# Zastosowanie technik rozmytych do oceny jakości sygnału PPG

Tomasz Hawro
Jan Słowik
Karolina Antonik

# Problem do rozwiązania

- Problem

  Sygnały PPG pochodzące z urządzenia Empatica E4 są wątpliwej jakości

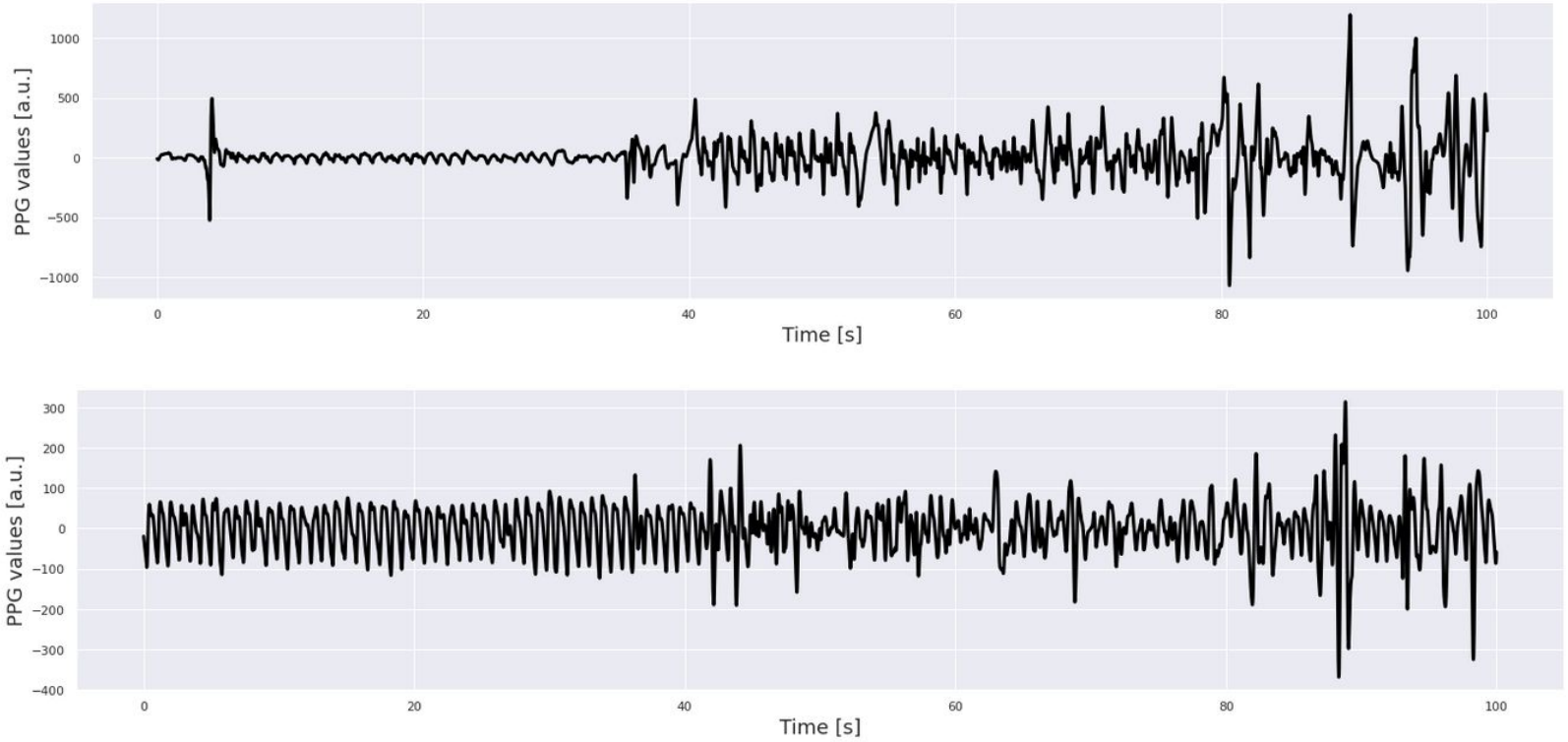
- Rozwiązanie

  **Automatyczne narzędzie oceniające jakość sygnału PPG**

- Zysk

  Odfiltrowane sygnały o dobrej jakości są odpowiednim inputem do modeli AI

# Przykładowe sygnały PPG



# Kroki do osiągnięcia celu

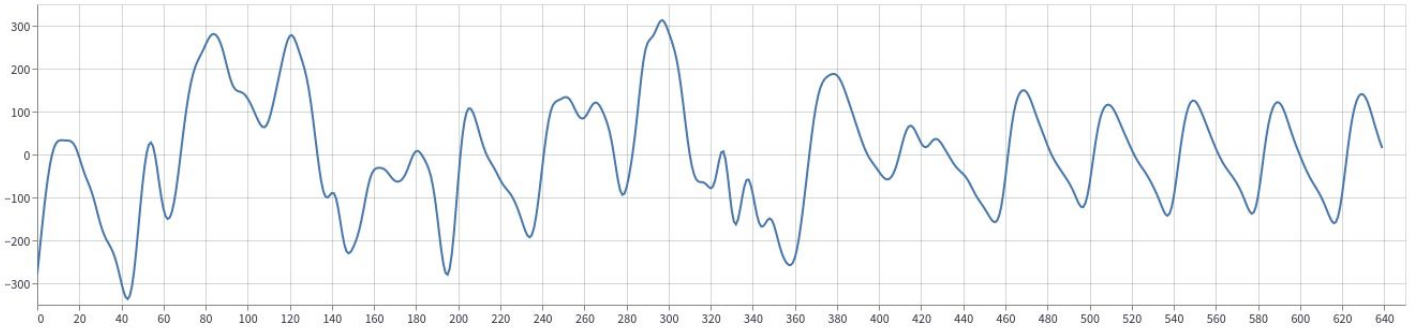
1. Stworzenie narzędzia do adnotacji danych
2. Adnotacja ok. tysiąca 10-sekundowych fragmentów sygnałów pod kątem jakości
3. Analiza sygnałów w celu znalezienia cech, które mogłyby świadczyć o jakości
4. Stworzenie zbioru danych składającego się z cech sygnałów (input) i etykiet jakości (target)
5. Wyznaczenie reguł służących do wyuczenia modelu **`Mamdani`**
6. Wyuczenie modelu **`Mamdani`**

# Narzędzie do adnotacji danych

## Fuzzy Annotator!

You are Tomek

### BVP signal (57 sample)




### Label

Good

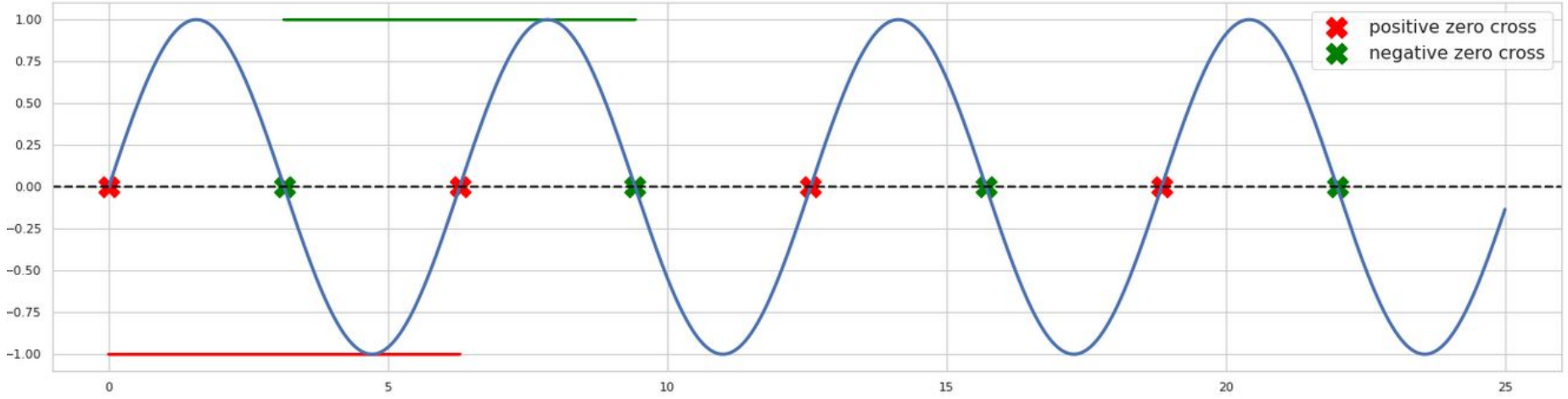
Medium

Poor

# Cechy świadczące o jakości

- **std** - odchylenie standardowe sygnału
- **skew** - skośność dystrybucji wartości sygnału
- **kurtosis** - kurtoza dystrybucji wartości sygnału
- **entropy** - entropia sygnału
- **pos_zc_int_mean** - średnia interwałów przecięć przez zero z minusa na plus
- **pos_zc_int_std** - odchylenie standardowe interwałów przecięć przez zero z minusa na plus
- **neg_zc_int_mean** - średnia interwałów przecięć przez zero z plusa na minus
- **neg_zc_int_std** - odchylenie standardowe interwałów przecięć przez zero z plusa na minus

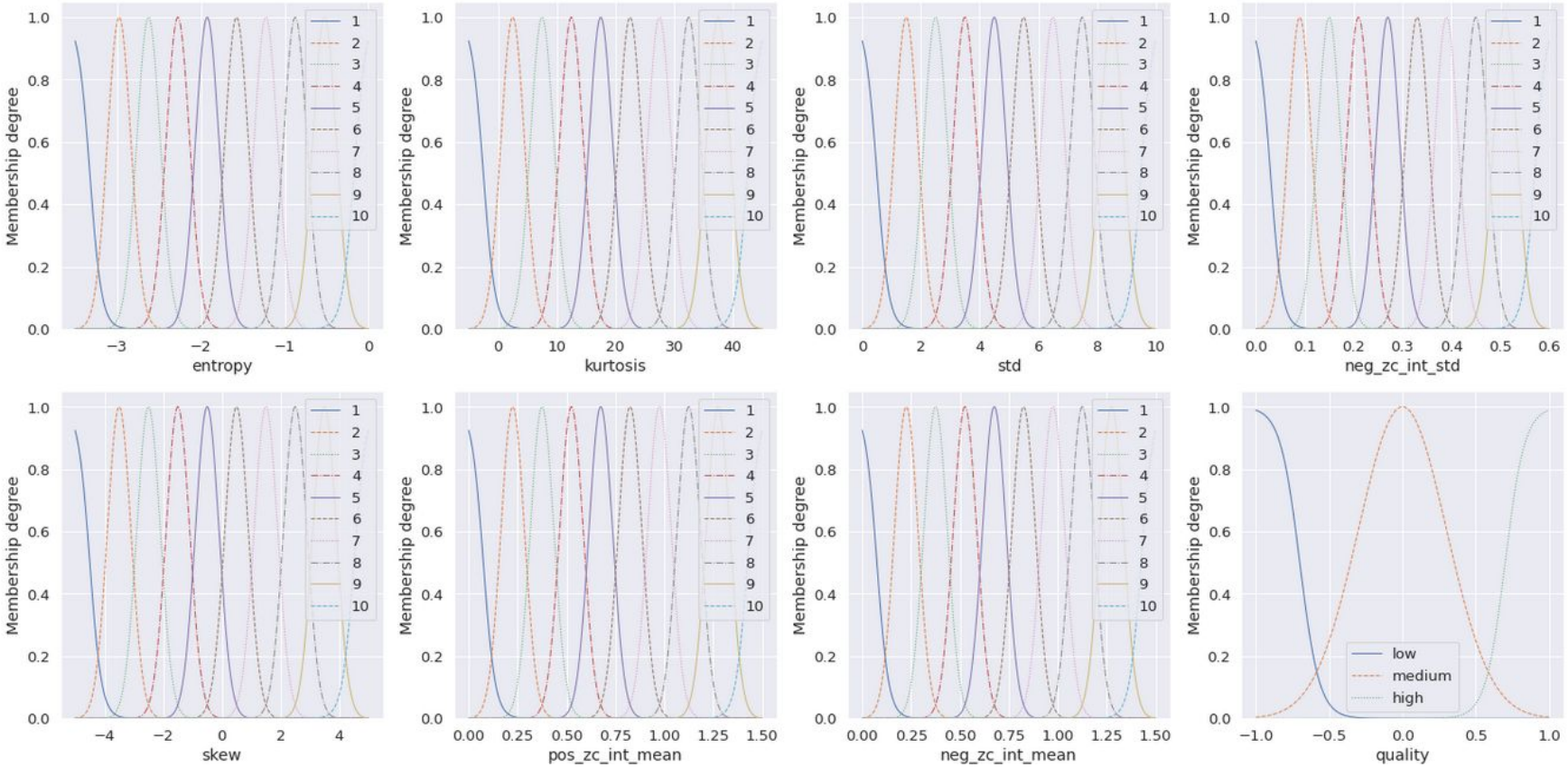
# Przecięcia przez zero



# Zbiór danych

| | std | skew | kurtosis | entropy | pos_zc_int_mean | pos_zc_int_std | neg_zc_int_mean | neg_zc_int_std | label |
|---|---|---|---|---|---|---|---|---|---|
| **count** | 1049.000000 | 1049.000000 | 1049.000000 | 1049.000000 | 1049.000000 | 1049.000000 | 1049.000000 | 1049.000000 | 1049.000000 |
| **mean** | 71.398569 | -0.005707 | 2.155045 | -0.966497 | 0.670913 | 0.213404 | 0.670496 | 0.216351 | -0.342231 |
| **std** | 80.771093 | 0.887452 | 5.050068 | 0.531913 | 0.259973 | 0.167205 | 0.259651 | 0.156003 | 0.865000 |
| **min** | 0.498119 | -6.278308 | -1.453442 | -3.427461 | 0.171845 | 0.007440 | 0.170886 | 0.010155 | -1.000000 |
| **25%** | 10.505289 | -0.421837 | -0.758180 | -1.205916 | 0.472673 | 0.092818 | 0.465444 | 0.100608 | -1.000000 |
| **50%** | 47.408679 | -0.108233 | 0.222546 | -0.785828 | 0.667757 | 0.200875 | 0.666049 | 0.198341 | -1.000000 |
| **75%** | 100.776425 | 0.247694 | 2.904381 | -0.557921 | 0.843835 | 0.295669 | 0.838565 | 0.305581 | 1.000000 |
| **max** | 754.943358 | 4.913352 | 42.376044 | -0.325507 | 2.190898 | 2.684132 | 2.170234 | 1.801960 | 1.000000 |

# Stworzenie rozmytego zbioru



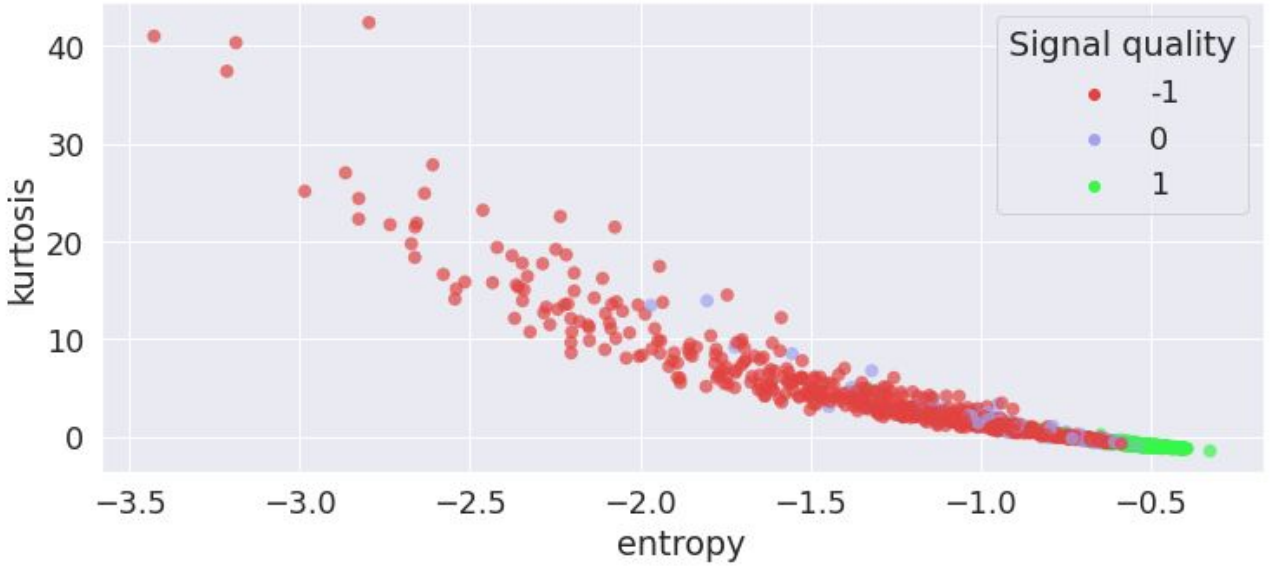
# Reguły

1. IF (entropy IS 10) AND (kurtosis IS 1) THEN (quality IS high)
2. IF (kurtosis IS 5) OR (kurtosis IS 6) OR (kurtosis IS 7) OR (kurtosis IS 8) OR (kurtosis IS 8) OR (kurtosis IS 10) THEN (quality IS low)
3. IF (skew IS 3) AND (NOT (pos_zc_int_mean IS 1)) THEN (quality IS high)
4. IF (entropy IS 7) AND (kurtosis IS 1) THEN (quality IS medium)
5. IF (entropy IS 8) AND (neg_zc_int_std IS 1) THEN (quality IS high)
6. IF (entropy IS 7) AND (neg_zc_int_std IS 2) THEN (quality IS medium)
7. IF (pos_zc_int_mean IS 1) AND (neg_zc_int_mean IS 1) THEN (quality IS low)
8. IF (pos_zc_int_mean IS 1) AND (neg_zc_int_mean IS 1) THEN (quality IS low)
9. IF (pos_zc_int_mean IS 2) AND (neg_zc_int_mean IS 2) THEN (quality IS low)
10. IF (pos_zc_int_mean IS 3) AND (neg_zc_int_mean IS 3) THEN (quality IS low)
11. IF (pos_zc_int_mean IS 8) AND (neg_zc_int_mean IS 8) THEN (quality IS high)

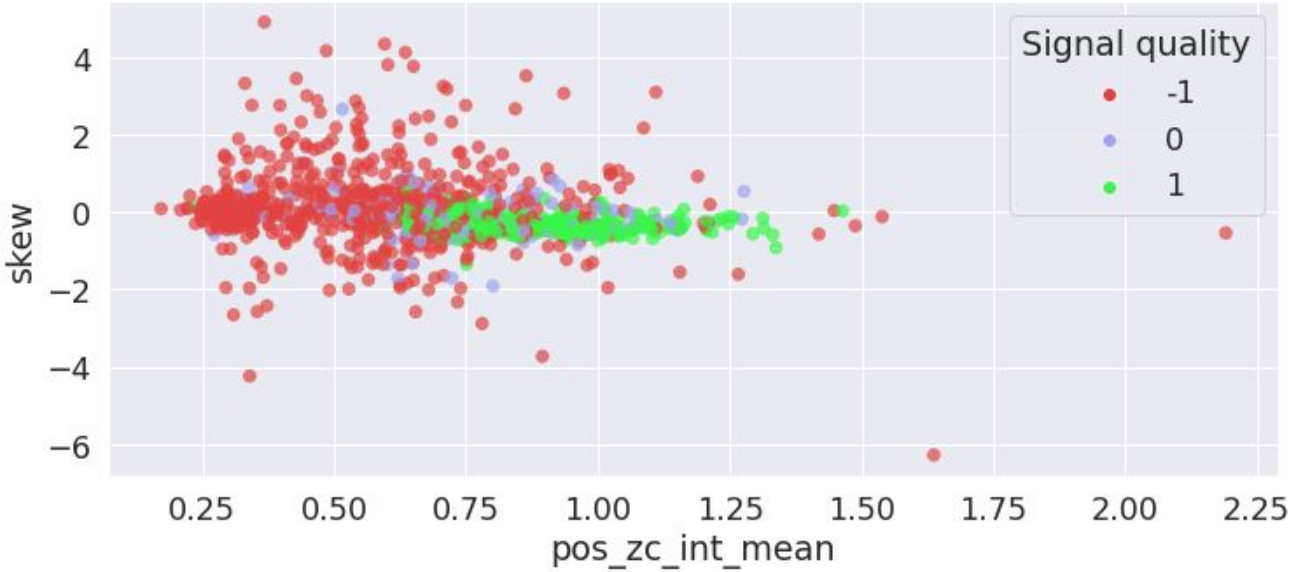
IF (**entropy IS 10**) AND (**kurtosis IS 1**) THEN (**quality IS high**)

IF (**kurtosis IS 5**) OR (**kurtosis IS 6**) OR (**kurtosis IS 7**) OR (**kurtosis IS 8**) OR (**kurtosis IS 9**) OR (**kurtosis IS 10**) THEN (**quality IS low**)

IF (**entropy IS 7**) AND (**kurtosis IS 1**) THEN (**quality IS medium**)



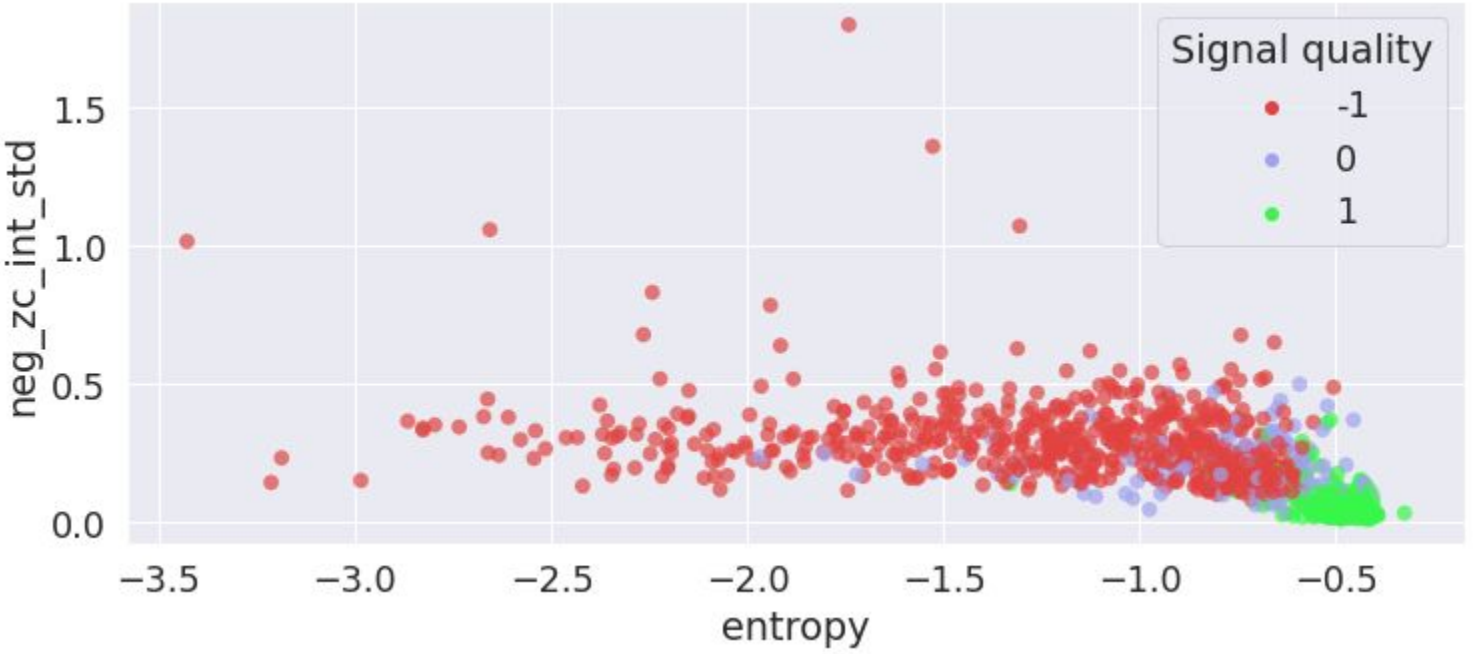
IF (**skew IS 3**) AND (**NOT (pos_zc_int_mean IS 1)**) THEN (**quality IS high**)



IF (**entropy IS 8**) AND (**neg_zc_int_std IS 2**) THEN (**quality IS medium**)

IF (**entropy IS 9**) AND (**neg_zc_int_std IS 1**) THEN (**quality IS high**)

IF (**entropy IS 10**) AND (**neg_zc_int_std IS 1**) THEN (**quality IS high**)

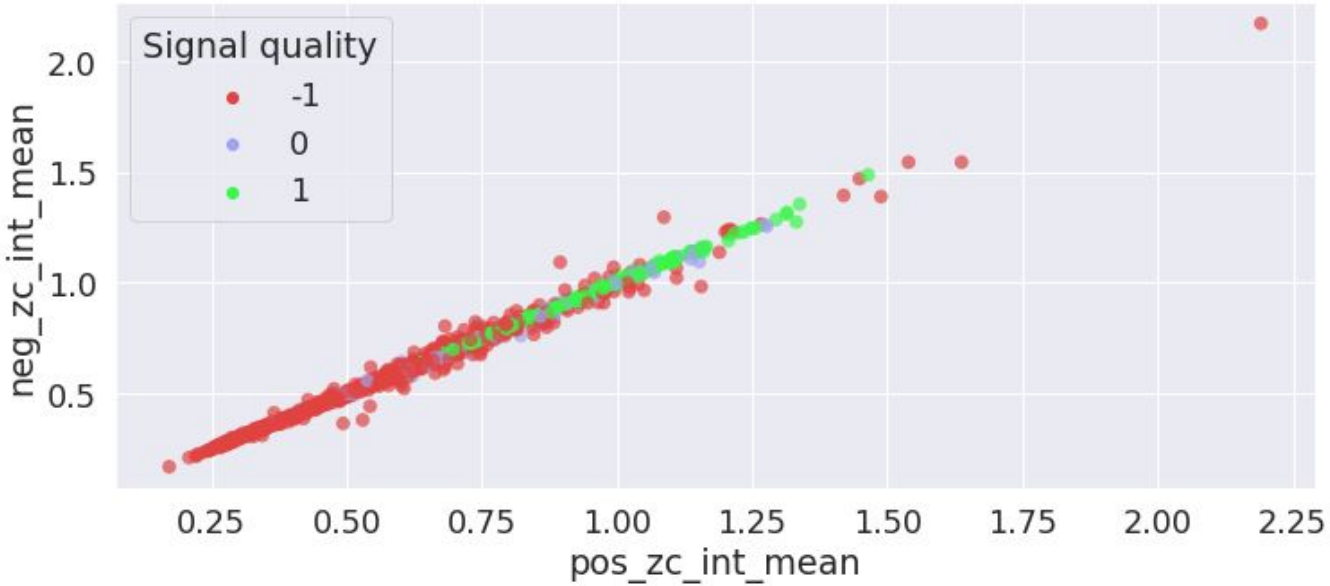


IF (**pos_zc_int_mean IS 1**) AND (**neg_zc_int_mean IS 1**) THEN (**quality IS low**)

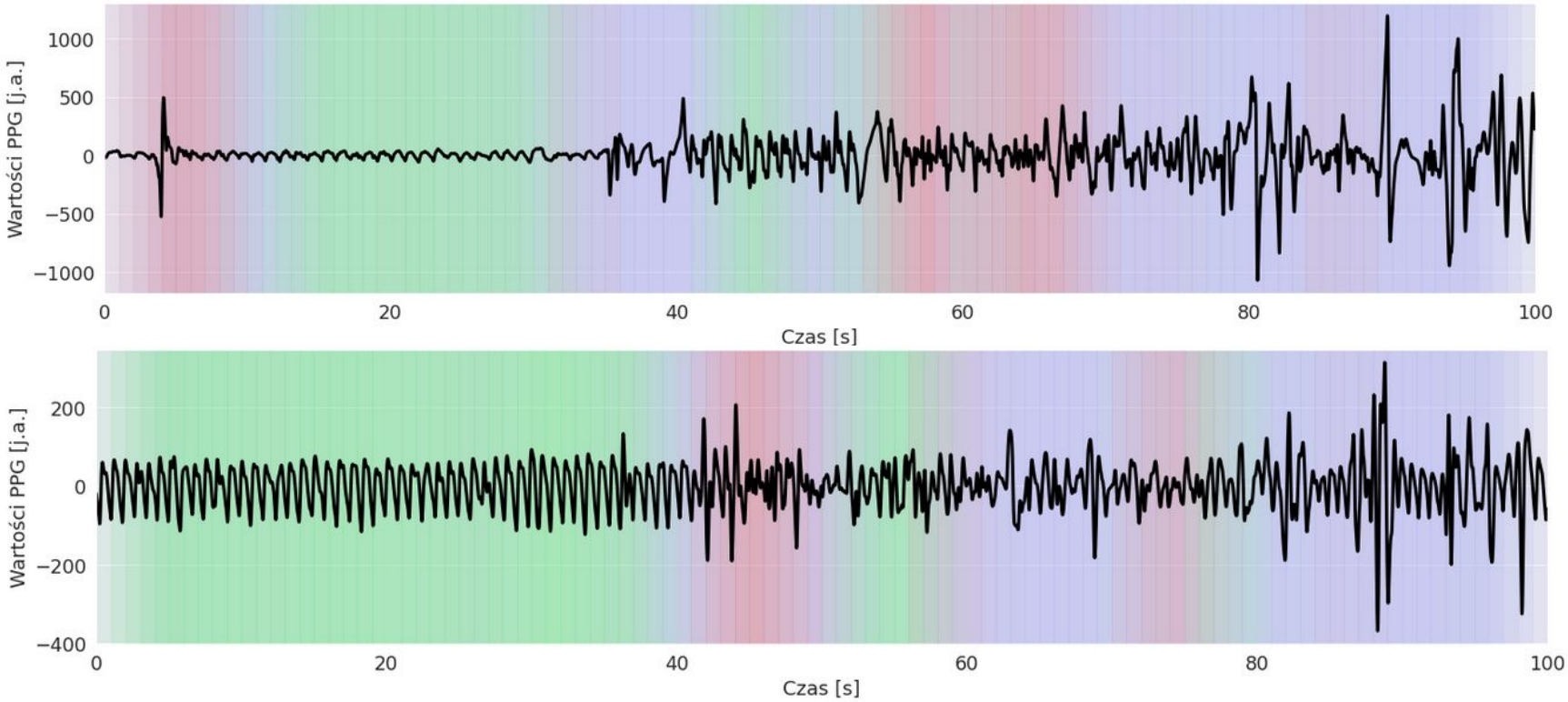
IF (**pos_zc_int_mean IS 2**) AND (**neg_zc_int_mean IS 2**) THEN (**quality IS low**)

IF (**pos_zc_int_mean IS 3**) AND (**neg_zc_int_mean IS 3**) THEN (**quality IS low**)

IF (**pos_zc_int_mean IS 8**) AND (**neg_zc_int_mean IS 8**) THEN (**quality IS high**)



# Wyniki


Wartości PPG [j.a.]
1000
500
0
−500
−1000
0
20
40
60
80
100
Czas [s]
Wartości PPG [j.a.]
200
0
−200
−400
0
20
40
60
80
100
Czas [s]

# Zalety stosowania technik rozmytych

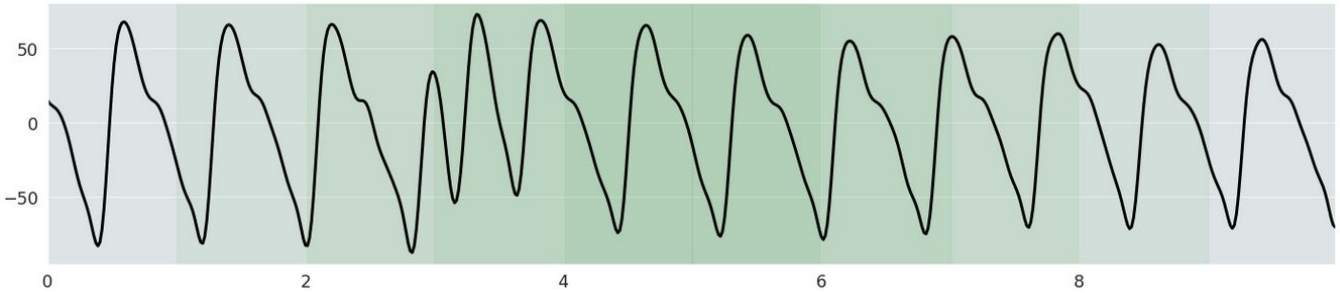
- Proste podejście oparte o reguły
- Otrzymana jakość sygnału jest ciągła z przedziału `[-1, 1]`. Klasyczne metody uczenia maszynowego (np. drzewo decyzyjne) zwracają jakość ze zbioru `{-1, 0, 1}`

# Wady stosowania technik rozmytych

- Dużo pracy przy wyznaczaniu odpowiednich reguł
- Dłuższy czas inferencji
  - RandomForest: `150 ms ± 4.45 ms`
  - Fuzzy: `2.92 s ± 98.3 ms`

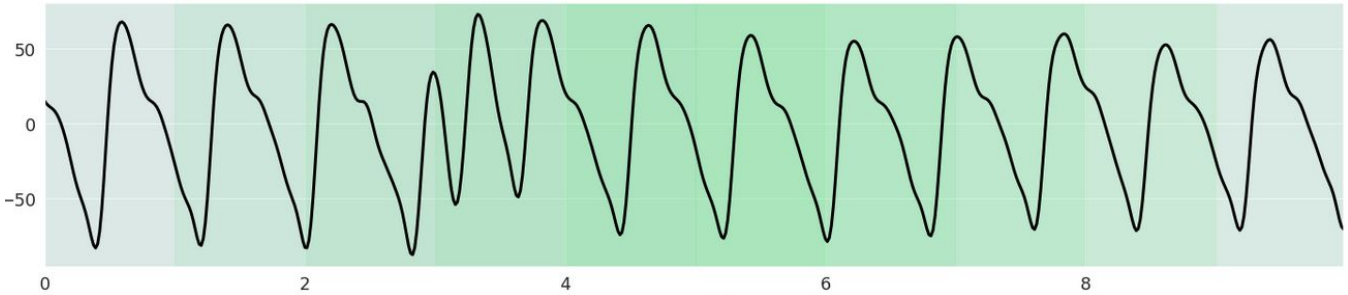
# Inferencja RandomForest vs Mamdani

## RandomForest




| start | end | label |
|---|---|---|
| 0 | 5 | 1.0 |
| 1 | 6 | 1.0 |
| 2 | 7 | 1.0 |
| 3 | 8 | 1.0 |
| 4 | 9 | 1.0 |
| 5 | 9 | 1.0 |

## Mamdani

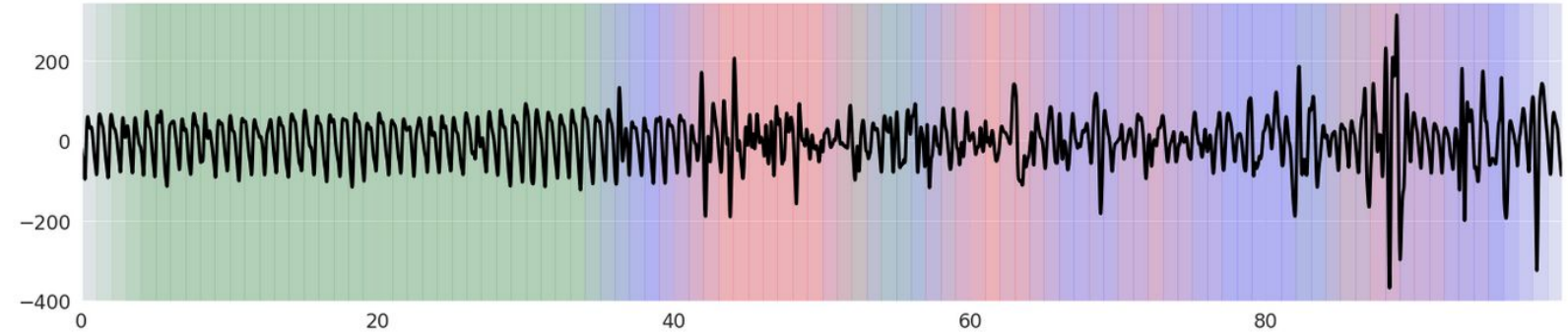



| start | end | label |
|---|---|---|
| 0 | 5 | 0.658495 |
| 1 | 6 | 0.682337 |
| 2 | 7 | 0.674330 |
| 3 | 8 | 0.676506 |
| 4 | 9 | 0.816360 |
| 5 | 9 | 0.816332 |

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | ... | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **start** | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | ... | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 |
| **end** | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | ... | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 99 |
| **label** | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | ... | 0 | -1 | 0 | 0 | -1 | 0 | 0 | 0 | 0 | 0 |

3 rows × 96 columns

## RandomForest




| | 0.0 | 1.00 | 2.00 | 3.00 | 4.00 | 5.00 | 6.00 | 7.00 | 8.00 | 9.0 | ... | 86.0 | 87.0 | 88.0 | 89.0 | 90.0 | 91.0 | 92.0 | 93.00 | 94.0 | 95.0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **start** | 0.0 | 1.00 | 2.00 | 3.00 | 4.00 | 5.00 | 6.00 | 7.00 | 8.00 | 9.0 | ... | 86.0 | 87.0 | 88.0 | 89.0 | 90.0 | 91.0 | 92.0 | 93.00 | 94.0 | 95.0 |
| **end** | 5.0 | 6.00 | 7.00 | 8.00 | 9.00 | 10.00 | 11.00 | 12.00 | 13.00 | 14.0 | ... | 91.0 | 92.0 | 93.0 | 94.0 | 95.0 | 96.0 | 97.0 | 98.00 | 99.0 | 99.0 |
| **label** | 0.7 | 0.68 | 0.66 | 0.67 | 0.74 | 0.74 | 0.77 | 0.78 | 0.69 | 0.7 | ... | -0.0 | 0.0 | -0.0 | 0.0 | -0.0 | -0.0 | -0.0 | 0.08 | 0.0 | 0.0 |

3 rows × 96 columns

## Mamdani