Digital Biomarkers HS24
Examination

1 Content

2 General

- Bring figures for each exercise (screenshot each output of the exercise)
- Code is not expected to be presented
- Comment on code if he shows you some, but don’t bring your own code
- Why does the output look like, is it reasonable?
- Understand the distance measurements
- ||y1-y2||→ what is it? Distance
- ||Y2-lambda * y1 || → what is lambda, what is vector, what is scaling

Per Exercise 1 Slide, answering questions about how did you solve, what do you see..

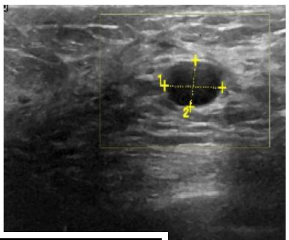
3 Imaging

- What are textural features and how could you implement?
- What should you be careful with for developing a model?
- What should be the size of data, How should you split the dataset?
- Is the dataset good for the question, what could be enhanced, [data quality]
- Why did you choose this feature? Why did you think another feature was not so good?
- What did you do first to use this data?
- If I only had half the data, would the feature still have worked with the methodology you did?

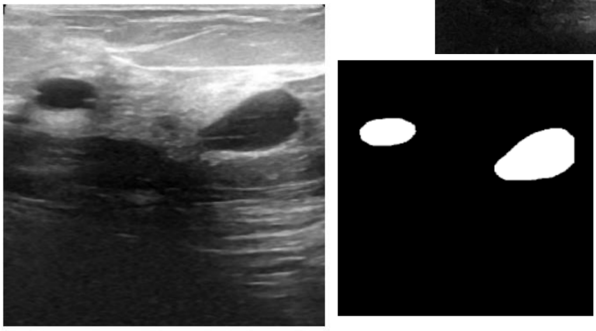
3.1 Exercise 1

3.2 Part 1 – data analysis

- Images (png) structured into folders with the three classes: benign (437), malignant (210) and normal (133).
- Images in normal are either not classified or no tumor existing, thus not including them into my analysis
- Images are loaded with keras package load_img(), 2D, grayscaled
- Some images have markers, which are excluded, ending with 177 for both classes (benign & malignant)

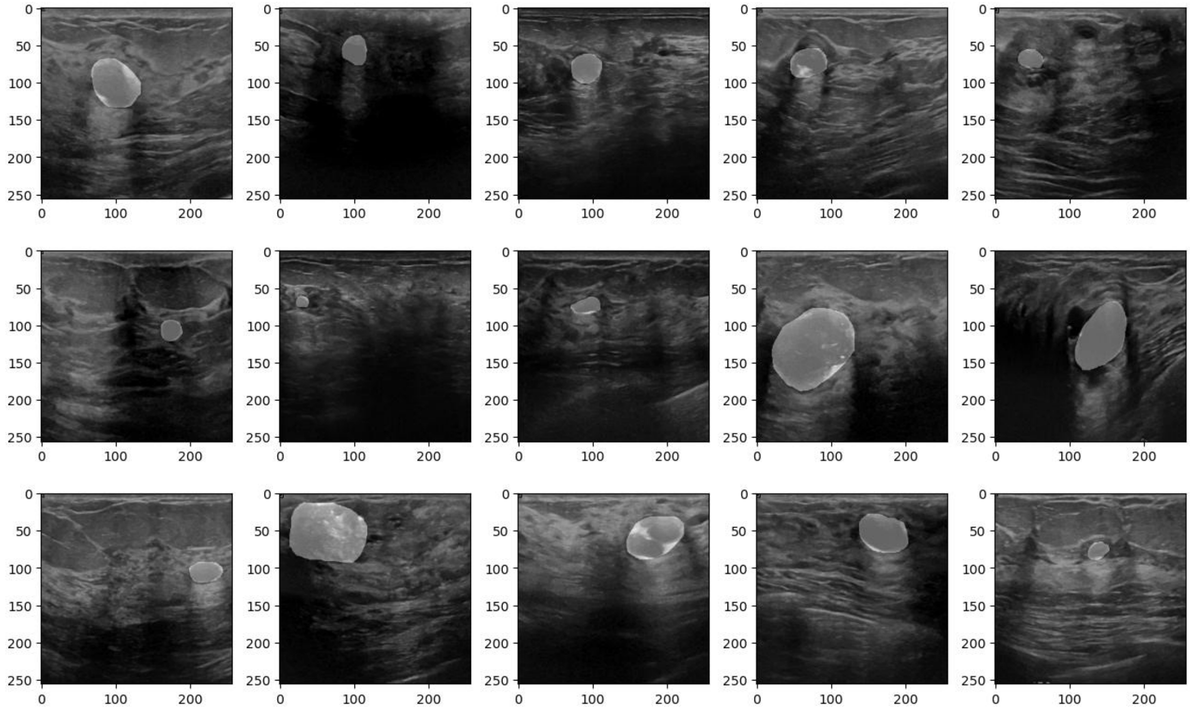


- Some images have double_masks (but not part of the 177 samples),

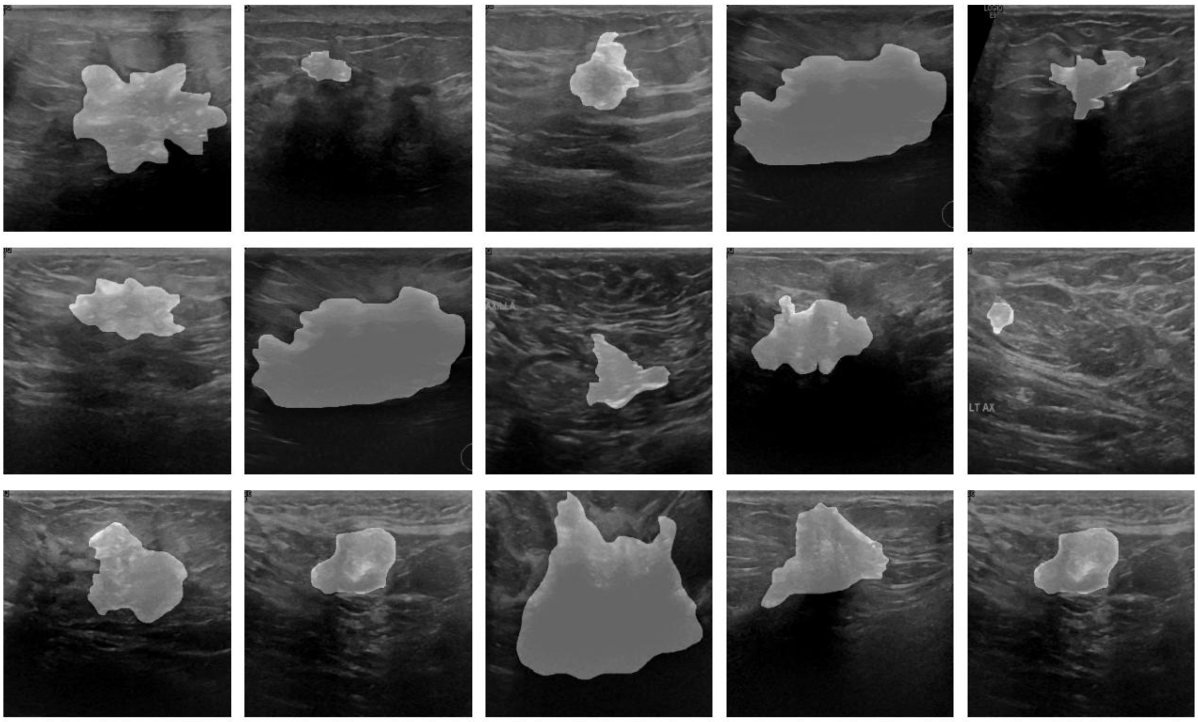


3.3 Part 2 – feature extraction

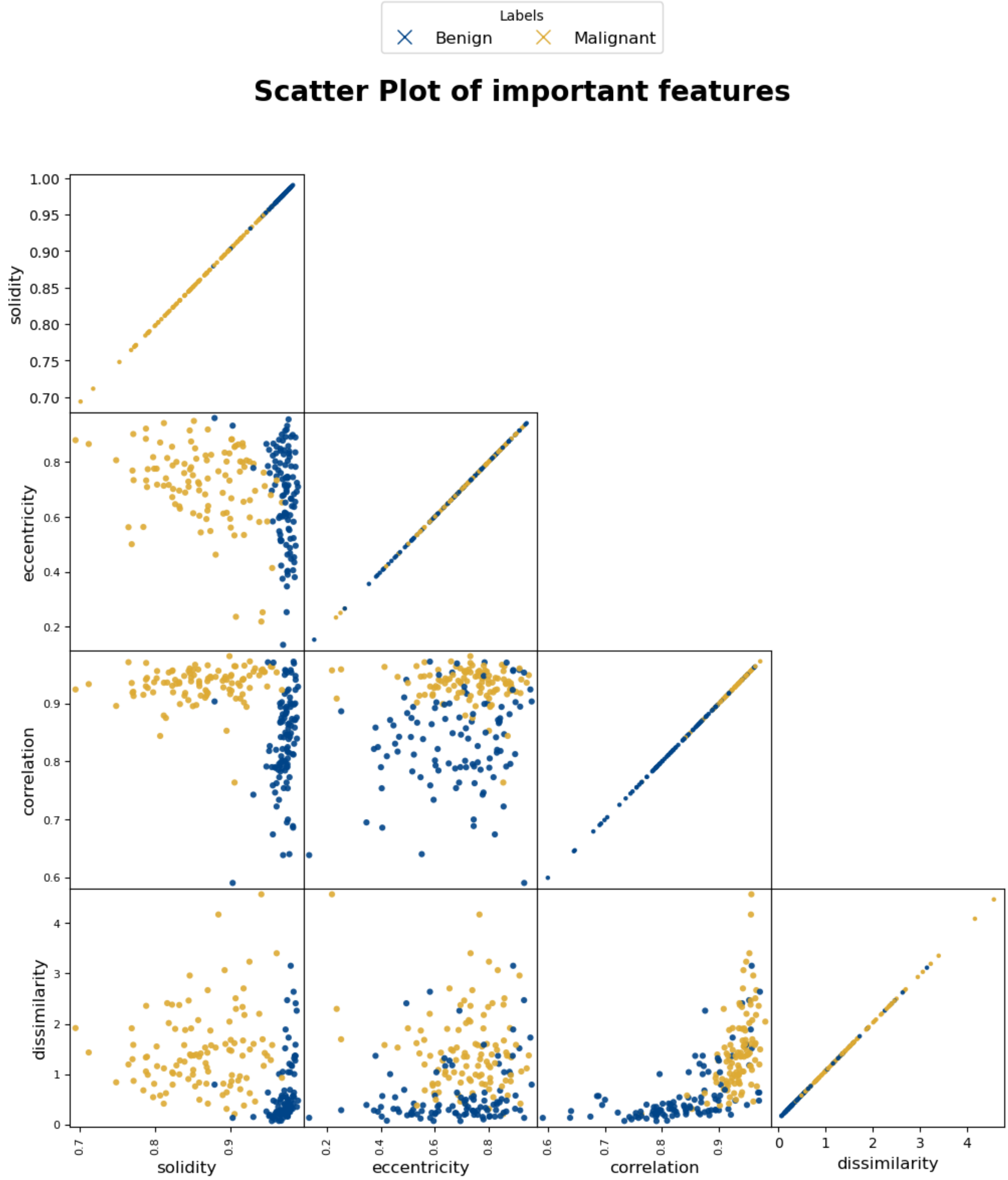
3.3.1 Benign



3.3.2 Malignant



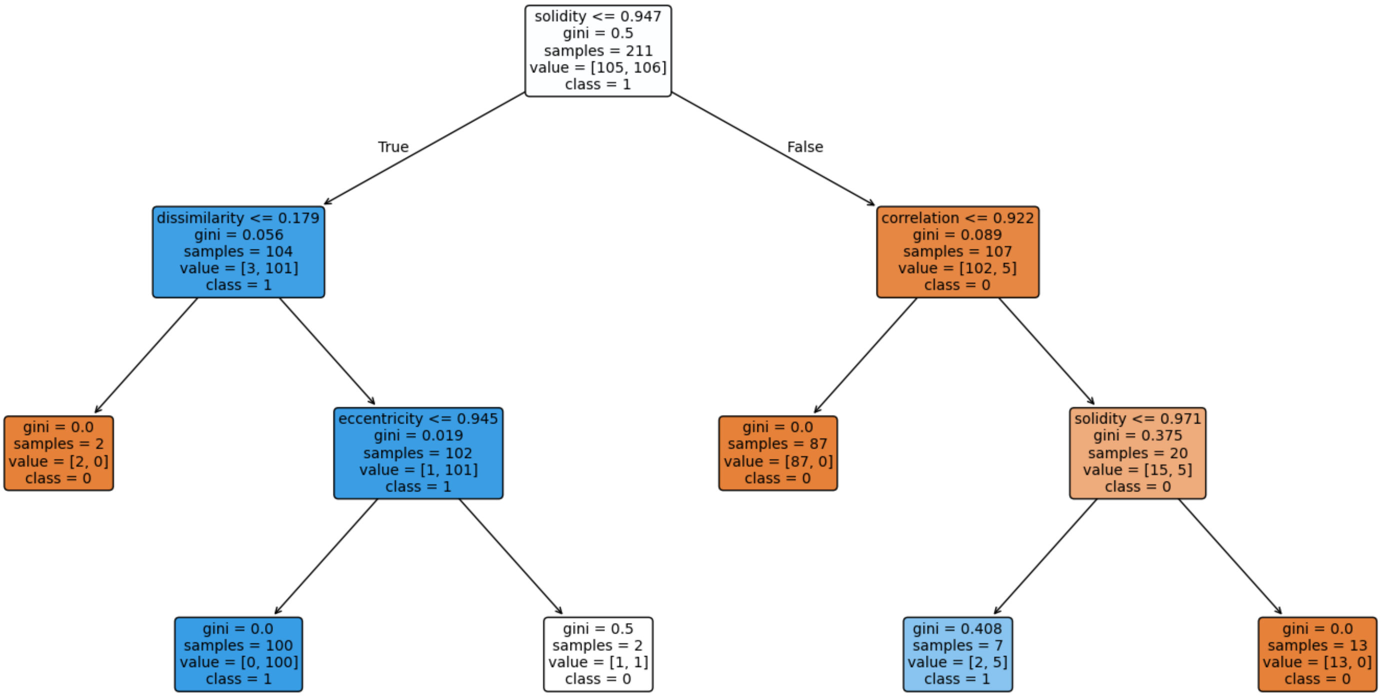
- Which features are unusable for this modality?
    - Features like perimeter, area, diameter are not usable due to different image sizes
- What does the literature say?
- In what format do I need the features for the classification?
    - Features that are not bound to size
    - Features that work on greyscale images



3.4 Part 3 – classification

- Which model should I choose?

Decision Tree Visualization



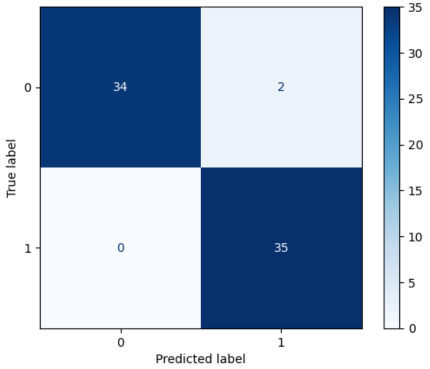
- Is this good result based on the specific train-validation split?

  Number of train images: 211
  Number of validation images: 70
  Number of test images: 71

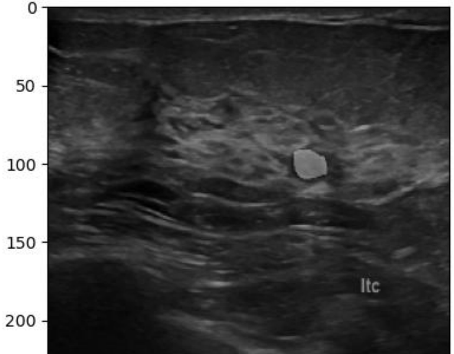
  - No, in my point of view the masks are highly biased, because when searching through the dataset, malignant masks are way more detailed than benign.
  - Based on this property (having much circular masks) of benign, a very simple and easy to interpret model has been chosen

- Which features?
  - Solidity (ratio, malignant irregular boundaries)
  - Correlation (texture feature, lin. dependency of gray levels in neighborhood)
  - Eccentricity (shape-stretching)
  - Dissimilarity (GLCM, measures relationship of neighborhood pixels)

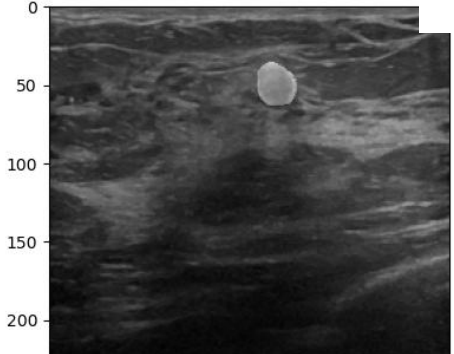
Confusion Matrix



Index: 10
Solidity: 0.94, Eccentricity: 0.55
Correlation: 0.94, Dissimilarity: 1.78



Index: 32
Solidity: 0.96, Eccentricity: 0.80
Correlation: 0.96, Dissimilarity: 0.39



4 Signal Processing

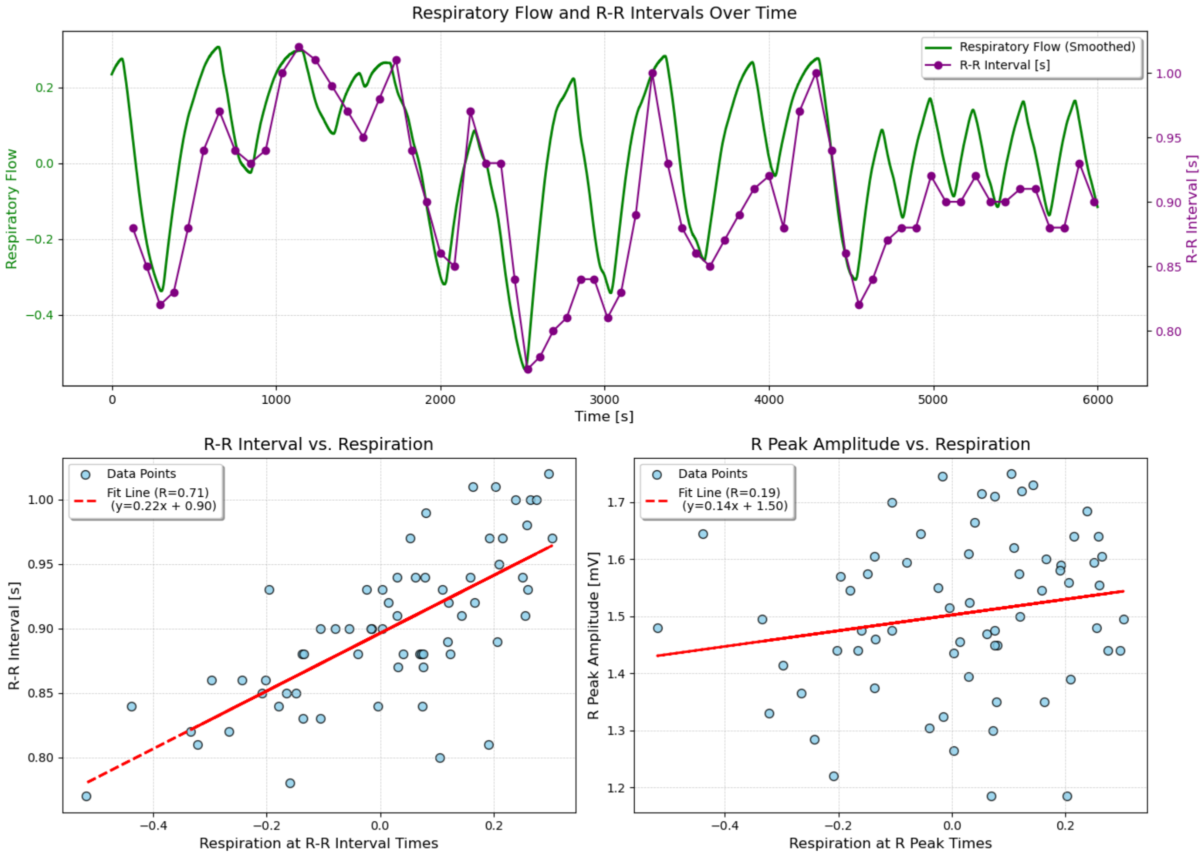
4.1 Exercise 1 – RR-Intervals & Respiration

How did I solve it?

The R-R interval is the time interval between two consecutive R-Waves (peaks) in an ECG signal. (It reflects the time between two successive heartbeats)

Questions:

- Is there a correlation between the rr_interval and the respiratory flow?
- Is there a correlation between the r_peak and respiration?
- How did you extract the r-peaks?
  - Setting the threshold of the find_peak function to height = 0.8, so that only the highest values found to be a peak.
  - Algorithm looks at each datapoint and checks if it is a local maximum. Local maximum, is a point where its value is greater than its immediate neighbours. Only peaks greater than 0.8 height are considered to be valid peaks.
  - y[i] > y[i−1] and y[i]>y[i+1]
- Assume that you have to identify the position and height of the t-waves from an ecg, what are possible difficulties and how do you handle them?
  - T-Waves may overlap with the QRS-complex (if the heart rate is very fast or unhealthy), apply filtering
  - T-Waves can vary in shape (flat, biphasic, or inverted) depending on health conditions = finding the peaks is harder. Solution: template matching, clustering techniques.
  - Noise (muscle movement, baseline wander). Solution: Low-pass filter, high-pass filter to address high frequency noise



Observations:

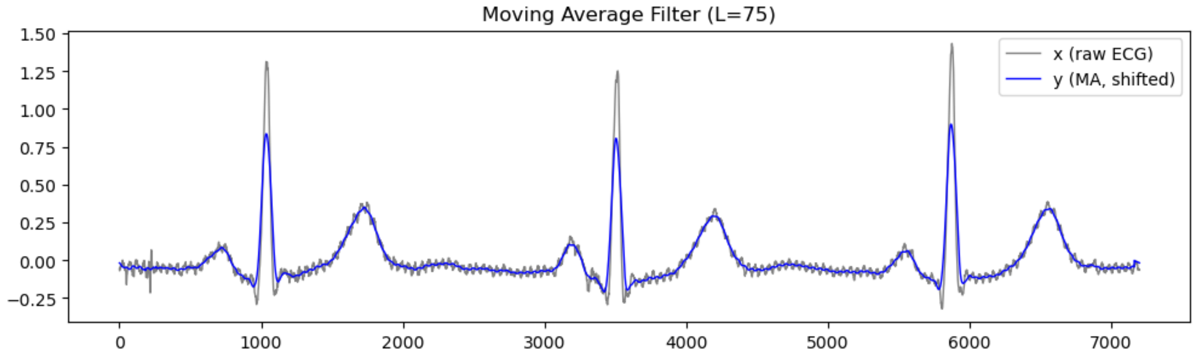
- Top Plot: green-curve shows respiration flow, purple markers indicate the time between two consecutive heartbeats (R-R Interval).
    - Time between heartbeats vary a lot 0.8 – 1 second, with mean of 0.9, relating to heartbeat of 67 heart beats per minute (bpm)
- Bottom Left: Scatter plot shows R-R interval (time between heartbeats) against the respiration at R-Peaks times.
    - Positive trend is visible, higher R-R intervals correlate to higher respiration values
    - Slope: 0.22403276883896822
    - Intercept: 0.8963000648782656
    - **R-Value**: 0.7080134252087623
    - P-Value: 2.965662696088793e-11
    - STD Error: 0.027932328978557863
- Bottom Right: Scatter plot shows relationship between R-Peaks amplitudes (voltage measured at the R-Peak, electrical activity during heartbeat) and respiration at respiration R-Peak times.
    - Weak positive trend, means minimal dependency between respiration at peak time and the measured electrical activity at R-Peak amplitudes.
    - Slope: 0.1375911872433881
    - Intercept: 1.5022731152409803
    - **R-Value**: 0.18533343877202338
    - P-Value: 0.13626940895025494
    - STD Error: 0.09119206718113497

4.2 Exercise 2 – Moving Average

How did I solve it?

Applying an lfilter with

- L = 75 (last 75 datapoints),
- 1 = FIR-Filters,
- X = signal input
- Delay = (L – 1) // 2 (delay introduced is half the filter-length)



Observations:

- Grey curve is the raw ECG-Signal, having sharp peaks representing the QRS-Complex
- Blue Curve is the moving average filter (with L=75)
- Noise and small fluctuations in the signal are suppressed.
- R-Peaks, P-Waves, T-Waves are preserved in the signals but slightly blunted

MA is commonly used for Noise Reduction, Feature Detection (R-Peaks, T-Wave) amplitudes becomes easier.

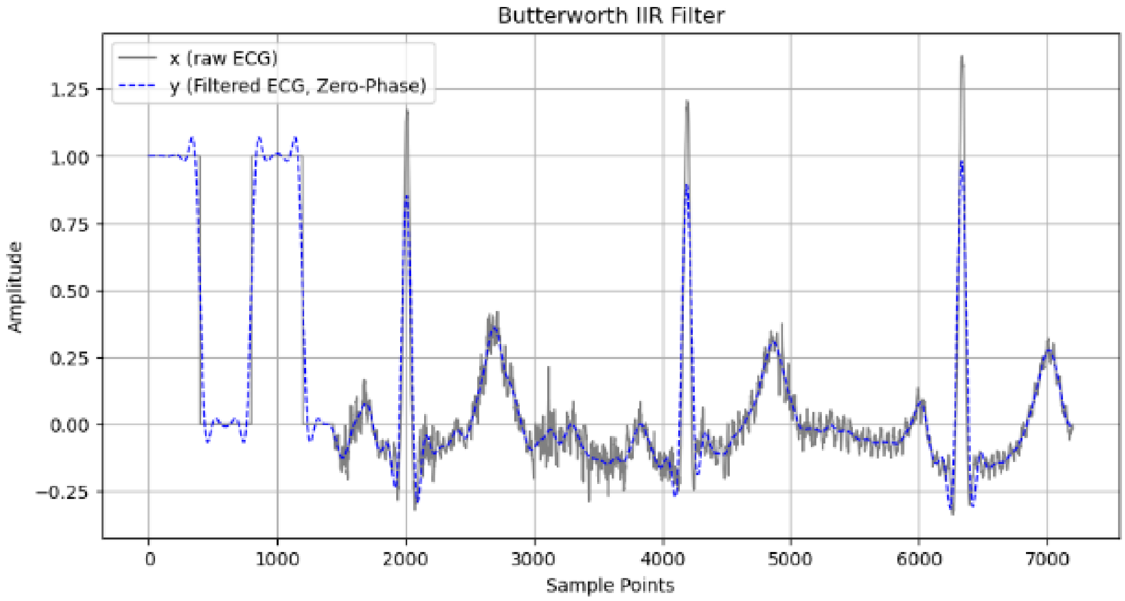
4.3 Exercise 3 – Filters

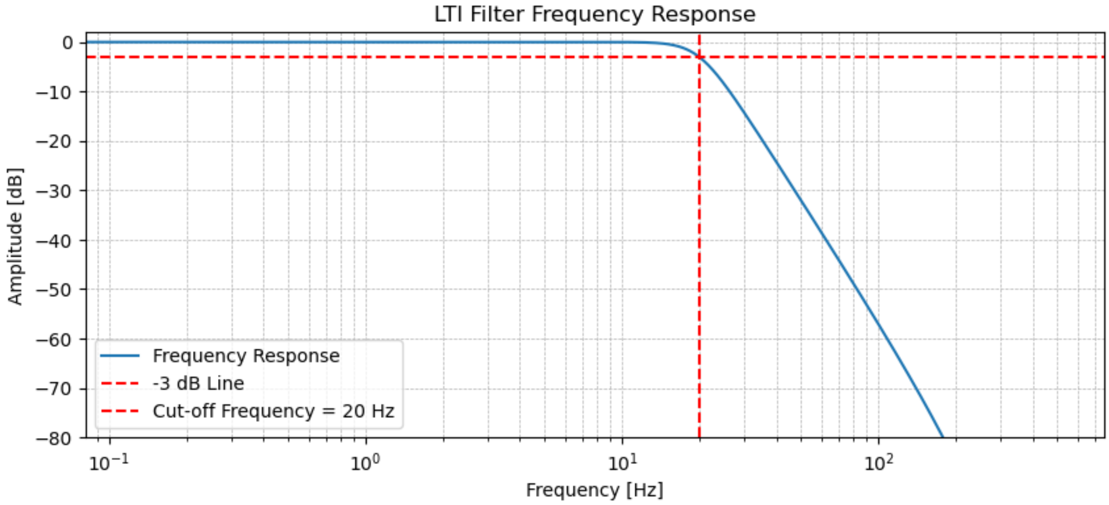
4.3.1 Butterworth Filter

How did I solve it?

Applying an IIR-filter with order = 4, fc = 20Hz & zero-phase filtering, fillfit to apply the filter forward and backwards. By doing this the distortions introduced in the forward filterting are canceled out in the backwards pass.

Cut-Off: Butterworth filters are designed to introduce minimal distortion in frequencies below the cut-off rate: 20Hz





Observations:

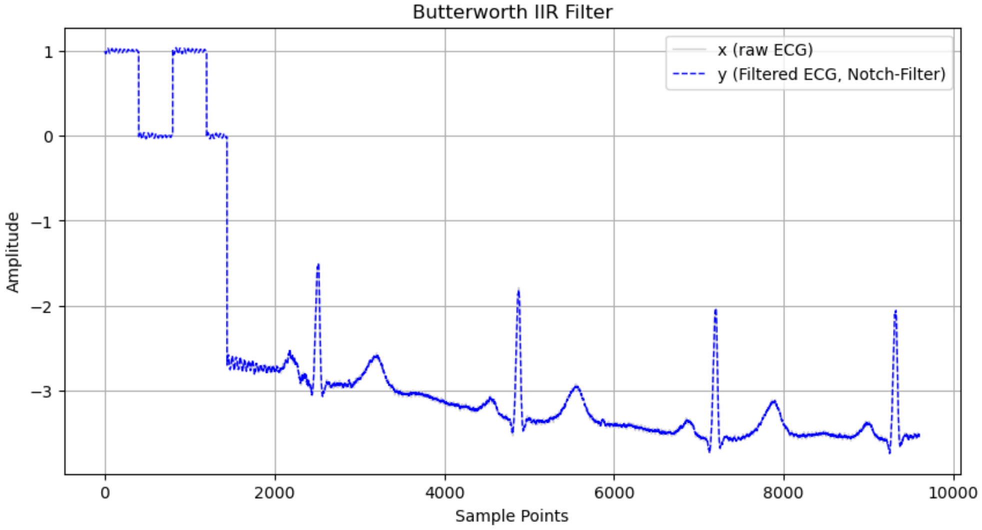
- Raw signal is noisy (visible by the high-frequency noise)
- Effect of the Butterworth filter:
  - Achieved, Reduce high-frequency noise (it becomes visible in the low-amplitude regions (P & T-Waves)
  - Achieved, The QRS Complexes are preserved yet somehow a bit reduced by 0.30 (filfit) & 0.26 (lfit), the sharp peaks as well as the P & T-Waves)
  - Achieved, Filter does not shift thanks to zero-phase filtering (shifting)
  - Unable to filter the strong noise in the beginning (~1200 samples)
    - Padding the first few artifacts could mitigate the starting behavior

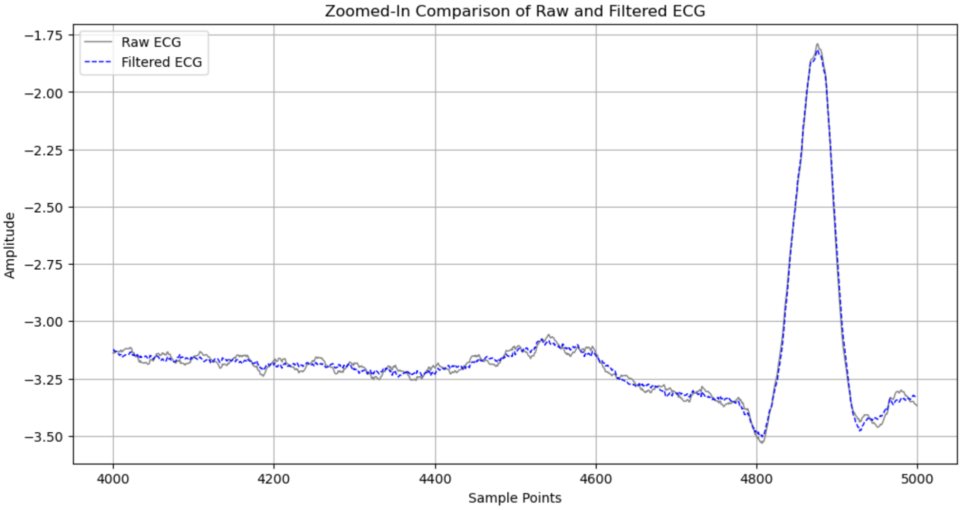
4.3.2 Notch Filters

How did I solve?

F0 = 50 (frequency to remove), Q = 30 (quality Factor, defines filter bandwidth relative to f0)

Notch filters are designed to remove specific frequency components.





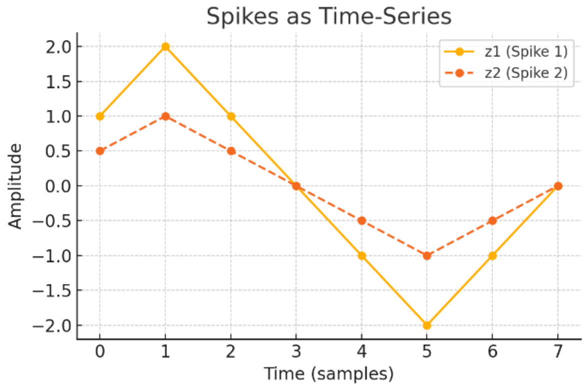
Observations:

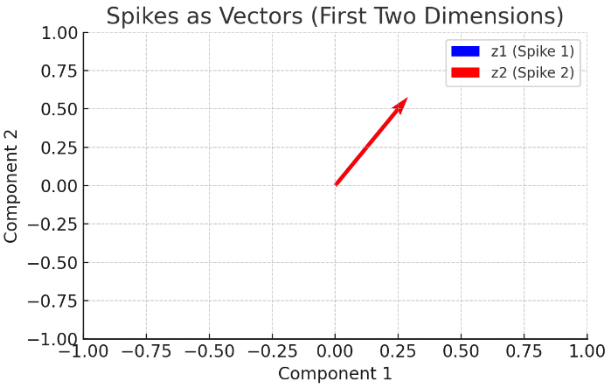
- Noise removal, in low-amplitude regions
- QRS-Complex fully preserved, thanks to the high quality factor.(ECG usually (1-40Hz) are not affected by notch-filter
- Amplitude Reduction: -0.04

4.4 Exercise 4 – Neural Spike detection

4.4.1 Detection

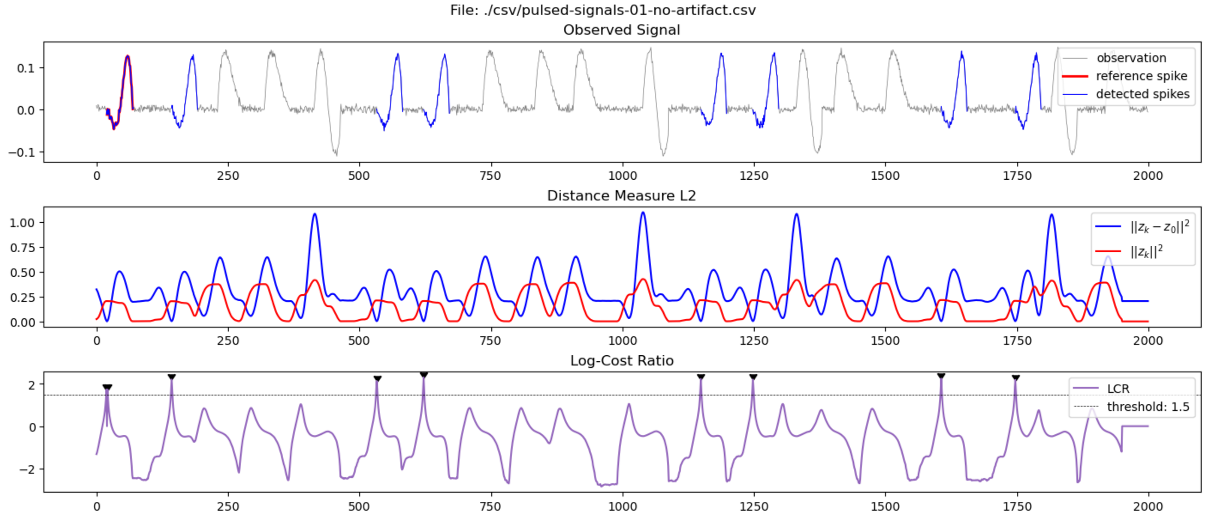
- We observe neural spikes as shown below, In your opinion, which shapes are similar? Explain! How do you define similarity?
    - If the waveforms are similar (shape, amplitude, duration)
- Which kind of distance measures do you know?
    - Euclidean Distance, Scale-Invariant, Cosine, Euclidean distance between centered (mean)
- What is the Euclidean distance between two 1D shapes of fixed length L?
    - Square root of sum of the squared differences z1, z2
    - Z1=[1,2,3], z2=[4,0,3]; [1-4, 2-0, 3-3]; [(-3)^2, 2^2, 0]; 9 + 4 + 0; sqrt(13) = 3.61
- What is the effect/consequence on the example below when applying these distance measures?
    - Spike that look similar to the reference spike have smaller Euclidean distances, and are flagged as similar.
- Wha is the key difference between using cosine distance versus cosine similarity? What do you need to know about a feature or 1D biomarker in order to correctly choose between the two methods?
    - Cosine Similarity: Measures similarity (angles of two vectors, magnitude irrelevant), -1 to +1, high value = similar, directional similarity
        - np.inner(z, z_ref) / (np.linalg.norm(z) * np.linalg.norm(z_ref))
    - Cosine Distance: Measures dissimilarity (based on the angle btwn. Them), 0 to 2, Low value = similar, directional dissimilarity
- Draw the shapes / spikes as vectors





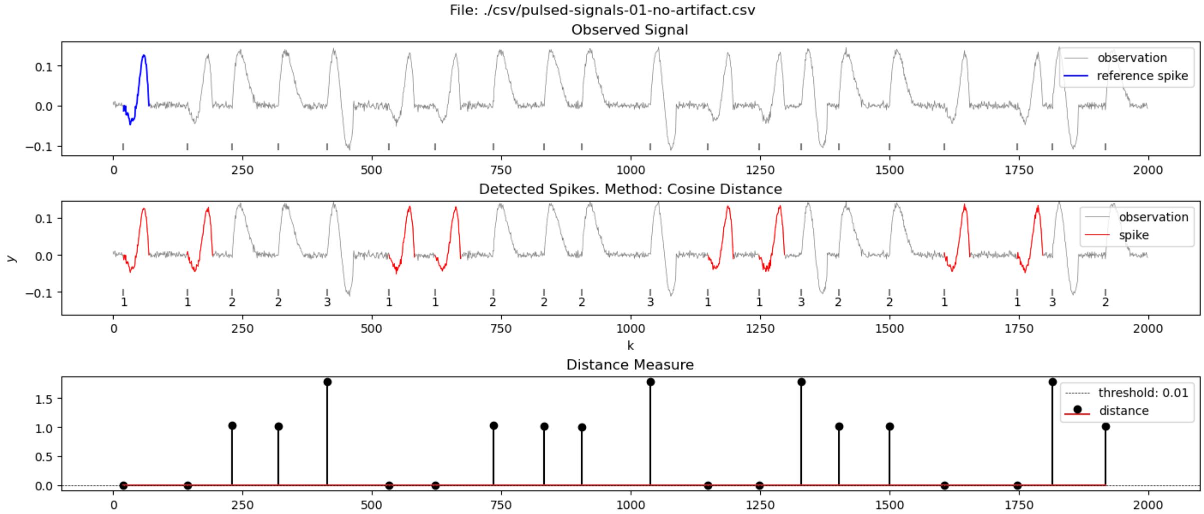
What do I see:

- Squared Distance: ||zk-z0||^2 – squared distance to the z0 (reference spike), the lower the distance, the higher the similarity.
- Energy: ||zk||^2 – amplitude squared (of the spike), spikes with similar energy often share similar amplitudes = similar shape
- LCR: normalizes distance by the energy -0.5 * log(distance / energy), high LCR = high similarity in Distance-Energy Ratio.
- LCR_TRESHOLD = 1.5



4.4.2 Sorting

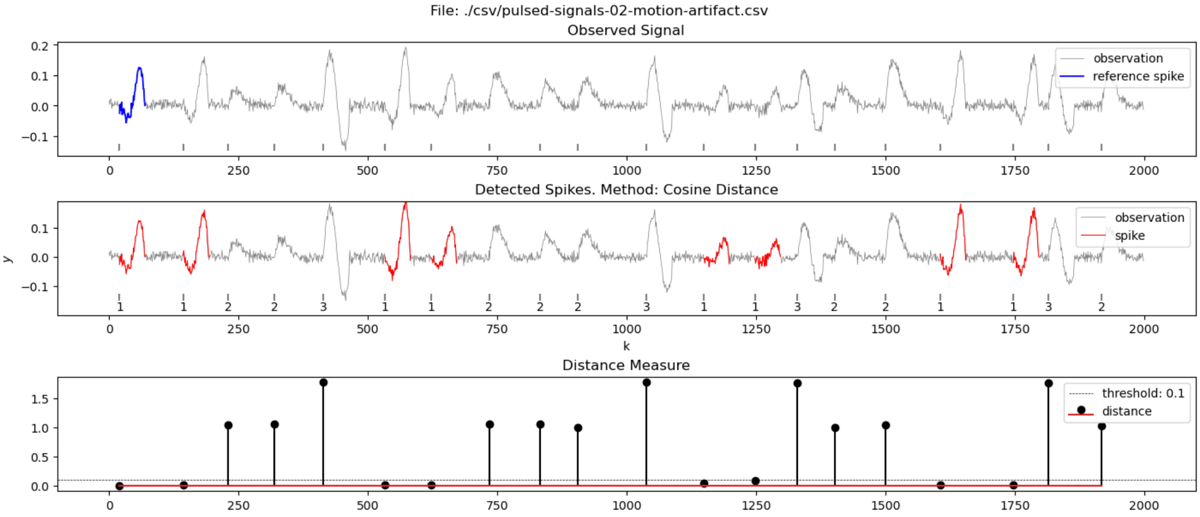
4.4.2.1 No-Artifact



Observations:

- All distance measures worked well, Euclidean (0.06), Scale Invariant (0.15), Cosine (0.01), mean centered (0.1)
- Cosine Distance is very effective in identifying spike similar to Label1, this is due to the signals have very similar shapes leading to angles close to 0° (1 – cosine similarity], amplitude only vary only slightly.

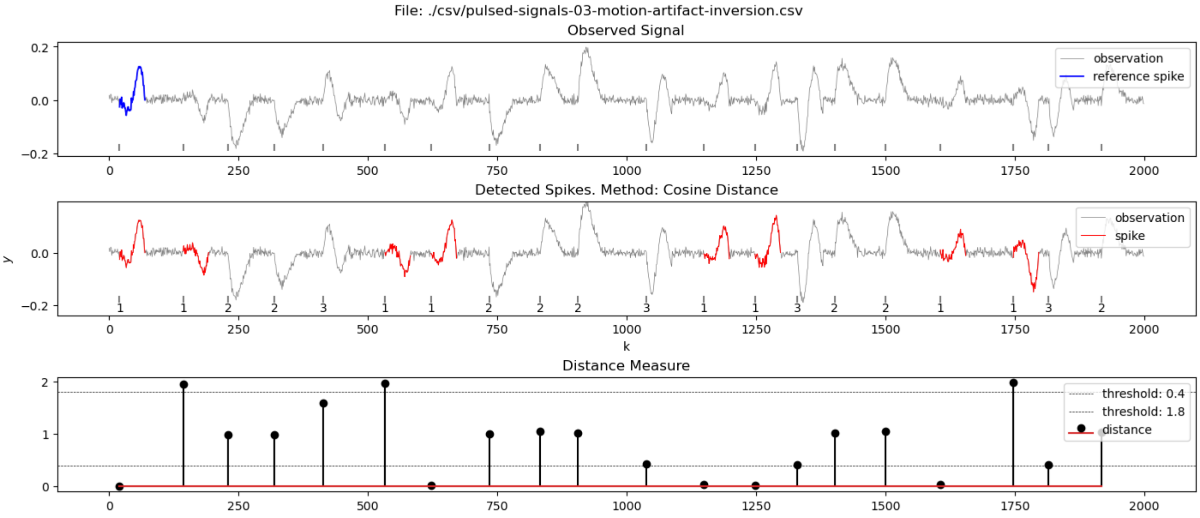
4.4.2.2 Motion Artifact



Observations:

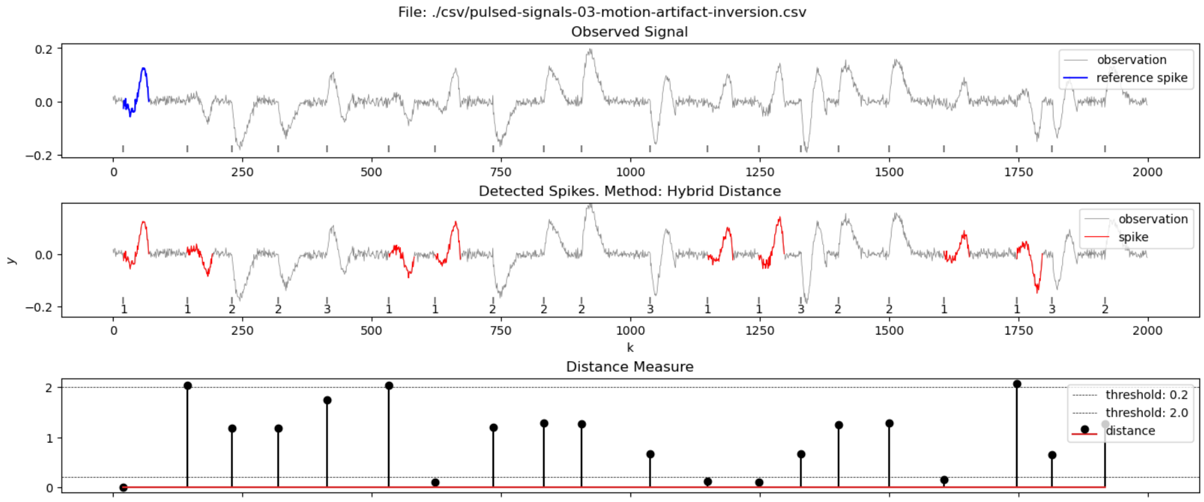
- Again, dissimilar spikes produce higher cosine distances, higher than the threshold of 0.1
- The shape remains almost the same (despite noise introduced), amplitude variations are removed due to distance normalization.

4.4.2.3 Motion Artifact - Inversion



Observations:

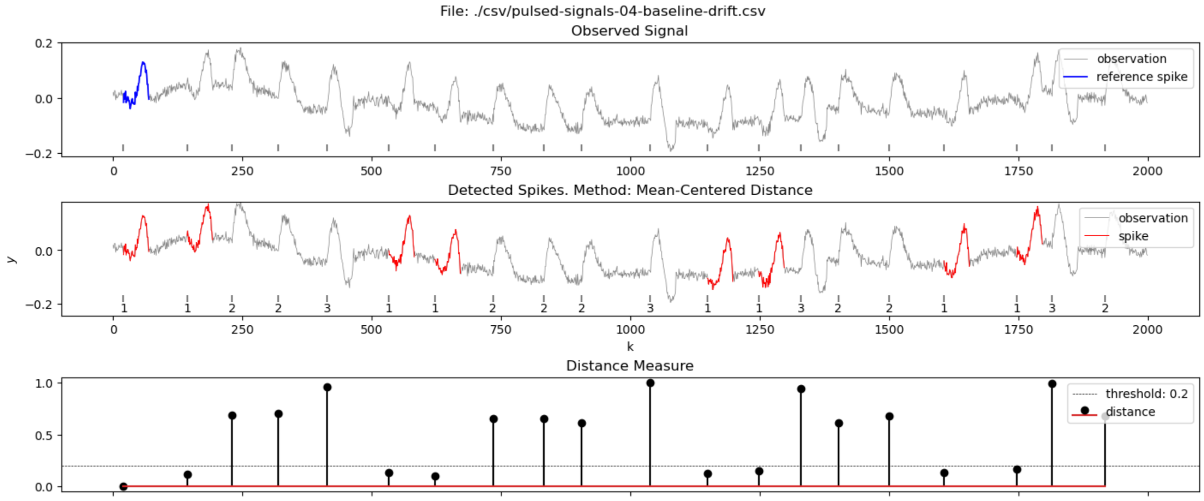
- Problem solved by introducing a upper & lower threshold. Inverted similaritiy, vectors inverted.



Observations:

- {'euclidean': 0.0, 'scale_invariant': 0.4, 'cosine': 0.6, 'mean_centered': 0.1}
- Trying to maximize the invertion, by adding scaling of the amplitudes.
    - Scale invariant: helps to detect amplitude variation
    - Cosine: Helps to match similar shape as the angle of vector is measured (also if inverted, TOP_THRESHOLD)

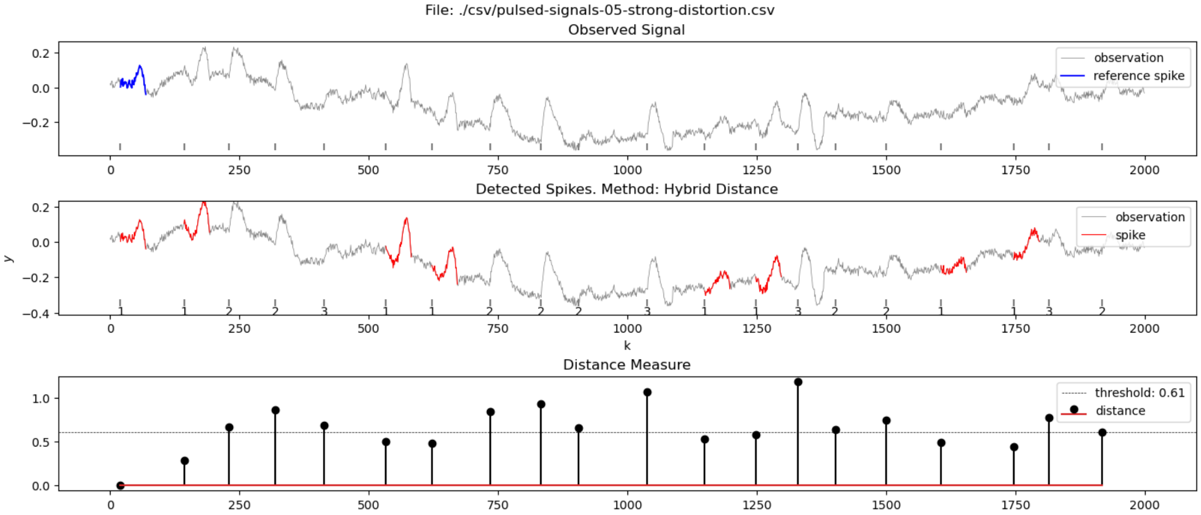
4.4.2.4 Baseline Drift



Observations:

- Cosine Distance had one false-positive, mean-centered performed better
- The baseline-drift changes overall direction of the vectors, by adding a constant offset to the signal.
- By substracting the mean (centering around 0), eliminates the drift.

4.4.2.5 Strong distortion



Observation:

- Grid-Search approach to find the best weights (Accuracy 100%):
  - {'euclidean': 0.0, 'scale_invariant': 0.1, 'cosine': 0.1, 'mean_centered': 0.8}
  - Mean-Centered Distance addresses the (baseline drift)
  - Captures angular alignment (noise)
  - Scale-Invariant: accounting spikes with different magnitudes (amplitudes)