Wprowadzenie do Data Science i metod uczenia maszynowego **2020/2021**

Prowadzący: mgr inż. Rafał Woźniak Wtorek, 13:15

Szymon Gruda	239661	239661@edu.p.lodz.pl
Jan Karwowski	239671	239671@edu.p.lodz.pl
Michał Kidawa	239673	239673@edu.p.lodz.pl
Kamil Kowalewski	239676	239676@edu.p.lodz.pl

Zadanie 5.: Problem Set 5

Spis treści

1. Wprowadzenie

Do poniższych zadań klasyfikacji oraz klasteryzacji zostały wykorzystane takie same zbiory jak w zadaniu 3 - klasyfikacja oraz zadaniu 4 - klasteryzacja.

1.1. Klasyfikacja

Do badań zostały wykorzystane następują klasyfikatory z wybranymi parametrami.

Zbiór Heart:

— KNN - *k=9*, *metryka=Euclidesowa*
— Bayes - brak parametrów
— SVM - *kernel=poly*, *C=1.6*, *gamma=0.0001*
— Lasy losowe - *min_samples_leaf=10*, *n_estimators=50*, *max_samples=0.5*

Zbiór Gestures:

— KNN - *k=9*, *metryka=Euclidesowa*
— Bayes - brak parametrów
— SVM - *kernel=rbf*, *C=2.0*, *gamma=0.0001*
— Lasy losowe - *min_samples_leaf=8*, *n_estimators=500*

Zbiór Weather:

— KNN - *k=9*, *metryka=Euclidesowa*
— Bayes - brak parametrów
— SVM - *kernel=rbf*, *C=1.6*, *gamma=0.001*
— Lasy losowe - *max_depth=5*, *n_estimators=200*, *max_samples=0.05*

1.2. Klasteryzacja

Do badań zostały wykorzystane następujące modele klasteryzacji z wybranymi parametrami.

Zbiór Iris:

— K-Means - *k=3*
— Metoda aglomeracyjna - *k=3*, *metryka=Manhattan*, *linkage=complete*
— Metoda expectation-maximization - *k=4*, *covariance_type=full*, *max_iter=200*,
— DBSCAN - *min_samples=7*, *epsilon=0.9*, *metryka=minkowski*

Zbiór Customers:

— K-Means - *k=6*
— Metoda aglomeracyjna - *k=6*, *linkage=Ward*
— Metoda expectation-maximization - *k=6*, *covariance_type=diag*, *max_iter=200*,
— DBSCAN - *min_samples=7*, *epsilon=23*, *metryka=minkowski*

Zbiór Moons:

— K-Means - *k=8*
— Metoda aglomeracyjna - *k=8*, *linkage=Ward*
— Metoda expectation-maximization - *k=9*, *covariance_type=full*, *max_iter=200*,
— DBSCAN - *min_samples=5*, *epsilon=0.2*, *metryka=minkowski*

2. Wyniki

2.1. Klasyfikacja

2.1.1. Zbiór Heart

Classifier	Accuracy	Sensitivity	Specificity	Precision
knn	0.7143	0.7292	0.6977	0.7292
bayes	0.8132	0.7708	0.8605	0.8605
svm	0.8242	0.8958	0.7442	0.7963
random forest	0.8791	0.875	0.8837	0.8936

Tabela 1.

30	13
13	35

Tabela 2. knn

37	6
11	37

Tabela 3. bayes

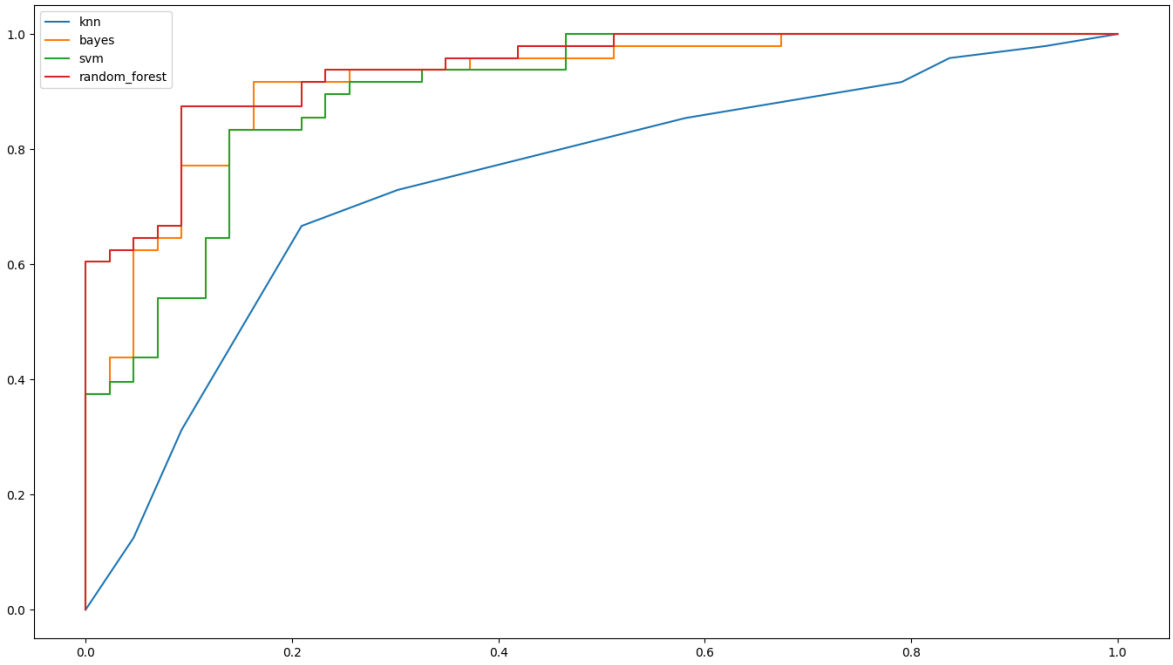
32	11
5	43

Tabela 4. svm

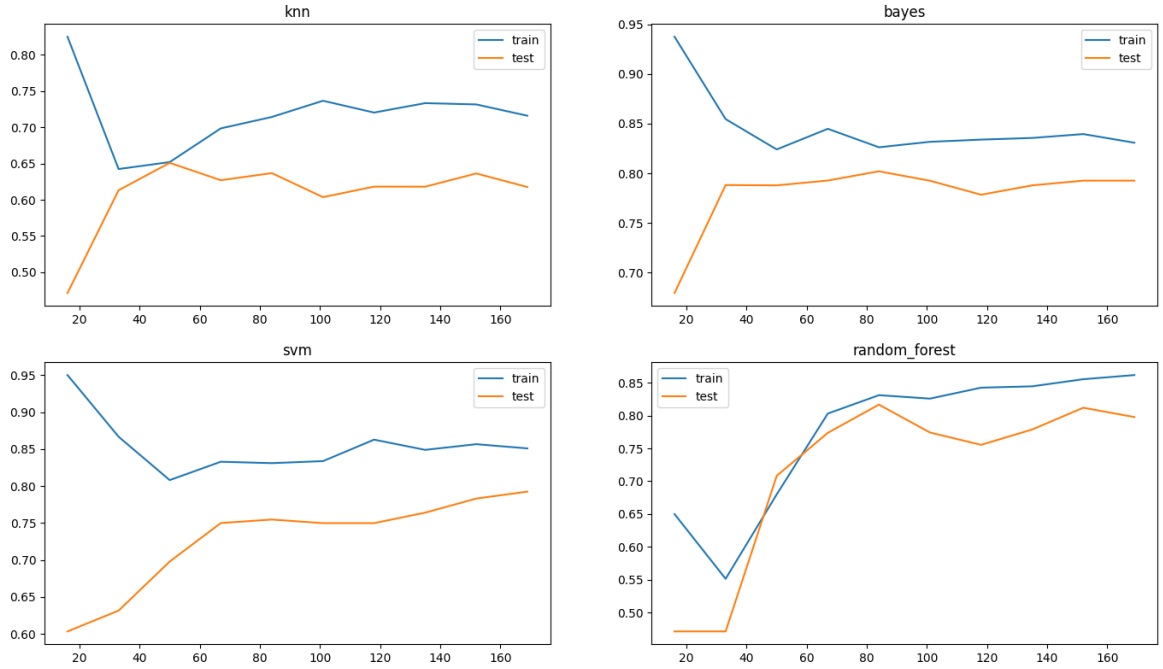
38	5
6	42

Tabela 5. random forest

Tabela 6. Macierze pomyłek



Rysunek 1. Krzywa ROC



Rysunek 2. Krzywa uczenia

2.1.2. Zbiór Gestures

Classifier	Accuracy	Sensitivities	Precisions
knn	0.6821	[0.8708 0.9374 0.1254 0.7758]	[0.9949 0.5744 0.8926 0.5845]
bayes	0.8779	[0.9157 0.9451 0.9048 0.7378]	[0.924 0.8206 0.9431 0.8316]
svm	0.8913	[0.9674 0.9681 0.7793 0.8422]	[0.9535 0.8819 0.9129 0.8189]
random forest	0.9118	[0.9551 0.9 0.9373 0.8529]	[0.9169 0.9457 0.9129 0.8694]

Tabela 7.

775	74	9	32
0	853	0	57
3	374	108	376
1	184	4	654

Tabela 8. knn

815	1	8	66
0	860	28	22
15	29	779	38
52	158	11	622

Tabela 9. bayes

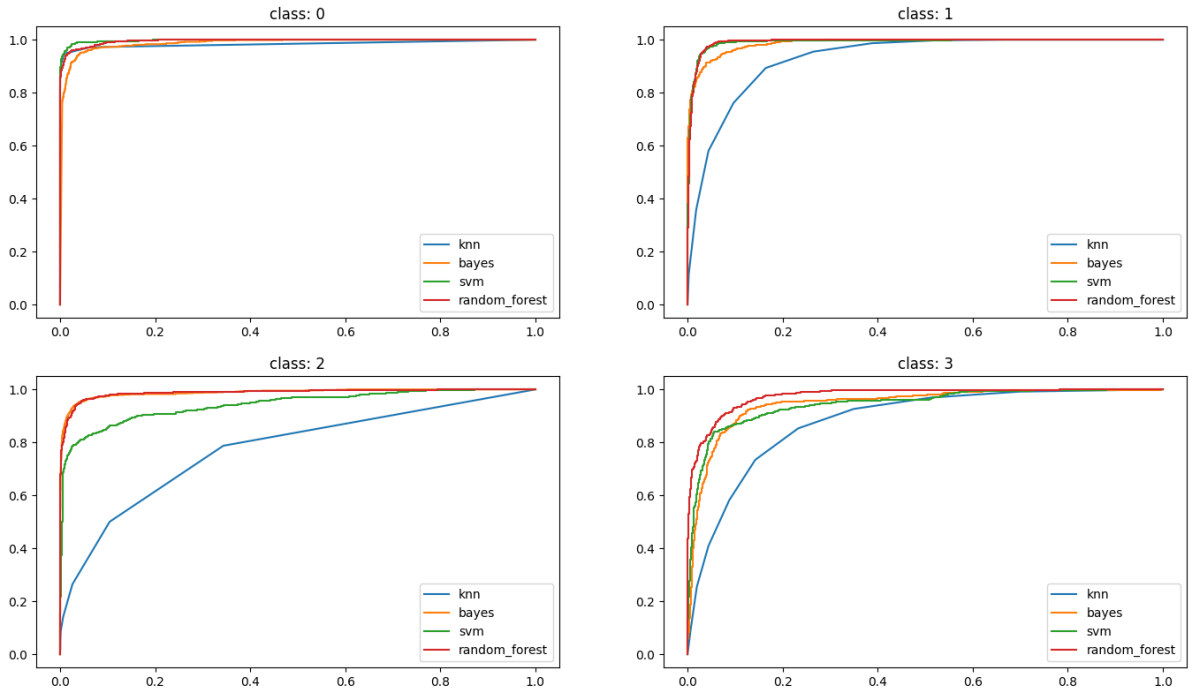
861	15	14	0
5	881	4	20
11	42	671	137
26	61	46	710

Tabela 10. svm

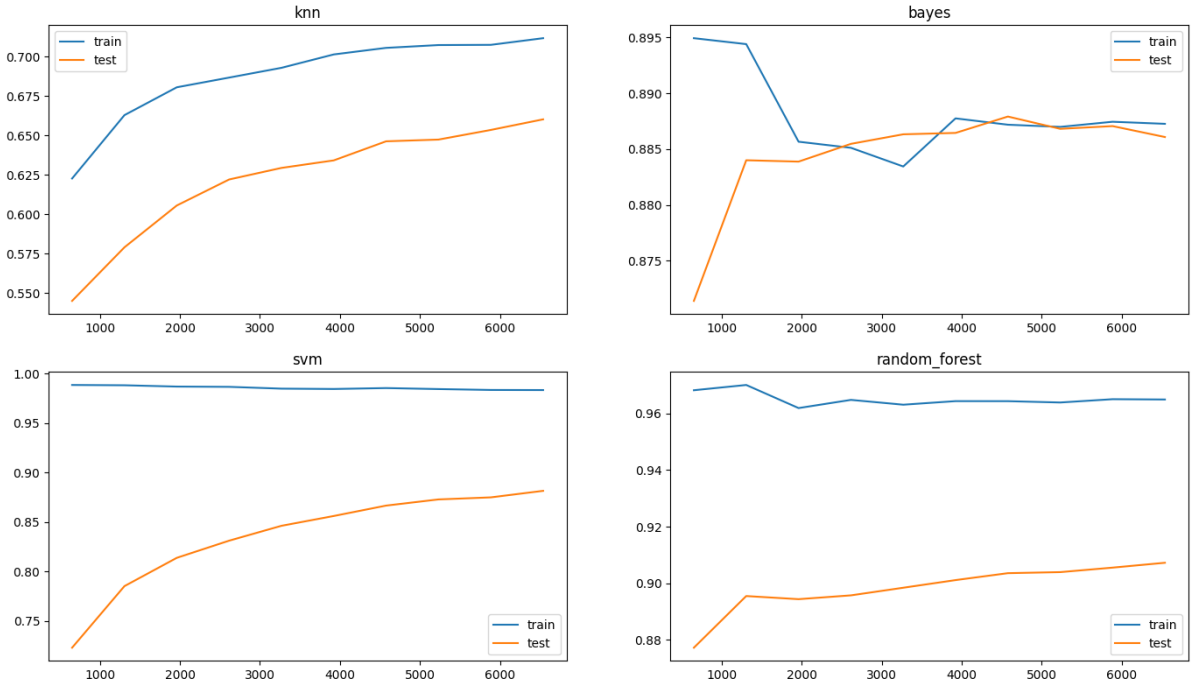
850	0	12	28
0	819	37	54
14	14	807	26
63	33	28	719

Tabela 11. random forest

Tabela 12. Macierze pomyłek



Rysunek 3. Krzywa ROC



Rysunek 4. Krzywa uczenia

2.1.3. Zbiór Weather

Classifier	Accuracy	Sensitivity	Specificity	Precision
knn	0.849	0.4936	0.9505	0.74
bayes	0.8004	0.6822	0.8342	0.5402
svm	0.8604	0.5346	0.9534	0.7663
random forest	0.85	0.4367	0.968	0.7959

Tabela 13. Weather basic metrics

12514	652
1904	1856

Tabela 14. knn

10983	2183
1195	2565

Tabela 15. bayes

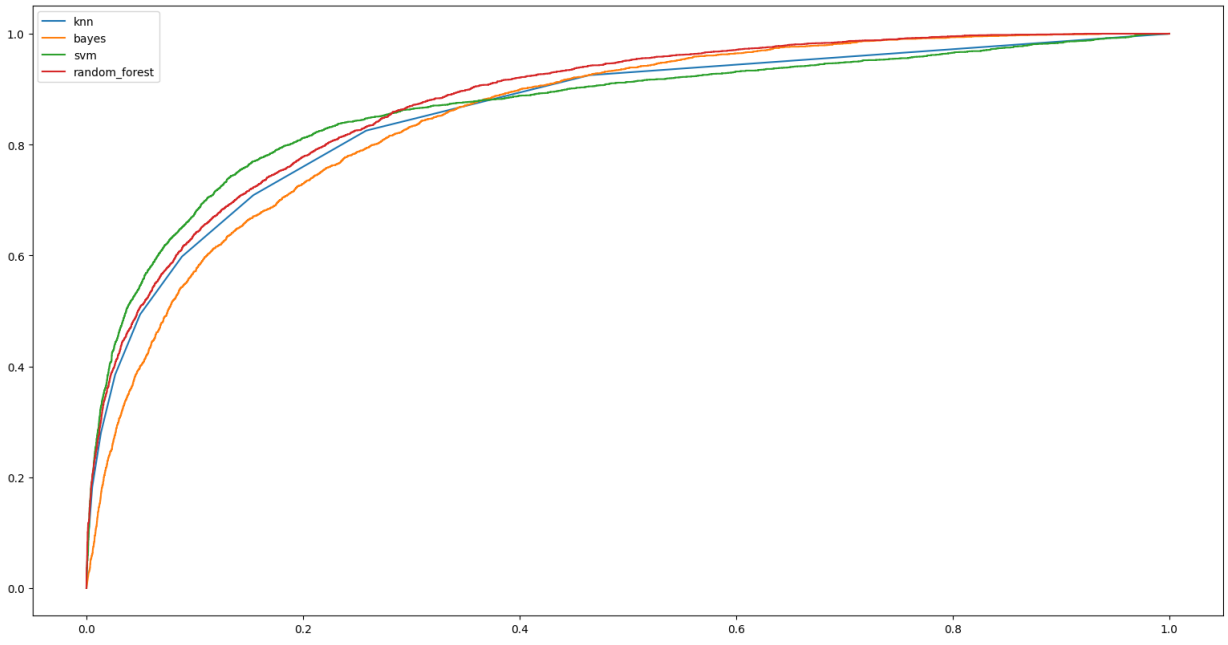
12553	613
1750	2010

Tabela 16. svm

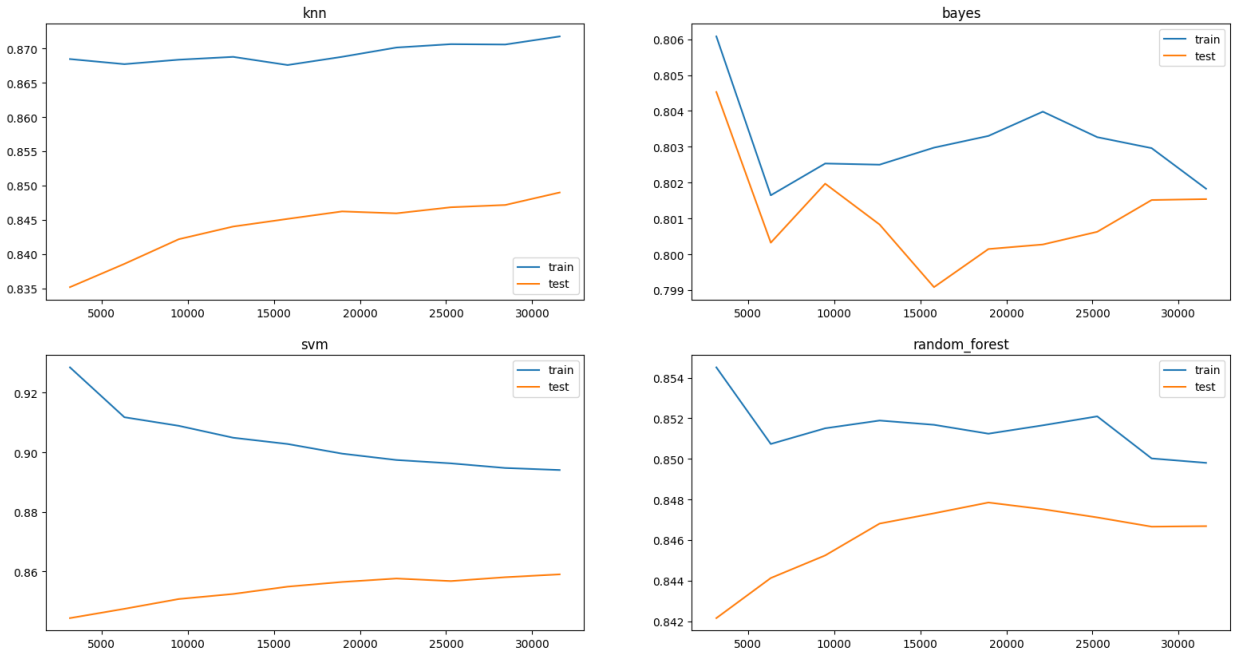
12745	421
2118	1642

Tabela 17. random forest

Tabela 18. Macierze pomyłek



Rysunek 5. Krzywa ROC



Rysunek 6. Krzywa uczenia

2.2. Klasteryzacja

Classifier	Silhouette	Calinski_Harabasz	Davies_Bouldin	Rand_score	Fowlkes_Mallows
k_means	0.553	560.4	0.662	0.88	0.821
agglomerative	0.554	551.057	0.654	0.88	0.824
expectation maximization	0.409	399.935	0.953	0.917	0.866
db_scan	0.686	501.925	0.384	0.776	0.771

Tabela 19. Wyniki wybranych metryk dla zbioru danych Iris

Classifier	Silhouette	Calinski_Harabasz	Davies_Bouldin
k_means	0.452	166.576	0.745
agglomerative	0.443	159.329	0.769
expectation_maximization	0.446	163.052	0.739
db_scan	0.39	4.613	0.42

Tabela 20. Wyniki wybranych metryk dla zbioru danych Customers

Classifier	Silhouette	Calinski_Harabasz	Davies_Bouldin	Rand_score	Fowlkes_Mallows
k_means	0.512	1617.388	0.583	0.622	0.493
agglomerative	0.503	1517.307	0.573	0.627	0.504
expectation maximization	0.498	1630.128	0.585	0.614	0.476
db_scan	0.325	456.067	1.16	1.0	1.0

Tabela 21. Wyniki wybranych metryk dla zbioru danych Moons

3. Dyskusja

3.1. Klasyfikacja

Algorytm K Najbliższych Sąsiadów sprawuje się zdecydowanie gorzej od pozostałych. Widać to bardzo wyraźnie, w przypadku zbiorów danych „hearts" oraz „gestures" oraz trochę mniej wyraźnie w przypadku zbioru danych „weather". Zaczynając od pierwszego z nich, już po macierzy pomyłek można łatwo spostrzec słabość tego algorytmu. Widać też, że zarówno sensitivity, specificity jak i precision są niższe niż dla pozostałych algorytmów, samo accuracy również. Na wykresie krzywych ROC niebieska linia jest znacznie niżej od pozostałych. W przypadku drugiego zbioru danych sytuacja jest bardzo podobna - wartości wszystkich metryk są zdecydowanie niższe. Ponadto można dostrzec, że klasyfikator w ogóle nie nauczył się rozpoznawać przykładów z klasy trzeciej i ma tendencję do przyporządkowywania przykładów do klasy drugiej. Jest zdecydowanie niedouczony, co widać dla tego zbioru również po krzywej uczenia. Również krzywe ROC prezentują jego niższość, wobec pozostałych metod. Jeżeli chodzi o trzeci zbiór danych, to tutaj wszystkie cztery krzywe ROC niemalże się pokrywają. Klasyfikatory sprawują się więc stosunkowo podobnie. Można jednak zaryzykować stwierdzenie, że KNN podjął tutaj słaby kompromis między sensitivity a specificity. To pierwsze ma bardzo niską wartość, a to drugie bardzo wysoką. Jego wyniki są najbardziej podobne do działania lasu losowego, wykazującego się podobną tendencją, dla zbioru „weather". Klasyfikator KNN ma jeszcze tę cechę, że prawie zawsze, więcej przykładów uczących mu pomaga osiągnąć lepsze wyniki, zarówno na zbiorze uczącym jak i testowym - rosnące krzywe uczenia.

Naiwny Klasyfikator Bayesa najbardziej różni się od pozostałych algorytmów krzywymi uczenia. Są one najbardziej chaotyczne i zbiegają się, a czasem nawet przecinają. W przypadku zbioru danych „weather" są to ruchy prawie zupełnie losowe, gdyż zakres na osi dokładności jest bardzo niewielki. W przypadku pozostałych zbiorów, wyniki dla zbioru testowego przy małej ilości próbek są wyższe, niż dla innych metod. Klasyfikator ten ma mniejszą tendencję do przetrenowania i szybciej się uczy. Często niewiele daje mu więcej próbek uczących. Jego krzywe ROC są zbliżone do krzywych lasu losowego i SVM (poza ostatnim zbiorem danych, gdzie wszystkie są podobne). Trudno porównać jego dokładność z pozostałymi algorytmami. Te trzy,

poza KNN, sprawdzają się bardzo podobnie i jedne osiągają wyższą czułość dla jednych klas, a drugie dla innych. Tutaj należałoby stwierdzić, co jest właściwym zadaniem klasyfikatora i czy powinien on bardziej skupiać się na eliminowaniu błędów pierwszego, czy drugiego rodzaju.

Maszyna Wektorów Nośnych sprawuje się podobnie jak Naiwny Klasyfikator Bayesa i Lasy Losowe. Charakteryzuje się natomiast tym, że trochę podobnie jak w przypadku KNN, zdecydowanie więcej przykładów uczących poprawia wyniki na zbiorze testowym. Nie zawsze dzieje się tak natomiast w przypadku Naiwnego Klasyfikatora Bayesa i Lasów Losowych. Co do ciekawszych spostrzeżeń, to widać również, że dla zbioru „weather" udaje się klasyfikatorowi SVM osiągnąć chyba najlepszy kompromis między czułością a specyficznością. Krzywa ROC jest najbardziej zbliżona do lewego, górnego rogu. W przypadku zbioru „gestures" jest ona natomiast, dla niektórych klas, gorsza niż pozostałe algorytmy, a dla innych lepsza, chociaż oczywiście zawsze lepsza, niż KNN. Być może wynika to z faktu, że algorytm ten jest przystosowany do klasyfikacji binarnej. Poza tymi uwagami, sprawuje się on, jak już wspomniano, bardzo podobnie do pozostałych dwóch lepszych algorytmów - kwestia zdefiniowania, co jest najbardziej istotne w danym zadaniu klasyfikacji.

Algorytm Lasów Losowych jest ostatnim i jednocześnie prawdopodobnie najlepszym algorytmem klasyfikacji, spośród testowanych w ramach tego zadania. Osiąga on najwyższe wartości dokładności, a krzywe ROC okazują się być tak samo dobre albo lepsze, niż dla pozostałych algorytmów. Dla zbioru „hearts" sprawuje się on zdecydowanie lepiej niż Naiwny Klasyfikator Bayesa i bardzo podobnie do SVM (zależy od tego, co w danym zadaniu jest krytyczne). Dla zbioru „gestures" również osiąga średnio najlepsze wyniki. Tutaj jednak dla różnych klas, różnie sprawują się różne klasyfikatory. Można jednak również zaryzykować stwierdzenie, że sprawdził się najlepiej - zwłaszcza, że w przypadku wielu klas znacznie trudniej wydzielić te bardziej istotne i uśredniony wynik (czyli miara accuracy) może okazać się najlepszy do porównania z innymi metodami. W zbiorze „weather" osiągnął on dokładność mniejszą niż SVM, który tutaj sprawuje się najlepiej. Co więcej, podobnie jak KNN mamy tutaj do czynienia z dużym naciskiem na jedną z dwóch klas. Tak więc klasyfikator ten nauczył się stosunkowo dobrze rozpoznawać klasę negatywną, natomiast próbki z klasy pozytywnej zostały zaklasyfikowane w mniej niż 50%. Pozwoliło to jednak osiągnąć najwyższą precyzję, ze wszystkich klasyfikatorów. Wracamy więc znowu do stwierdzenia, że najważniejsze to określić, które miary są krytyczne i najistotniejsze - to natomiast zależy od samego tematu - dziedziny badań.

3.2. Klasteryzacja

Podczas przeprowadzonych eksperymentów badaliśmy wyniki uzyskane przez algorytmy w postaci następujących metryk:

— wewnętrznych:
  — indeks Silhouette

  - indeks Calińskiego-Harabasza
  - indeks Daviesa-Bouldina
- zewnętrznych:
  - indeks Rand
  - indeks Fowlkes–Mallows

Ze względu na charakterystykę zbioru Customers (nie ma informacji o klasach, które mogą zostać przypisane), niemożliwe było wykorzystanie dla niego metryk zewnętrznych. W przypadku każdej z nich poza indeksem Daviesa-Bouldina, wyższa wartość oznacza lepszy wynik. Nie można jednak porównywać metryk bezpośrednio między sobą.

W przypadku metryki Silhouette określane jest podobieństwo obiektu do własnego skupienia w porównaniu do innych skupień. Indeks przyjmuje wartości z przedziału od -1 do +1, gdzie wysoka wartość wskazuje, że dany obiekt został dobrze dopasowany do własnego klastra i słabo dopasowany do sąsiednich skupień. Najlepsze wartości Silhouette score zostały osiągnięte dla metody k-średnich i algorytmu aglomeracyjnego dla zbiorów posiadających etykietę. Wartości silhoutte osiągały wyższe wyniki dla metody innej niż dbscan z wyjątkiem zbioru danych Irysów, gdzie właśnie dbscan zanotował najwyższy wynik, co może być skutkiem samej charakterystyki zbioru.

Indeks Calinski-Harabasz jest obliczany jako stosunek sumy dyspersji pomiędzy klastrami do sumy dyspersji wewnątrz klastra. Dla zbiorów w przypadku których analizowane było więcej próbek wartości indeksu osiągały większe wartości.

Davies-Bouldin to indeks, w przypadku którego sprawdzanie poprawności grupowania odbywa się przy wykorzystaniu ilości cech charakterystycznych dla zestawu danych. Wartości tej metryki osiągały wyższe wyniki dla każdej metody poza dbscan z wyjątkiem zbioru danych Moons.

Indeks Rand oblicza stopień podobieństwa między skupiskami poprzez porównanie wszystkich par próbek i obliczenie liczby próbek, które są przydzielone poprawnie, lub niepoprawnie do danego skupiska.

Indeks Fowlkes-Mallows jest obliczany jako wartość średniej geometrycznej miar precision oraz recall.

4. Wnioski

- Algorytm KNN sprawuje się zazwyczaj gorzej niż inne, bardziej złożone algorytmy klasyfikacji, potrzebuje również znacznie więcej przykładów uczących. Cechuje go natomiast prostota i łatwość w implementacji.
- Pozostałe algorytmy osiągają bardzo zbliżone wyniki, niektóre z pozoru mogą wydawać się lepsze od pozostałych poprzez osiągnięcie wyższej wartości miary *accuracy*. Po przeanalizowaniu wyników innych miar oraz samych macierzy pomyłek, widać wyraźnie, że każdy z nich radzi sobie, w jakimś aspekcie, lepiej od pozostałych. Tak więc kluczowym dla zadania klasyfikacji jest zdefiniować, która miara jest krytyczna - czy bardziej istotne są błędy pierwszego czy drugiego rodzaju, a może któraś z wielu różnych klas odgrywa najistotniejszą rolę.

— Dbscan zadziałał najlepiej w przypadku klasteryzacji zbioru Moons. Jako metoda gęstościową najlepiej poradził sobie z danymi zgodnie z "intuicją ludzką", co widać po wynikach metryk zewnętrznych
— Miary wewnętrzne nie określają jednoznacznie jakości klasyfikacji

Literatura

[1] https://www.kaggle.com/ronitf/heart-disease-uci
[2] https://www.kaggle.com/kyr7plus/emg-4
[3] https://www.kaggle.com/jsphyg/weather-dataset-rattle-package
[4] https://www.kaggle.com/uciml/iris
[5] https://www.kaggle.com/vjchoudhary7/customer-segmentation-tutorial-in-python
[6] https://scikit-learn.org/stable/modules/generated/sklearn.datasets.make_moons.html