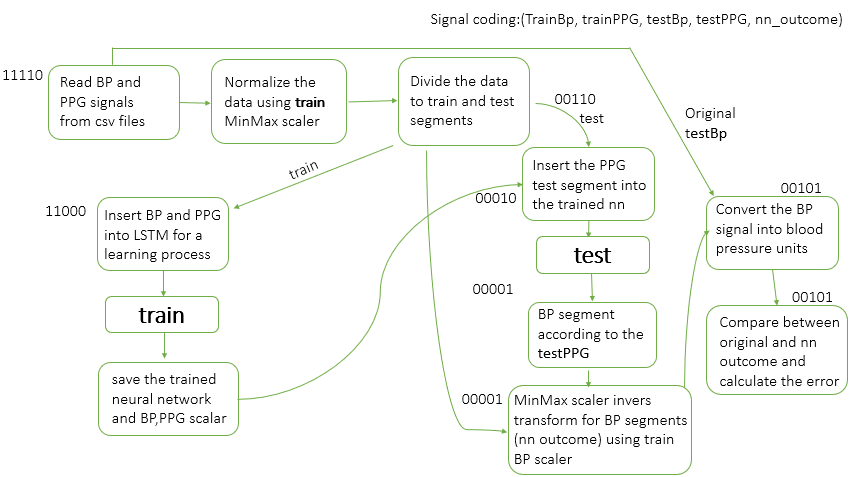
- both BP and PPG segments should be properly measured and without noise. Our data base is raw and the signals come from critically-ill patients, so the data can contain exceptions and spurious signals that can confuse our NN. The noise can originate from movement of the patient, treatment from the medical staff or measurement noise from the monitor or the recording unit. //לשפר את הניסוח

- we tried to choose a patient that was recorded long enough (every segment is 40 minutes approximately), so we could test the trained NN for different periods of time after the train period //2 hours after the train, 5 hours, 22 hours….

**4.3 – block diagram:**

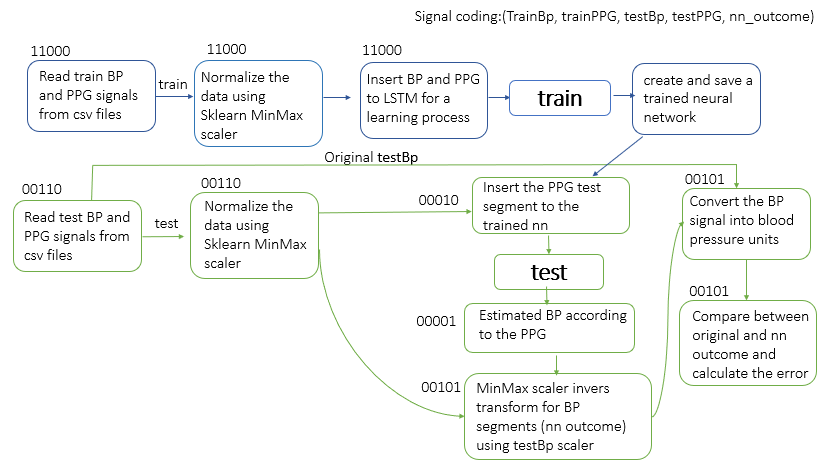
In the following diagrams we will show the signal flow from the database to the BP estimation:

**Train and test on the same patient:**



INVERSE , NOT INVERS ^

**Train on one patient and test on another:**



**Presenting the result:**

The output graph we will inspect to evaluate our NN contains a graph for systolic value (high line) and diastolic value (lower line). We used an average over percentage value for every 30 seconds. The 95% of the BP signal should represent systolic value and 5% of the BP signal should represent the diastolic value. The NN output and the expected BP (the expected signal is drawn from the data base) is on the same graph so we can compare them.

**5. Results**

**5.1 - Creation and optimization of the net:**

Creation and optimization of the net //dorin

Took an existing lstm and changed it to our needs -from where?

Talk about the nn structure-

Adam optimizer

Activation function- tanh

Loss function- MSE

What happens in every iteration

Add mean loss graph

Decide the nn characters with graphs to show

Without batch and then with batch

hidden layers and all- try different values and show results.

GPU batches- faster- explain why

As we began our journey of creating and optimizing our LSTM, we made a research about its features. We used an open source code of LSTM network creation; the link is given in the appendix. We chose Adam to be our network's optimizer. Adam is an adaptive learning rate method, based on stochastic gradient descent.

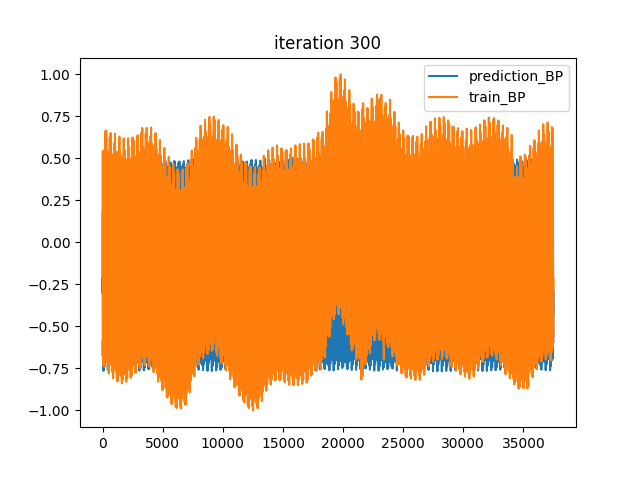
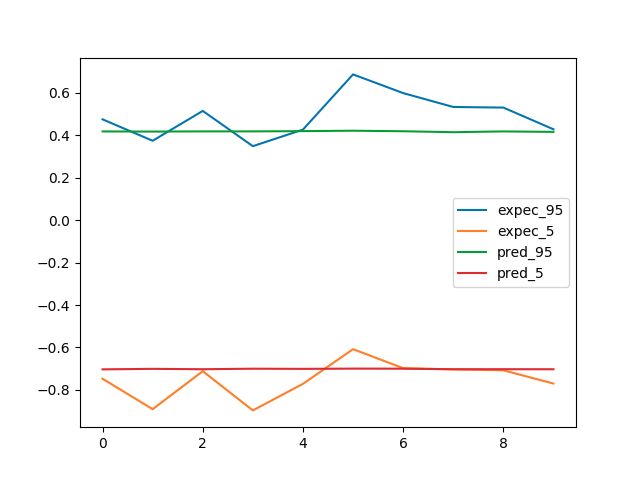
Maybe we don’t need to show all the nn characters?

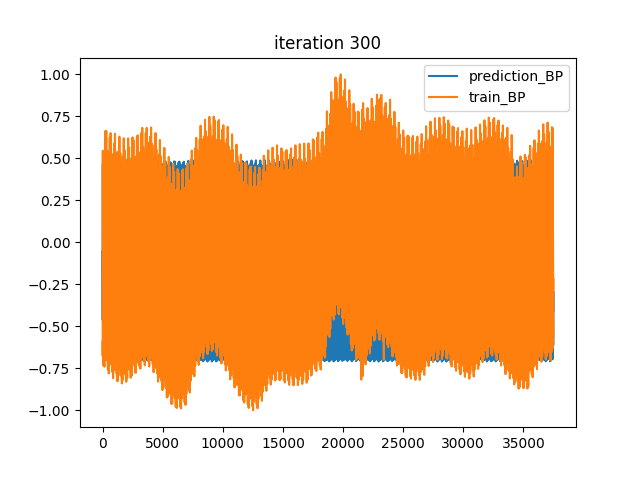
At first, we tried a straight forward approach – 1 big vector of 10 minute train set. We performed many trials for the network features – we changed the number of layers, number of hidden layers, amount of iterations, and step size. Bellow we will present an example of the network's outcomes along with changing one of its attributes- the number of LSTM layers.

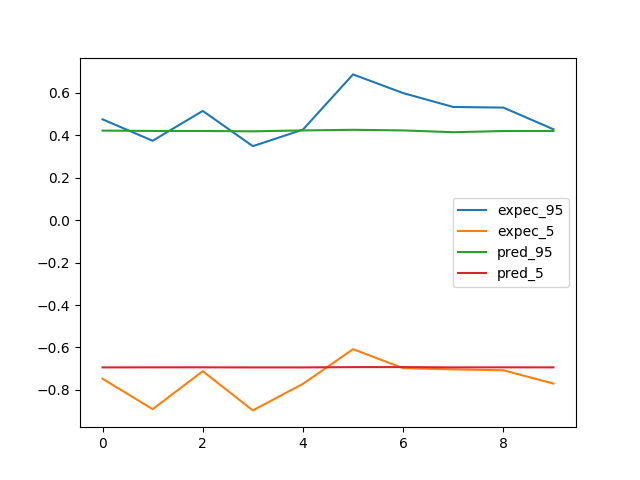
Network's features:

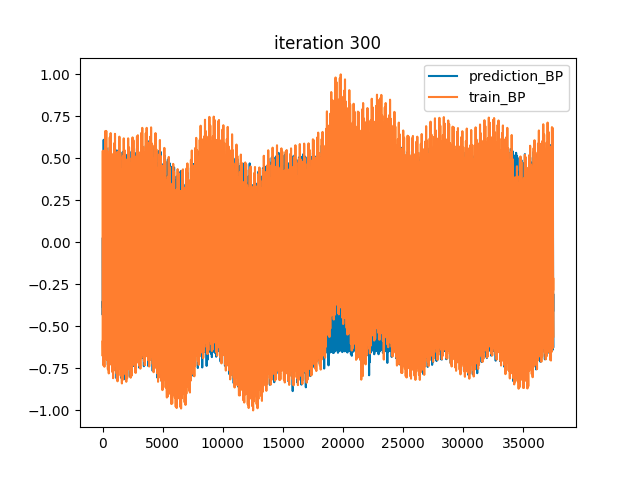
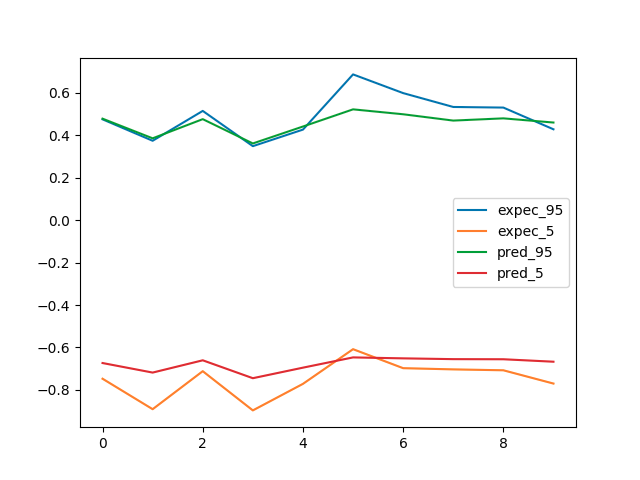
* Hidden layers number: 30
* Step size: 1e-3
* Number of iterations: 300

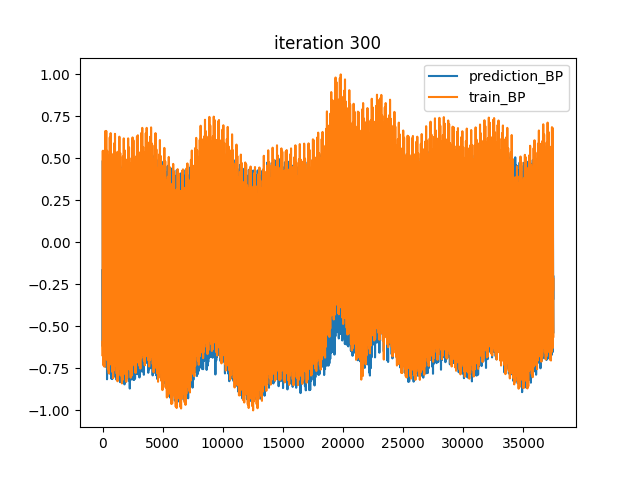
In the next graphs we will present the last training iteration over 5 minutes, and a test over the following 9 minutes.

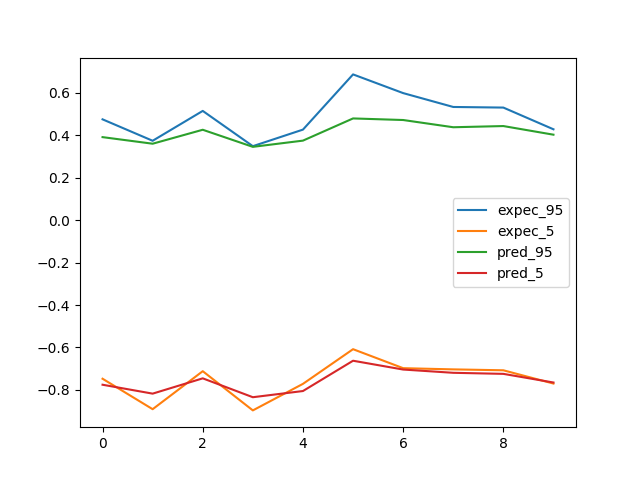
One LSTM layer:

Two LSTM layers:



Three LSTM layers:

Four LSTM layers:



We can see that as we add LSTM layers, the network's results are getting better.

Eventually, we were able to achieve good results, yet, the whole process of training took a lot of time – more than 1 hour for 10 minute train set.

In order to take advantage of the qualities of GPU and shorten the train process - we tried a parallel approach.

After several trials, we decided to perform the train procedure with 2\*\*16 samples, which are equivalent to approximately 9 minutes. We divided this train set to batches of 16 seconds, which means 32 batches. This way we reached improvement by 32 time than without batches (without batches means the whole train set in one vector). On one hand, using GPU can reach 2500 train iterations in 10 minutes. On the other hand, we utilized the memory of the LSTM for only 16 seconds.

The final parameters:

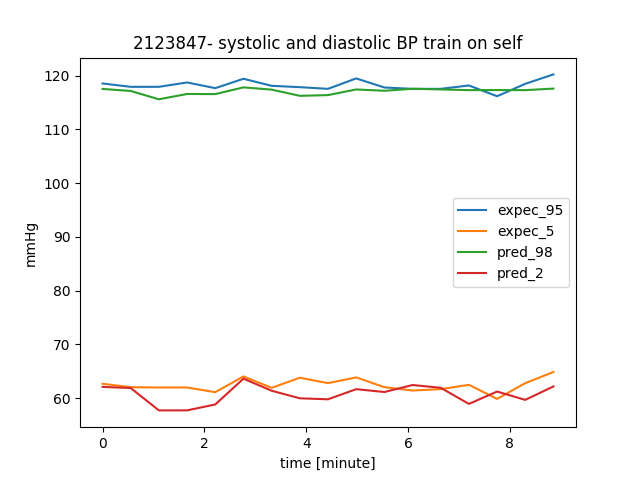
|  |  |  |  |
| --- | --- | --- | --- |
| **Layers** | **Hidden size** | **Step size** | **Iterations** |
| 2 | 12 | 1e-3 | 2500-3500 |

**5.2 - Training on a single patient and predicting BP of the same patient**

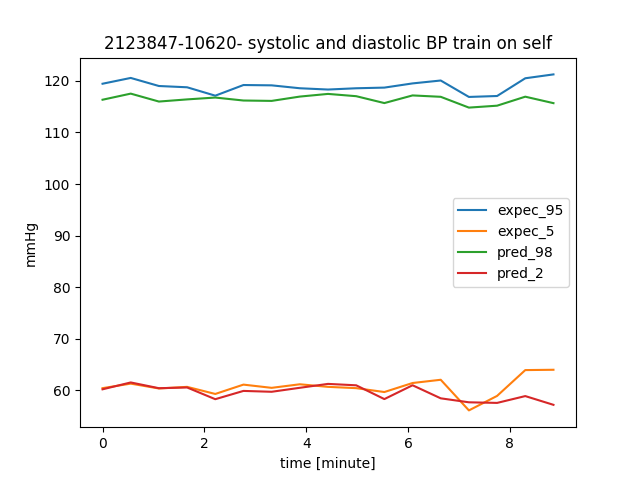
We succeeded in learning a patient's behavior and infer about future and past times (of the certain patient). We took a train set that was various enough, we made sure that the BP segment contained increasing and decreasing values in BP. The variation is important in the train segment because the NN can't produce values that it hasn’t seen in the train part.

With 10 minute of train segment, the NN succeed in estimating future BP based on PPG segment.

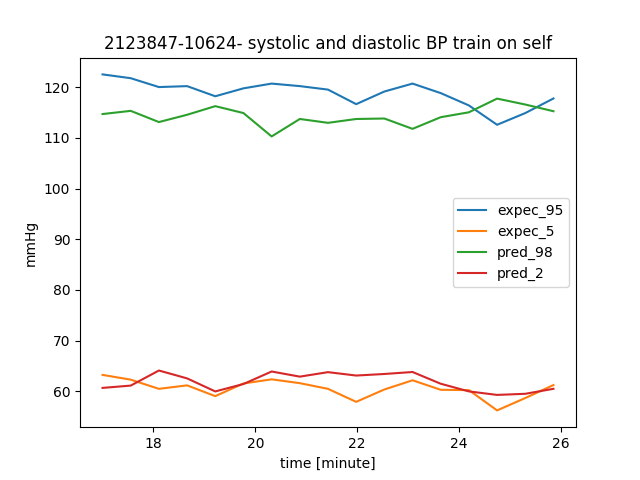
For example, we trained on patient 2123847-10616 for 9 minutes:



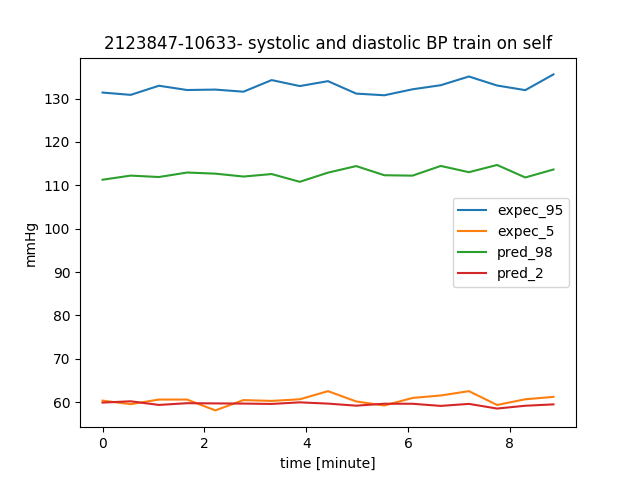
Test after 2.5 hours:



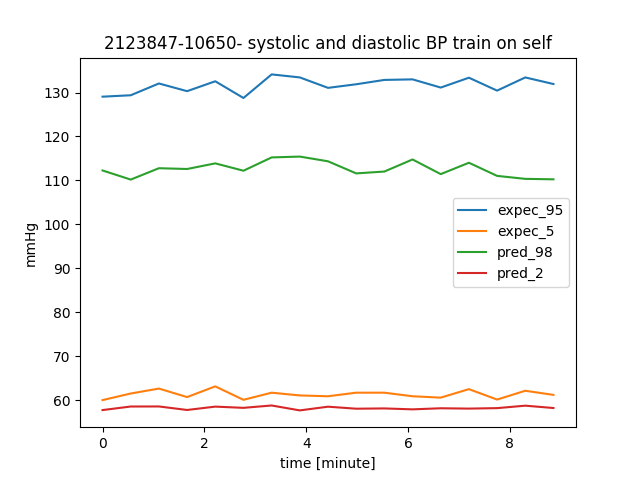
Test after 5.5 hours:



Test after 11 hours:



Test after 22 hours:

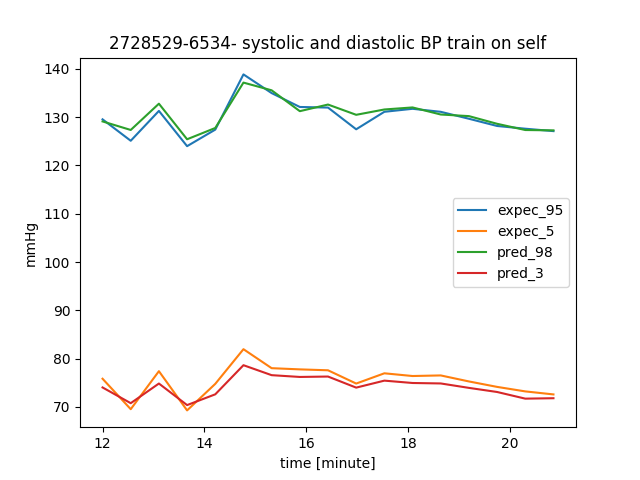


We can see that sometimes the NN estimates BP values pretty well in the next few hours after the train, and sometimes the estimation isn't that good. All of the results are detailed in the appendix.

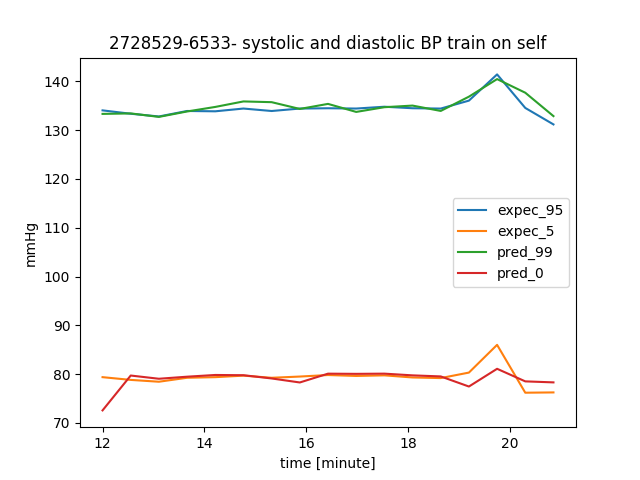
We assume that the faults in the estimation are due to changes in PPG segment. The patient's physiology can change sometimes due to medications or medical condition. The change in PPG can also arise from the way the clip~~s~~ is suited on the patient's hand (figures in the appendix).

We discovered that the NN can estimate BP in the past – which means that we managed to learn a patient's physiology.

Train on 2728529-6534:



Test on a segment 27 minutes earlier (before the train):

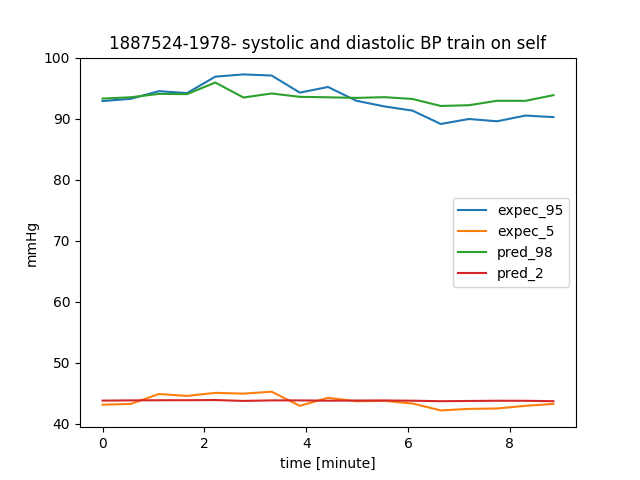


In chapter 7.2 (appendix) we can see several examples of the NN we created, that trained on 9 minutes segment on a single patient. For some patients, the estimation for the next (and sometimes previous) hours is good, and sometimes has some mismatches (פספוסים). We assume that these mismatches occur due to changes in PPG shape. In 7.2 we can see that sometimes these changes affect the bad estimation, and sometimes the PPG looks similar and it can't explain this assumption.

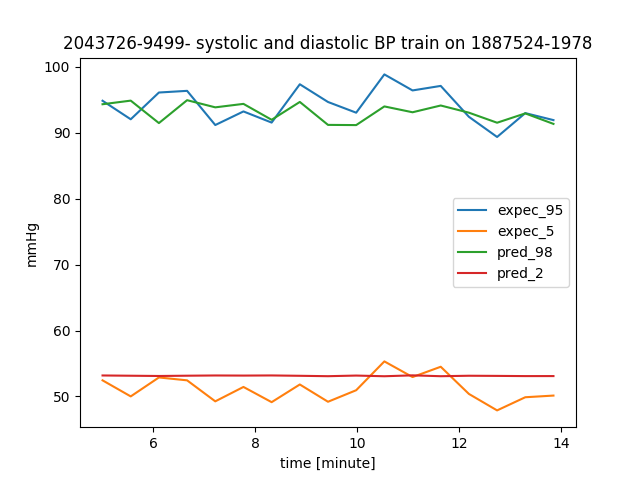
**5.3 - Training on a single patient and predicting BP of different patients**

We examined a NN that was trained on a certain patient (10 minute of a various segment) – perform an estimation for another patient based on the other patient's PPG. The data normalization has been done with the test segment.

We trained a NN for 9 minutes on 1887524-1978:



We performed a 9 minutes test on 2043726-9499:



We can see that the estimation isn’t good – the NN couldn’t estimate the diastolic at all. The systolic estimation has a certain delay in the estimation, and the values are not reaching the expected minimum and maximum.

We assume that bad estimation of the diastolic is because the NN was exposed in the train set to values in range 40-50, and here in the test patient there are values in range of 45-55. This outcome is reasonable, because the NN can't deliver values that it wasn’t exposed to during training. Moreover, we can see that the BP of the test patient is more dynamic in comparison to the train patient. In chapter 7.3 (appendix) we can see several examples of bad estimation even when the range of BP value of the test segment consists of the range of BP value from the train segment.

**6. Conclusion**

6.1 discussion:

We've tried to create a neural network that with studying approximately ten minutes of PPG and BP coordinated signals, it will be able to evaluate the BP only by PPG signals in the next period of time. We couldn’t achieve an unambiguous conclusion concerning the successfulness and accuracy of the BP evaluation as it was changing along with the distance of time from the training set.

We've learned that in order for our neural network to successfully predict the BP, the train set should be as diverse as possible, reaching high and low values.

Upon all of the above, and the research we made, we developed an algorithm for measuring a patient's blood pressure, in the less invasive way possible.

As the patient arrives to the hospital, the medical team will take his BP using a catheter to an artery. That phase must be performed for our nn scalers to be calibrated, and in order to get a hold of the patient signals shape and pattern.

The invasive BP measuring will continue until we get a divers BP signal.

Through the whole process, his PPG measurements will be processed together with the BP, as an input to the neural network. That part is the network's learning process. After the catheter is out, the PPG signals will continue to flow as an input to the network, and the output is the estimated blood pressure. In order for the output to be more accurate, every hour (more or less- depending on the patient's state), the patient's blood pressure will be taken using a sphygmomanometer, the results will be inserted to the network for helping calibrating the scalers.   
with the help of the algorithm, we will achieve minimal exposure of the artery, and yet an accurate blood pressure measurement.

The algorithm we created has some flaws. First, we need to take into consideration that the way the patient wears the PPG clips can affect the signal's shape. As a result, the estimated BP will be affected as well, since it is directly calculated from the PPG.

Second, the goal is to minimize the exposed artery time, by that we get a shorter continuous BP sampling. In a shorter signal, we get less diversity of the BP which impairs our network results.

**6.2 Future works:**

Intro

First, we noticed a connection between the PPG wave structure and blood pressure changes. We think that investigating this connection can lead to better results.

Second, we believe that the biggest flaw in our algorithm is coming from the lack of diversity in the train set. For future project, we recommend trying to perform data augmentation to the input signals, so we can achieve various BP and PPG values, but still conserve its medical correctness. In that way, we might be able to predict radical BP values, and minimize the mean error.

Third, it is interesting to use various patients as a joint input to the network, and see how another patient's BP estimation is affected. In addition, to save runtime of network training, we can use a trained network of another patient. That way, the nn weights will already be calibrated and normalized.

**7. Appendix**

**7.1 – results from synthetic signals //Dorin**

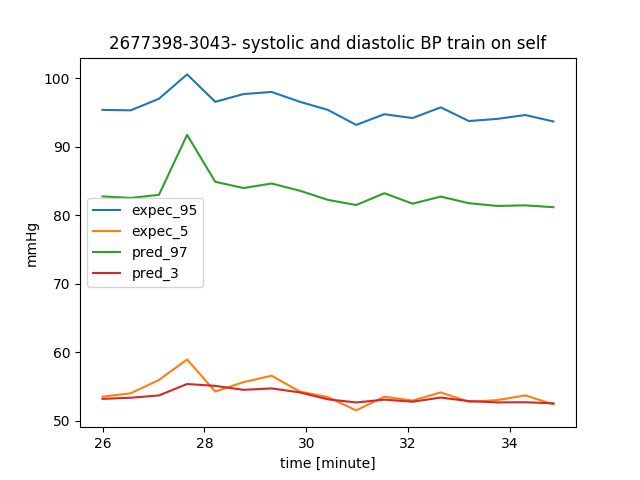
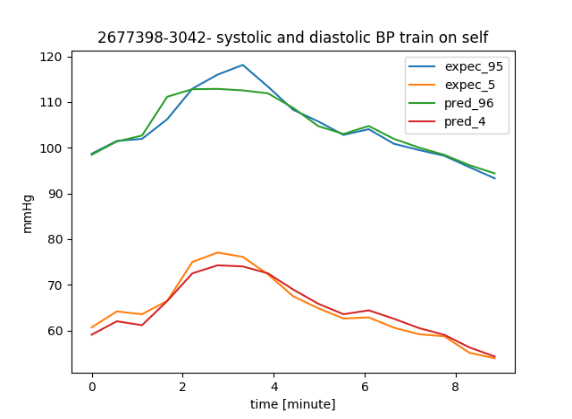
**7.2 – results from train and test on the same patient**

**Patient 2677398:**

**Test**

**65 min later**

**Train**

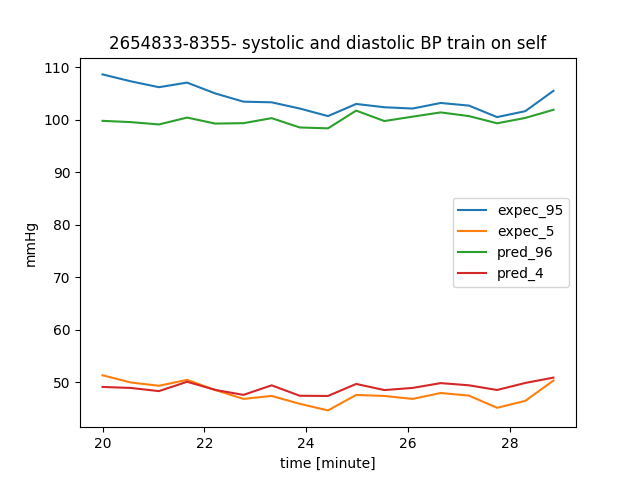
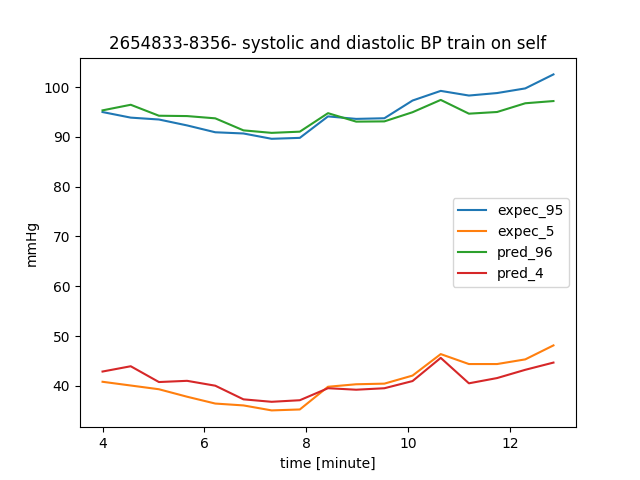


**Patient 2654833:**

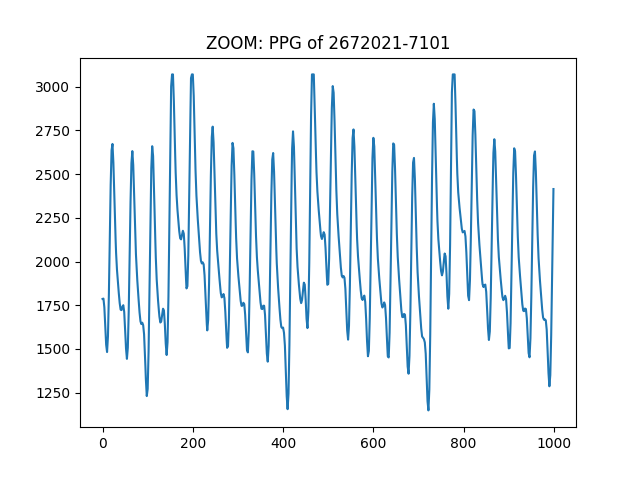
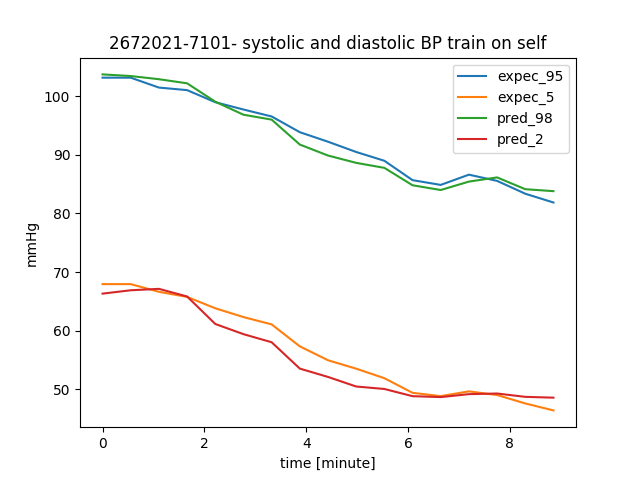
**Test**

**24 min earlier**

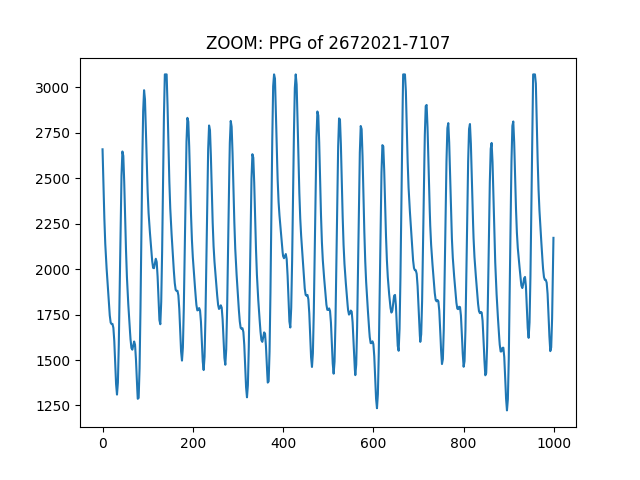
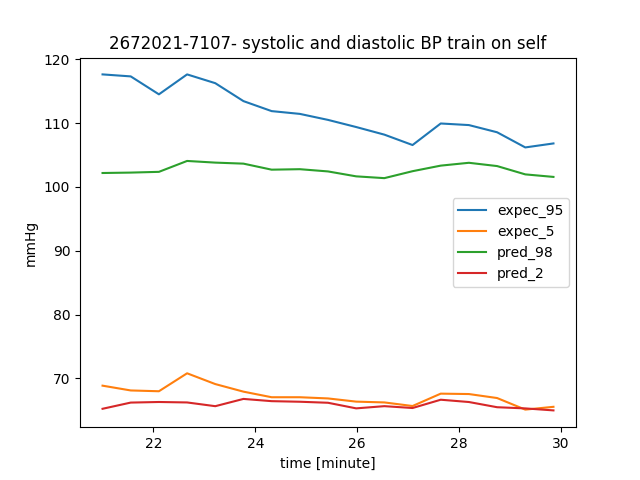
**Train**



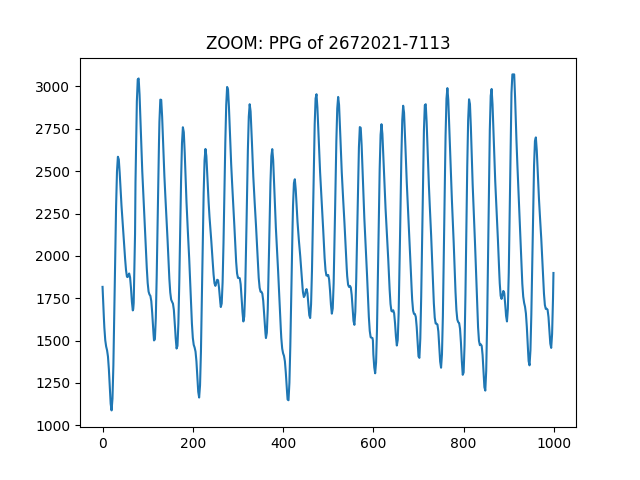
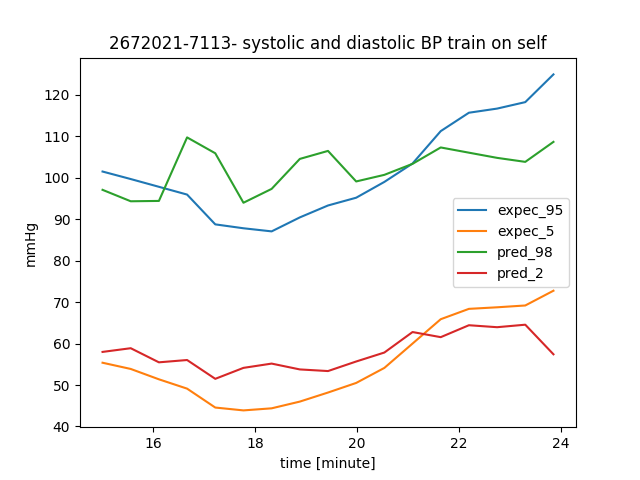
**Patient 2672021:**



**Test 4 hours later**

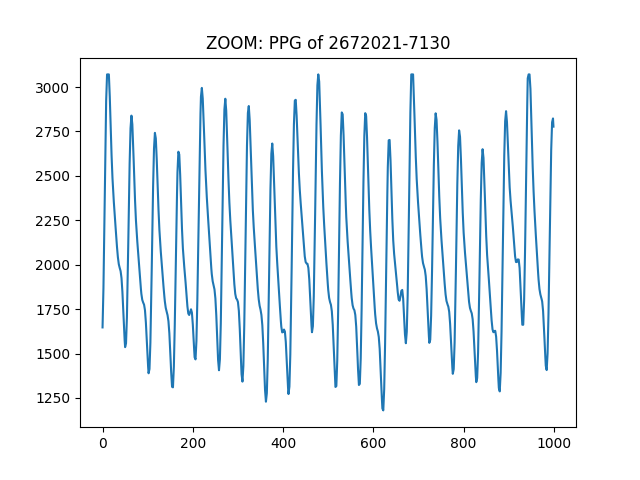
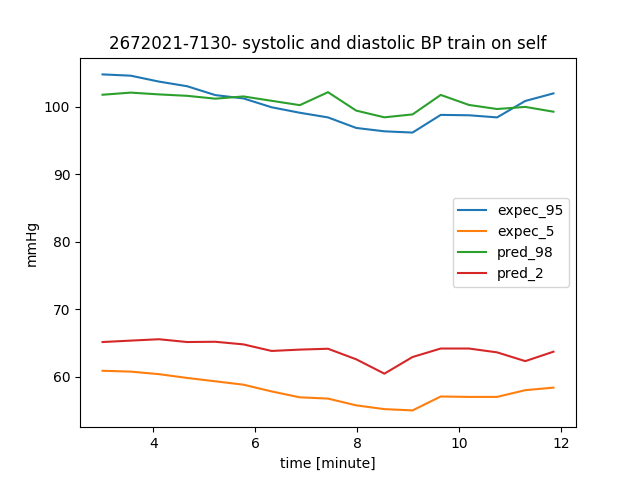


**Test 8 hours later**

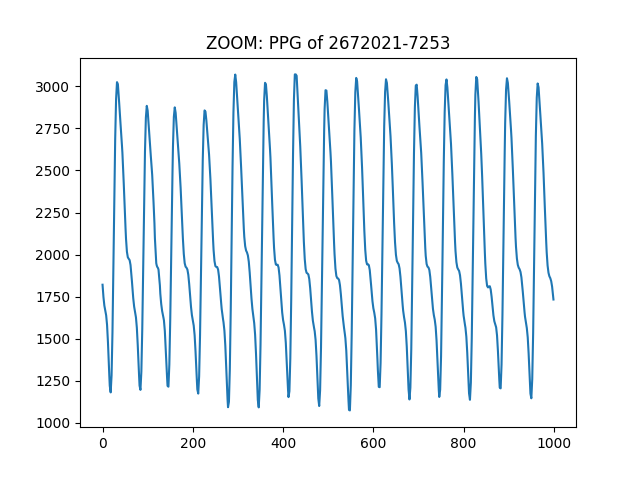
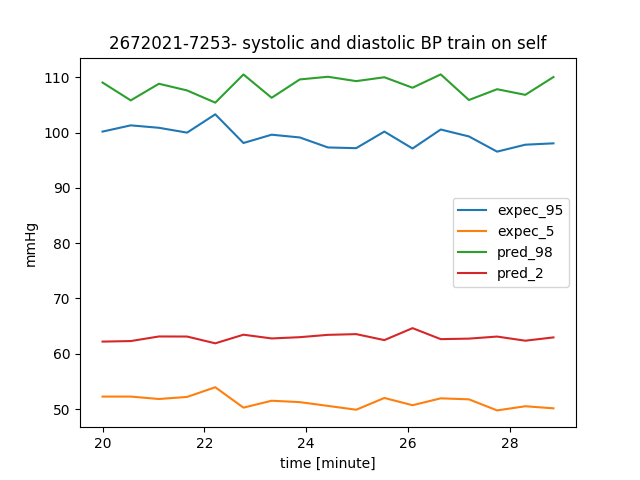


**Train**

**Test 19 hours later**



**Test 101 hours later**



**7.3 – results from train on a single patient and test on another patient:**

|  |  |
| --- | --- |
| **Train on 654833-8356** | **Test on 2043726-9499** |
| **Train on 1887524-1978** | **Test on 2043726-9499** |
| **Train on 2677398-3042** | **Test on 2728529-6534** |

In the chart above, we can see that a NN that trains it's weights according a single patient, can't estimate BP of another patient, based on the other patient PPG.

**8. References**

1. SD Goodfellow, A Goodwin, R Greer, PC Laussen, M Mazwi, D Eytan, Non-invasive Blood Pressure Estimation Using Physiological Signals Acquired by the Bedside Monitor: Insights from a Pediatric Critical Care Setting , 2019

2. Open source [code](https://nipunbatra.github.io/blog/2018/denoising.html) that was used to estimate a signal out of it's noised sample. We based our code on this example.