

# Projekt UMA – Dokumentacja Końcowa

Kamil Kośnik, Kacper Radzikowski

Repozytorium projektu: <https://github.com/FRSH-0109/ML-DT-DNA>

## Opis oryginalny dostępny na stronie “zapisy”:

Drzewo decyzyjne w zadaniu klasyfikacji miejsc rozcięcia w sekwencji DNA. Należy dopuścić alternatywę w testach, np. `if( atr12=='A' || atr12=='T' )`. Więcej informacji o specyfice problemu znaleźć można w opisie. Dane do pobrania: donory, akceptory. Przed rozpoczęciem realizacji projektu proszę zapoznać się z zawartością strony.

## Opis oryginalny dostępny na stronie prowadzącego ([link](#)):

Istnieją dwa rodzaje miejsc rozcięcia sekwencji kodującej białko: donory i akceptory. Ich odnalezienie otwiera drogę do znalezienia eksonów, czyli sekwencji kodujących białka.

Należy zaimplementować klasyfikator, następnie przeprowadzić jego trening i testowanie na 2 problemach:

1. szukanie donorów,
2. szukanie akceptorów

Każdy ze zbiorów danych należy rozdzielić na trenujący i testujący lub zastosować walidację krzyżową. Zaimplementowany klasyfikator należy przebadac (wykonać eksperymenty).

Jeżeli chodzi o dane to [w tym pliku](#) znajdują się przykłady donorów, a [w tym pliku](#) przykłady akceptorów. W pierwszej linii każdego z nich napisano, na której pozycji (licząc litery od lewej strony) we fragmentach sekwencji jest granica pomiędzy intronem a eksonem. Dana ta jest zbędna dla klasyfikatora - może jednak pomóc badaczowi w interpretacji wyników. Dalej w pliku występują parami: linia określająca czy jest to przykład pozytywny (1) czy negatywny (0) oraz sam przykład, czyli sekwencja DNA. Przykłady negatywne to takie, które częściowo wyglądają jak miejsca rozcięcia, ale nimi nie są.

## Skrócona interpretacja zadania

Celem zadania jest implementacja algorytmu drzewa decyzyjnego w taki sposób, aby finalny rezultat umożliwiał rozpoznawanie miejsc rozcięcia sekwencji kodującej białko. Model drzewa decyzyjnego będzie realizował dwa zadania klasyfikacji – szukanie donorów i akceptorów. Dodatkowo w ramach zadania przeprowadzone zostaną eksperymenty sprawdzające możliwości i właściwości modelu w zależności od jego parametrów. Na bazie wyników tych eksperymentów dokonane zostanie porównanie oraz wyciągnięcie wniosków na temat badanego algorytmu uczenia maszynowego.

## Skrócony opis algorytmu

Algorytm drzewa decyzyjnego realizowany jest w następujący sposób:

1. Inicjujemy algorytm pierwszym węzłem drzewa decyzyjnego nazywanego korzeniem drzewa (z ang. root) zawierającego cały zbiór danych trenujących **T**
2. Znajdujemy dla zbioru danych w węźle atrybut i próg (przy atrybutach, które są opisywane wartościami ciągłymi) wartości najlepiej maksymalizujący przyrost informacji
3. Dzielimy zbiór **T** na podzbiory dzieląc je na bazie wyznaczonej w kroku 2 pary wartość-próg (w przypadku, gdy atrybut ma wartości ciągłe)
4. Generujemy nowy węzeł decyzyjny zawierający wybrany najlepszy atrybut
5. Tworzymy rekursywnie nowe drzewa decyzyjne wykorzystując podzbiory danych, które stworzyliśmy w kroku 3 do momentu uzyskania jednego z kryteriów stopu: zbiór danych jest czysty – zawiera tylko przykłady jednej klasy, którą przypisujemy jako wartość ostatniego węzła tzw. Liścia drzewa; osiągniemy maksymalną założoną przez nas głębokość drzewa lub zbiór danych w węźle jest mniejszy niż ustalona przez nas wartość graniczna wielkości zbioru do podziału. W dwóch ostatnich przypadkach w zadaniu klasyfikacji, które będziemy realizować za wartość liścia przyjmujemy klasę, która reprezentuje najwięcej przykładów w zbiorze danych

W ramach Atrybutu Miary Wyboru możemy wykorzystać Entropię lub Współczynnik Gini'ego.

## Opis danych - przypomnienie

Dwa niezależne zbiory danych, które w formie pliku .dat znajdują się [tutaj](#) i [tutaj](#), opisują kolejno przykłady donorów i akceptorów. Każdą sekwencję DNA poprzedza jej klasa, pozytywna (1) lub negatywna (0), w oparciu o którą przebiegać będzie klasyfikacja. Klasa ta oznacza obecność rozcięcia sekwencji kodującej DNA w danym przykładzie. Dane należące do zbiorów uczącego i weryfikującego zostaną dobrane losowo, lecz ze zwróceniem uwagi na odpowiedni stosunek ilości przykładów pozytywnych i negatywnych w danej grupie. Ich rozmiar będzie jednym z parametrów modyfikacji podlegających ocenie.

## Miary jakości - przypomnienie

Planowane jest określenie jakości rezultatu przy pomocy:

- Tablicy pomyłek (ang. Confusion matrix)
- Błędu klasyfikacji na zbiorze trenującym
- Dokładności jako procent poprawnych predykcji w stosunku do całkowitej liczby próbek
- Precyzji jako stosunek liczby prawdziwie pozytywnych predykcji do sumy prawdziwie pozytywnych i fałszywie pozytywnych predykcji
- Czułości jako stosunek liczby prawdziwie pozytywnych predykcji do sumy prawdziwie pozytywnych i fałszywie negatywnych predykcji
- F1-score, czyli średniej harmonicznej precyzji i czułości. Pozwoli nam na ocenę, w której chcemy zwrócić uwagę na istotę równowagi między precyzją a czułością.

## Plan eksperymentów

- Implementacja własnego algorytmu oraz badanie jego skuteczności dla:
  - Różnych proporcji podziału zbioru trenującego i weryfikującego
  - Różnych kryteriów stopu:
    - Osiągnięcie czystego zbioru przykładu
    - Głębokości drzewa decyzyjnego
    - Zbyt mała ilość przykładów w konkretnym węźle
  - Różnych współczynników miar atrybutów pod względem zapewnienia najlepszej skuteczności modelu:
    - Miara Entropii
    - Współczynnik Gini'ego
  - Różnej wielkości zbioru uczącego i weryfikującego
- Porównanie gotowych algorytmów z własną implementacją, wykorzystując dostępne modyfikacje i parametryzacje

## Eksperymenty

Warto na wstępie zauważyć, że nasz algorytm ma za zadanie realizację dwa zadania klasyfikacji – szukania donorów i akceptorów. Oba te zbiory danych znacząco różnią się ilością dostępnych atrybutów (sekwencja w zestawie do klasyfikacji donorów ma 15 atrybutów, podczas, gdy dla akceptorów jest to już, aż 90 atrybutów). Ta różnica w charakterystyce skłania do postawienia hipotezy na wejściu do eksperymentów, że algorytm będzie zachowywał się inaczej dla takich samych parametrów w zależności od zadania klasyfikacji, które realizuje. Z tego też powodu przeprowadzimy oddzielne badania dla obu zestawów danych.

### Eksperymenty dla zbioru donorów

#### Badanie wpływu zbioru testującego i weryfikującego na skuteczność klasyfikacji:

Badania są przeprowadzone na próbie 50 zbiorów trenujących i weryfikujących uzyskanych przez losowanie dostarczonych danych w badanej w danym eksperymencie proporcji z zachowaniem stosunku przykładów pozytywnych i negatywnych niepodzielonego zbioru. Na cele eksperymentu przyjmujemy następujące parametry: **współczynnik miary atrybutu** – entropia, **minimalna liczba przykładów - 3**, **maksymalna głębokość** - 100. W celach weryfikacji eksperymenty przeprowadzamy również na implementacji drzewa decyzyjnego zaczerpniętego z biblioteki scikit-learn.

#### Wyniki eksperymentów:

Współczynnik danych uczących do wszystkich danych	Dokładność klasyfikacji dla zbioru testowego [%]		Dokładność klasyfikacji dla zbioru weryfikującego [%]	
	nasza implementacja	Biblioteka scikit-learn	nasza implementacja	Biblioteka scikit-learn
<b>0.4</b>	99,30	99,59	92,34	91,79
<b>0.5</b>	99,28	99,55	92,52	91,86
<b>0.6</b>	99,26	99,56	92,54	92,11
<b>0.7</b>	99,29	99,52	92,67	92,23
<b>0.8</b>	99,28	99,52	92,76	92,22

<b>0.9</b>	<b>99,30</b>	<b>99,47</b>	<b>92,99</b>	<b>92,21</b>
<b>0.95</b>	99,30	99,49	93,26	93,26
<b>0,98</b>	99,28	99,52	93,51	93,30
<b>0,99</b>	99,28	99,54	92,72	92,15

**Tabela 1. Dokładność klasyfikacji na zbiorze testującym i weryfikującym dla naszej i dostarczonej w bibliotece scikit-learn implementacji algorytmu – zestaw przykładów donorów**

Analiza wyników eksperymentów pokazuje nam, że proporcja w jakiej dane zostaną podzielone na zbiór trenujący i weryfikujący nie ma bardzo dużego wpływu na dokładność klasyfikacji. Jak widzimy na przestrzeni przeprowadzonych badań różnica między najgorszą (dla zbioru trenującego składającego się z 0,4 wszystkich danych), a najlepszą dokładnością (dla zbioru trenującego składającego się z 0,98 wszystkich danych) dla zbioru weryfikującego wynosi 1,17 pkt. procentowego co jest zauważalną, ale nie znaczącą różnicą. Ponadto warto zauważyć, że dla zbioru trenującego dokładność naszej implementacji waha się w przedziale 0,02 pkt. Procentowego co jest praktycznie pomijalną różnicą. Z racji na chęć uniknięcia sytuacji, w której klasyfikator może ulec nadmiernemu dopasowaniu będziemy dla dalszych eksperymentów używać zbioru trenującego stanowiącego 0,9 całego zbioru danych (lepiej sprawujące się proporcje sprawiają, że zbiór weryfikujący sięga rozmiarów kilkudziesięciu przykładów co może nie być wystarczającą próbką by uczciwie ocenić dokładność algorytmu i nie ulegnięcie nadmiernemu dopasowaniu). Warto w tym miejscu nadmienić, że dodatkowo porównaliśmy wyniki działania naszego algorytmu drzewa decyzyjnego z implementacją z biblioteki scikit-learn i możemy stwierdzić, że nasza implementacja sprawuje się dla każdego badanego stosunku danych trenujących do całości danych lepiej od implementacji bibliotecznej. Jest to bardzo wiarygodny wyznacznik, że zrealizowany przez nas algorytm poprawnie implementuje algorytm drzewa decyzyjnego.

## **Badanie wpływu kryteriów stopu na skuteczność algorytmu:**

Badania są przeprowadzone na próbie 50 zbiorów trenujących i weryfikujących uzyskanych przez losowanie dostarczonych danych w stosunku 0.9 (który w toku poprzednich eksperymentów uznaliśmy za najlepszy) z zachowaniem stosunku przykładów pozytywnych i negatywnych niepodzielonego zbioru.

### **Badanie 1 – maksymalna głębokość drzewa**

W tym eksperymencie przyjmiemy za **minimalną ilość przykładów w węźle** wartość 3, a za **współczynnik miary atrybutu entropie**.

**Wyniki eksperymentów:**

<b>Maksymalna głębokość drzewa</b>	<b>Dokładność klasyfikacji [%]</b>	<b>Precyzja klasyfikacji [%]</b>	<b>Czułość klasyfikacji [%]</b>	<b>F1-score [%]</b>
5	<b>94,78</b>	<b>87,50</b>	<b>88,29</b>	<b>87,89</b>
10	93,68	83,62	87,39	85,46
20	93,10	83,19	84,68	83,93
40	92,98	83,19	84,68	83,93
60	93,20	83,93	84,68	84,30
80	92,81	83,78	83,78	83,78S

100	92,83	83,04	83,78	83,41
150	93,25	84,68	84,68	84,68
200	92,91	83,78	83,78	83,78
400	93,19	83,33	85,59	84,44

**Tabela 2. Wyniki badań maksymalnej głębokości drzewa na zbiorze weryfikującym dla naszej implementacji algorytmu – zestaw przykładów donorów**

Maksymalna głębokość drzewa	Dokładność klasyfikacji [%]	Precyzja klasyfikacji [%]	Czułość klasyfikacji [%]	F1-score [%]
5	92,11	81,82	81,08	81,45
10	<b>93,40</b>	<b>84,82</b>	<b>85,59</b>	<b>85,20</b>
20	92,21	82,57	81,09	81,82
40	92,41	82,72	81,98	82,35
60	92,10	81,25	81,61	81,61
80	92,06	82,41	80,18	81,28
100	92,05	82,57	80,36	81,45
150	92,30	82,57	81,08	81,82
200	91,92	81,65	80,18	80,91
400	92,46	82,88	82,14	82,51

**Tabela 3. Wyniki badań maksymalnej głębokości drzewa na zbiorze weryfikującym dla implementacji algorytmu z biblioteki scikit-learn - zestaw przykładów donorów**

Z wyniku badań maksymalnej głębokości drzewa dla zadania klasyfikacji donorów możemy wyciągnąć dwa wnioski. Po pierwsze nasza implementacja radzi sobie odrobine lepiej niż biblioteka scikit-learn z zadaniem klasyfikacji niezależnie od porównywanej wartości parametru (jedynym wyjątkiem jest pomiar precyzji klasyfikacji w przypadku maksymalnej głębokości drzewa 10, jednak jest to pojedynczy wypadek w skali wszystkich badanych wartości parametru i mierzonych miar jakości). Naszą hipotezą na temat przyczyny tego stanu rzeczy jest to zasługa tego, że nasza implementacja akceptuje alternatywę atrybutów jako kryterium podziału. Jednak naszą hipotezę z racji na nieznaną dokładną implementację algorytmu drzewa decyzyjnego realizowanego przez scikit-learn. Drugim wnioskiem jest stwierdzenie, że wartością parametru, która zapewnia najlepszą jakość klasyfikacji jest maksymalna głębokość drzewa o wartości 5. Przyczyną tego jest fakt, że przykłady donorów składają się zaledwie z 15 atrybutów. Co za tym idzie głębokości drzewa większe niż 15 tracą na sensowności i narażają nas na nadmierne dopasowanie drzewa.

## Badanie 2 – minimalna ilość przykładów w węźle

W tym eksperymencie przyjmijmy za **maksymalną głębokość drzewa** wartość 5 którą uznaliśmy w poprzednim eksperymencie za najlepszą wartość parametru, jako **współczynnik miary atrybutu** przyjmijmy ponownie entropię.

### Wyniki eksperymentów:

Minimalna ilość przykładów	Dokładność klasyfikacji [%]	Precyzja klasyfikacji [%]	Czułość klasyfikacji [%]	F1-score [%]
40	94,88	88,29	88,29	88,29
20	94,88	87,61	89,19	88,39
10	<b>95,03</b>	<b>88,39</b>	<b>89,19</b>	<b>88,79</b>
5	94,80	87,61	89,19	88,39
4	95,03	88,39	89,19	88,79
3	94,83	87,61	89,19	88,39
2	94,69	87,50	88,29	87,89

Tabela 6. Wyniki badań minimalnej ilości przykładów w węźle na zbiorze weryfikującym dla naszej implementacji algorytmu – zestaw przykładów donorów

Minimalna ilość przykładów	Dokładność klasyfikacji [%]	Precyzja klasyfikacji [%]	Czułość klasyfikacji [%]	F1-score [%]
40	92,33	81,98	81,98	81,98
20	92,08	81,82	81,08	81,45
10	<b>92,36</b>	<b>82,72</b>	<b>81,25</b>	<b>81,98</b>
5	92,00	81,82	81,08	81,45
4	92,27	82,57	81,09	81,82
3	92,29	81,98	81,98	81,98
2	92,01	81,82	81,08	81,45

Tabela 7. Wyniki badań minimalnej ilości przykładów w węźle na zbiorze weryfikującym dla implementacji algorytmu z biblioteki scikit-learn – zestaw przykładów donorów

Wyniki eksperymentów pokazują nam podobnie jak w poprzednim eksperymencie, że nasza implementacja sprawuje się lepiej od implementacji dostarczonej w bibliotece scikit-learn. Hipotezę tłumaczącą ten fakt przedstawiliśmy we wnioskach do poprzedniego eksperymentu stąd powstrzymamy się od ich powtórzenia. Pochylimy się bardziej nad doбором najlepiej radzącej sobie wartości parametru. To z racji na to, że w naszej implementacji miary jakości dla wartości 10 i 4 są dokładnie identycznie. Decyzje musimy więc podjąć na bazie intuicji popartej rozumieniem funkcjonowania algorytmu. Na tej bazie wybieramy minimalną ilość próbek do podziału w algorytmie na 10 z racji na świadomość, że im niższa wartość parametru tym większe ryzyko wygenerowania przez algorytm nadmiernie dopasowanego drzewa decyzyjnego. Dlatego do dalszych badań przyjmujemy wartość 10.

### Badanie 3 – Współczynnik Miary Atrybutu

W tym eksperymencie przyjmujemy dla pozostałych parametrów wartości, które uznaliśmy za najlepsze w poprzednich eksperymentach, czyli - **maksymalną głębokość drzewa** o wartości 5 i **minimalna ilość przykładów w węźle** wynoszącą 10.

### Wyniki eksperymentów:

Stosowany współczynnik miary	Dokładność klasyfikacji [%]	Precyzja klasyfikacji [%]	Czułość klasyfikacji [%]	F1-score [%]
Entropia	94,79	87,61	89,19	88,39
Współczynnik Gini'ego	94,71	88,29	87,5	87,89

**Tabela 8. Wyniki klasyfikacji dla różnych współczynników miary atrybutu na zbiorze weryfikującym dla naszej implementacji algorytmu – zestaw przykładów donorów**

Minimalna ilość przykładów	Dokładność klasyfikacji [%]	Precyzja klasyfikacji [%]	Czułość klasyfikacji [%]	F1-score [%]
Entropia	92,42	82,73	81,98	82,35
Współczynnik Gini'ego	92,00	81,82	81,08	81,45

**Tabela 9. Wyniki klasyfikacji dla różnych współczynników miary atrybutu na zbiorze weryfikującym dla implementacji algorytmu z biblioteki scikit-learn – zestaw przykładów donorów**

Analiza wyników badania wpływu wybranej miary współczynników Miary Atrybutu pokazuje nam bardzo zbliżone wyniki działania algorytmu dla obu współczynników miar. Dla żadnej miary jakości, prócz czułości klasyfikacji w naszej implementacji (dla której entropia prezentuje lepszą czułość o 1,69 pkt. procentowego) nie widzimy różnicy w konkretnej mierze większej niż o 1 punkt procentowy, co przy rzędach wartościach ponad 80% nie jest tolerowalną różnicą. Warto jednak zauważyć, że we wszystkich miarach jakości lepsze wyniki prezentuje entropia. Możemy więc dojść do konkluzji, że w sytuacjach w których nie mamy przeciwwskazań do narzędzi matematycznych jak logarytm entropia jest lepszym wyborem. Współczynnik Gini'ego jednak warto pamiętać wymaga do realizacji tylko dostępnej operacji potęgi (dokładnie drugiego stopnia), co może mieć znaczenie przy ewentualnych ograniczeniach w dostępności bibliotek, które oferują realizację operacji logarytmu. Jest to jednak rzadki przypadek stąd w powszechnych sytuacjach entropia powinna być preferowanym Współczynnikiem Miary Atrybutu

## Eksperymenty dla zbioru akceptorów

### Badanie wpływu zbioru testującego i weryfikującego na skuteczność klasyfikacji:

Badanie tak jak w przypadku zestawu donorów będzie przeprowadzone na próbie 50 zbiorów trenujących i weryfikujących uzyskanych przez losowanie dostarczonych danych w badanej w danym eksperymencie proporcji z zachowaniem stosunku przykładów pozytywnych i negatywnych niepodzielonego zbioru. Na cele eksperymentu przyjmujemy następujące parametry: **współczynnik miary atrybutu** – entropia, **minimalna liczba przykładów - 3**, **maksymalna głębokość - 100**. W celach weryfikacji eksperymenty przeprowadzamy również na implementacji drzewa decyzyjnego zaczerpniętego z biblioteki scikit-learn.

Współczynnik danych uczących do wszystkich danych	Dokładność klasyfikacji dla zbioru testowego [%]		Dokładność klasyfikacji dla zbioru weryfikującego [%]	
	nasza implementacja	Biblioteka scikit-learn	nasza implementacja	Biblioteka scikit-learn
<b>0.4</b>	99,70	99,85	83,47	80,88
<b>0.5</b>	99,71	99,84	83,80	81,51
<b>0.6</b>	99,69	99,85	84,25	81,95
<b>0.7</b>	99,73	99,86	84,31	81,89
<b>0.8</b>	99,73	99,87	84,69	82,23
<b>0.9</b>	<b>99,73</b>	<b>99,87</b>	<b>84,82</b>	<b>83,02</b>
<b>0.95</b>	99,76	99,89	85,45	82,88
<b>0.98</b>	99,76	99,89	85,17	82,52
<b>0.99</b>	99,75	99,90	83,59	82,00

**Tabela 10. Dokładność klasyfikacji na zbiorze testującym i weryfikującym dla naszej implementacji algorytmu – zestaw przykładów akceptorów**

Z wyników testów widzimy, dwie obserwacje. Po pierwsze warto zauważyć, że na przestrzeni wszystkich badanych wartości parametrów algorytm naszej implementacji zarówno dla zbioru danych trenujących jak i co ważniejsze dla zbioru danych weryfikujących prezentuje lepszą dokładność, niż implementacja algorytmu dostarczona przez bibliotekę scikit-learn - tak samo jak w przypadku tego badania dla klasyfikacji donorów. Jest to dobry wyznacznik tego, że nasza implementacja algorytmu jest poprawna i w pełni działająca. Co do analizy dokładnych wyników, widzimy, że podobnie jak w przykładach klasyfikacji donorów proporcja w jakiej dokonamy podziału na zbiór trenujący i weryfikujący nie wpływa w sposób znaczący na dokładność klasyfikacji. Z racji więc na chęć uniknięcia nadmiernego dopasowania drzewa do zbioru trenującego wybierzemy do dalszych działań podział o współczynniku 0.9, który zaprezentował trzeci najlepszy wynik testowy ze wszystkich badanych wartości nie zbliżając się przy tym do wartości 0.95 i wyższych, które mogą wiązać się z ryzykiem nadmiernego dopasowania oraz mały zbiór przykładów testowych do dobrej weryfikacji działania algorytmu w dalszych badaniach.



## Badanie wpływu kryteriów stopu na skuteczność algorytmu:

Badania z racji na długi czas przeprowadzania eksperymentów dla próbki 50 zbiorów będą dalej przeprowadzona na próbie 25 zbiorów trenujących i weryfikujących uzyskanych przez losowanie dostarczonych danych w stosunku 0.9 (który w toku poprzednich eksperymentów uznaliśmy za najlepszy) z zachowaniem stosunku przykładów pozytywnych i negatywnych niepodzielonego zbioru.

### Badanie 1 – maksymalna głębokość drzewa

W tym eksperymencie przyjmujemy za **minimalną ilość przykładów w węźle** wartość 3, a za **współczynnik miary atrybutu** entropie.

#### Wyniki eksperymentów:

Maksymalna głębokość drzewa	Dokładność klasyfikacji [%]	Precyzja klasyfikacji [%]	Czułość klasyfikacji [%]	F1-score [%]
5	84,91	66,22	44,13	52,97
10	<b>85,74</b>	<b>62,18</b>	<b>66,67</b>	<b>64,35</b>
20	84,64	60,00	62,16	61,06
40	84,79	60,17	64,00	62,01
60	84,91	60,68	63,96	62,28
80	85,29	61,54	64,86	63,16
100	84,95	61,06	62,16	61,61
150	84,84	60,87	63,06	61,95
200	84,89	60,34	63,06	61,67
400	85,35	61,54	64,86	63,16

**Tabela 11. Wyniki badań maksymalnej głębokości drzewa na zbiorze weryfikującym dla naszej implementacji algorytmu – zestaw przykładów akceptorów**

Maksymalna głębokość drzewa	Dokładność klasyfikacji [%]	Precyzja klasyfikacji [%]	Czułość klasyfikacji [%]	F1-score [%]
5	82,93	58,33	40,54	47,87
10	<b>83,92</b>	<b>58,89</b>	<b>56,76</b>	<b>57,80</b>
20	82,77	55,65	57,66	56,64
40	82,79	55,65	57,66	56,64
60	82,41	54,70	57,14	55,90
80	83,10	56,14	57,66	56,89
100	82,78	55,26	56,76	56,00
150	82,56	54,87	55,86	55,36
200	82,58	54,87	55,86	55,36
400	82,89	55,26	56,76	56,00

**Tabela 12. Wyniki badań maksymalnej głębokości drzewa na zbiorze weryfikującym dla implementacji algorytmu z biblioteki scikit-learn – zestaw przykładów akceptorów**

W przypadku zadania klasyfikacji akceptorów warto zwrócić na początku uwagę na znacząco gorsze wyniki działania algorytmu pod względem precyzji, czułości i miary F1 – odnotowany jest spadek z wartości przekraczających 80 procent do wartości lekko ponad 60 procent dla naszej implementacji i

wartości ponad 50 procent dla implementacji z biblioteki scikit-learn. Pokazuje to, że klasyfikacja akceptorów jest znacznie trudniejszym zadaniem dla algorytmu drzewa decyzyjnego niż klasyfikacja donorów. Przyczyna tego w naszej opinii leży w fakcie, że przykłady akceptorów składają się, aż z 90 przykładów co jest 6-krotnie większą ilością atrybutów. Sprawia to, że algorytm ma znacznie większą ilość możliwości podziału i jest mu znacznie trudniej pozyskiwać wiedzę na temat klasyfikacji. Co do wniosków z porównania implementacji algorytmu bez zmian względem zadania klasyfikacji donorów lepsze wyniki prezentuje nasza implementacja. Na temat naszych przemyśleń co do powodu tego stanu wypowiedzieliśmy się we wcześniejszych wnioskach, dlatego tutaj zostaną one pominięte. Za wartość parametru, dla którego algorytm sprawował się najlepiej uznaliśmy maksymalną głębokość drzewa 10. Uzasadnieniem tej decyzji jest najlepsze sprawowanie się algorytmu dla tej maksymalnej głębokości drzewa pod względem dokładności, czułości i miary F1. W przypadku precyzji uległ on co prawda parametrowi ustawionemu na 5, jednak jego czułość o wartości 44,13% i co za tym idzie niski F1-score= 52,97 (w obu tych miarach zdecydowanie najgorzej sprawująca się z testowanych wartości parametru) sprawiają, że nadal wartość 10 stanowczo wypada najlepiej na tle konkurencji.

## Badanie 2 – minimalna ilość przykładów w węźle

W tym eksperymencie analogicznie jak przy testach dla zadania donorów przyjmujemy za **maksymalną głębokość drzewa** wartość 10, którą uznaliśmy w poprzednim eksperymencie za najlepszą wartość parametru, a jako **współczynnik miary atrybutu** przyjmujemy ponownie entropie.

### Wyniki eksperymentów:

Minimalna ilość przykładów	Dokładność klasyfikacji [%]	Precyzja klasyfikacji [%]	Czułość klasyfikacji [%]	F1-score [%]
40	<b>87,17</b>	<b>67,92</b>	<b>64,86</b>	<b>66,36</b>
20	86,32	64,60	65,77	65,18
10	86,23	63,11	69,37	66,09
5	86,35	64,35	66,67	65,49
3	86,10	63,56	66,96	65,22
2	85,65	62,18	66,67	64,35

**Tabela 13. Wyniki badań minimalnej ilości przykładów w węźle na zbiorze weryfikującym dla naszej implementacji algorytmu – zestaw przykładów akceptorów**

Minimalna ilość przykładów	Dokładność klasyfikacji [%]	Precyzja klasyfikacji [%]	Czułość klasyfikacji [%]	F1-score [%]
40	<b>85,09</b>	<b>63,27</b>	<b>55,86</b>	<b>59,33</b>
20	84,72	62,00	55,86	58,77
10	83,64	58,10	54,95	56,48
5	83,92	58,88	56,76	57,80
3	84,18	60,00	56,76	58,33
2	83,88	58,88	56,76	57,80

**Tabela 14. Wyniki badań minimalnej ilości przykładów w węźle na zbiorze weryfikującym dla implementacji algorytmu z biblioteki scikit-learn – zestaw przykładów akceptorów**

Analiza wyników eksperymentu po pierwsze umacnia wnioski wyciągnięte na podstawie poprzedniego eksperymentu na temat trudności zadania klasyfikacji akceptorów w porównaniu z zadaniem klasyfikacji donorów i powiązane z tym obserwacje na temat gorszej jakości działania

algorytmu - szczególnie pod względem precyzji, czułości klasyfikacji i F1-score. Ponownie nasza implementacja algorytmu wykazuje w pełnym spektrum wartości testowanych lepsze wyniki miar jakości niż implementacja dostarczona przez bibliotekę scikit-learn. Szczególnie lepsze wyniki widzimy w ramach czułości klasyfikacji i F1-score. W naszej opinii najlepiej sprawującym się parametrem dla naszej implementacji jest zaskakująco względem przewidywań na początku eksperymentu wartość minimalnej ilości 40 przykładów w węźle, który najlepiej wykazał się pod względem precyzji klasyfikacji sprawując się o ponad 3 pkt. procentowe lepiej niż następna w tej mierze wartość 20 przykładów. Co prawda wartość parametru 40 zanotowała nieznacznie gorszą (o 0,91 pkt. procentowego) niż wartość 20 czułość klasyfikacji, jednak lepszą dla wartości 40 miara F1-score i dokładność przekonuje nas do uznania jej za najlepszą. Wspomniane wcześniej zaskoczenie wynikało z faktu, że na wstępie do eksperymentu przewidywaliśmy, że z racji na większą liczbę możliwych do wyboru atrybutów, algorytm do lepszego działania będzie potrzebował dzielenia przykładów na mniejsze zbiory niż przy zadaniu klasyfikacji donorów. Okazuje się jednak, że większa liczba atrybutów niesie ze sobą tylko większe ryzyko nadmiernego dopasowania drzewa, co sprawia, że drobniejszy podział wpływa negatywnie na jakość klasyfikacji.

### Badanie 3 – Współczynnik Miary Atrybutu

W tym eksperymencie przyjmimy dla pozostałych parametrów wartości, które uznaliśmy za najlepsze w poprzednich eksperymentach, czyli - **maksymalną głębokość drzewa** o wartości 10 i **minimalna ilość przykładów w węźle** wynoszącą 40.

**Wyniki eksperymentów:**

Stosowany współczynnik miary	Dokładność klasyfikacji [%]	Precyzja klasyfikacji [%]	Czułość klasyfikacji [%]	F1-score [%]
Entropia	87,37	68,22	65,77	66,97
Współczynnik Gini'ego	86,35	65,42	63,06	64,22

**Tabela 15. Wyniki klasyfikacji dla różnych współczynników miary atrybutu na zbiorze weryfikującym dla naszej implementacji algorytmu – zestaw przykładów akceptorów**

Minimalna ilość przykładów	Dokładność klasyfikacji [%]	Precyzja klasyfikacji [%]	Czułość klasyfikacji [%]	F1-score [%]
Entropia	85,03	62,62	55,86	59,05
Współczynnik Gini'ego	84,29	61,54	50,45	55,45

**Tabela 16. Wyniki klasyfikacji dla różnych współczynników miary atrybutu na zbiorze weryfikującym dla implementacji algorytmu z biblioteki scikit-learn – zestaw przykładów akceptorów**

Wyniki eksperymentu uwypuklają obserwacje dostrzeżone przy porównaniu Współczynników Miar Atrybutu dla zadania klasyfikacji donorów. W tym przypadku współczynnik Gini'ego sprawuje się już zauważalnie gorzej niż entropia, co wyraźnie widać przy precyzji klasyfikacji, czułości i F1-score (odpowiednio o 2,8 pkt. procentowego, 2,71 pkt. procentowego i 2,75 pkt. procentowego na korzyść entropii). Ponadto czułość klasyfikacji dla implementacji z biblioteki scikit-learn jest, aż o 5,41 pkt.

procentowego gorsza przy współczynniku Gini'ego, a F1-score dla tej samej implementacji jest gorszy dla współczynnika dla Gini'ego o 3,6 pkt. procentowego w porównaniu z entropią. Pokazuje to nam, że wraz ze wzrostem liczby atrybutów w przykładach współczynnik Gini'ego sprawuje się coraz gorzej od entropii, a przedstawione przy tym samym porównaniu w zadaniu klasyfikacji donorów wskazania na korzyść współczynnika Gini'ego tracą na znaczeniu.

## Podsumowanie

Podsumowując wyniki naszych badań. Algorytm drzewa decyzyjnego w zadaniu klasyfikacji posiada bardzo sporo zalet. Jednym z najważniejszych jest sama prostota algorytmu, który jest prosty w zrozumieniu i implementacji, co czyni go bardzo dobrym narzędziem do realizacji prostszych zadań klasyfikacji. Następnie sama jakość klasyfikacji jest bardzo dobra – warto tutaj jednak mieć na uwadze, że drzewo decyzyjne potrzebuje do działania przyzwoitej ilości przykładów obu klas – dla naszych danych przykłady pozytywne stanowiły zaledwie około 20 procent wszystkich dostępnych przykładów, co sprawia, że algorytm ma znacznie trudniej pozyskać wiedzę na temat tych klas, co można zaobserwować w gorszej jakości precyzji uzyskanych w badaniach zadania klasyfikacji akceptorów, które z racji na więcej atrybutów w przykładach naturalnie potrzebowały więcej informacji o obu klasach, by móc bardzo dobrze klasyfikować obie klasy. Dodatkowo do wad algorytmu trzeba przypisać dużą wrażliwość jego na nadmierne dopasowanie. Wymusza to na osobie implementującej i dobierającej parametry działania drzewa dużą ostrożność w procesie doboru parametrów. Na koniec warto jeszcze wspomnieć o jednej zalecie algorytmu, którą wykazaliśmy realizując projekt - elastyczność. Algorytm drzewa decyzyjnego ze względu na swoją bardzo prostą i ogólną logikę daje programiście implementującemu algorytm duże możliwości przystosowania go do swojego przypadku. Algorytm na spokojnie radzi sobie z atrybutami ciągłymi, dyskretnymi (binarnymi, ale także takimi jak w naszym projekcie, gdzie atrybuty mogły przyjmować cztery, a w praktyce po analizie zbiorów przykładów nawet 6 wartości), a także ich kombinacje. Ponadto możemy stosować podział, który będzie wybierał jedną wartość atrybutu za kryterium albo jak w naszym algorytmie stosował alternatywę wartości atrybutów. Reasumując algorytm drzewa decyzyjnego jest bardzo dobrym i prostym algorytmem uczenia maszynowego do realizacji zadania klasyfikacji, który warto brać pod uwagę przy realizowaniu projektu wykorzystującego uczenie maszynowe.

W ramach projektu nauczyliśmy się realizować algorytm drzewa decyzyjnego, zrozumieliśmy metody eksperymentowania i badania algorytmów uczenia maszynowego, a także korzystać z narzędzi dostarczonych przez biblioteki numpy i scikit-learn.