Akademia Ekonomiczno-Humanistyczna w Warszawie

SPRAWOZDANIE

INTELIGENTNA ANALIZA DANYCH

**LAB3**

**ANALIZA TEKSTÓW,**

**PRZETRENOWANIE I OPTYMALIZACJA MODELU**

28.11.2021

JOANNA PRAJZENDANC

36358

MIŁOSZ SAKOWSKI

36381

**Spis treści**

[1. Cel i przebieg ćwiczenia 3](#_Toc27440)

[2. Definicje i założenia 3](#_Toc16619)

[2.1. Wyjaśnienie pojęć 3](#_Toc22854)

[3. Grid search 3](#_Toc23555)

[3.1. Zadanie #1 3](#_Toc26385)

[i. Treść polecenia 3](#_Toc26627)

[ii. Rozwiązanie 3](#_Toc11014)

[4. Przetrenowanie 3](#_Toc19431)

[4.1. Zadanie #2 3](#_Toc14652)

[i. Treść polecenia 3](#_Toc6210)

[ii. Rozwiązanie 3](#_Toc337)

[5. Analiza tekstu 3](#_Toc561)

[5.1. Analiza kodu 3](#_Toc8215)

[5.2. Zadanie #3 4](#_Toc9919)

[i. Treść polecenia 4](#_Toc27630)

[ii. Rozwiązanie 4](#_Toc25183)

[5.3. Zadanie #4 4](#_Toc30910)

[i. Treść polecenia 4](#_Toc9596)

[ii. Rozwiązanie 4](#_Toc30952)

[5.4. Zadanie #5 4](#_Toc13857)

[i. Treść polecenia 4](#_Toc19119)

[ii. Rozwiązanie 4](#_Toc4168)

[5.5. Zadanie #6 4](#_Toc15841)

[i. Treść polecenia 4](#_Toc24924)

[ii. Rozwiązanie 4](#_Toc15303)

# Cel i przebieg ćwiczenia

Celem ćwiczenia było utrwalenie wiedzy w zakresie

# Definicje i założenia

## Wyjaśnienie pojęć

W sprawozdaniu pojawiają się następujące pojęcia:

* grid search - sposób optymalizacji modelu polegający na zestawieniu ze sobą wszystkich możliwych konfiguracji zadanych parametrów
* przetrenowanie - stan modelu, który mimo idealnego dopasowania dla danych ze zbioru trenującego dopasowanie do danych ze zbioru testującego jest dużo niższe i niewystarczające

# Grid search

## Zadanie #1

### Treść polecenia

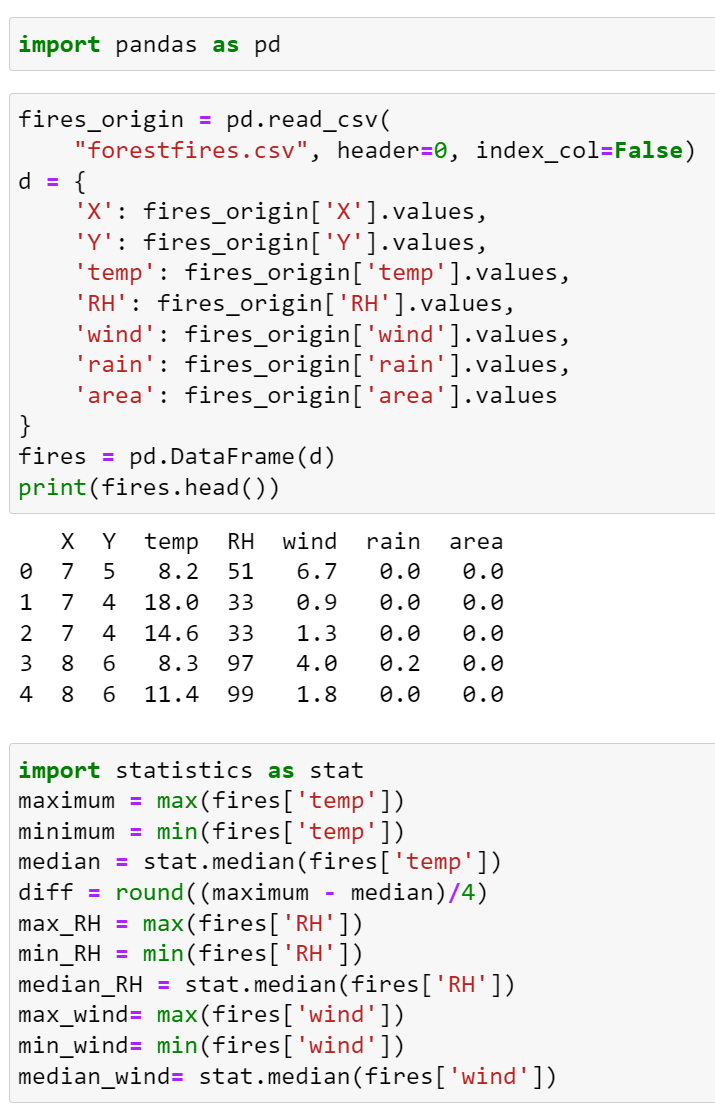
Proszę pobrać dowolny zbiór danych ze strony https://archive.ics.uci.edu/ml/index.php

Następnie proszę zoptymalizować las metodą Grid Search (trzy parametry modelu mają zostać optymalizowane).

Jakie wartości parametrów są najlepsze i dlaczego?

### Rozwiązanie

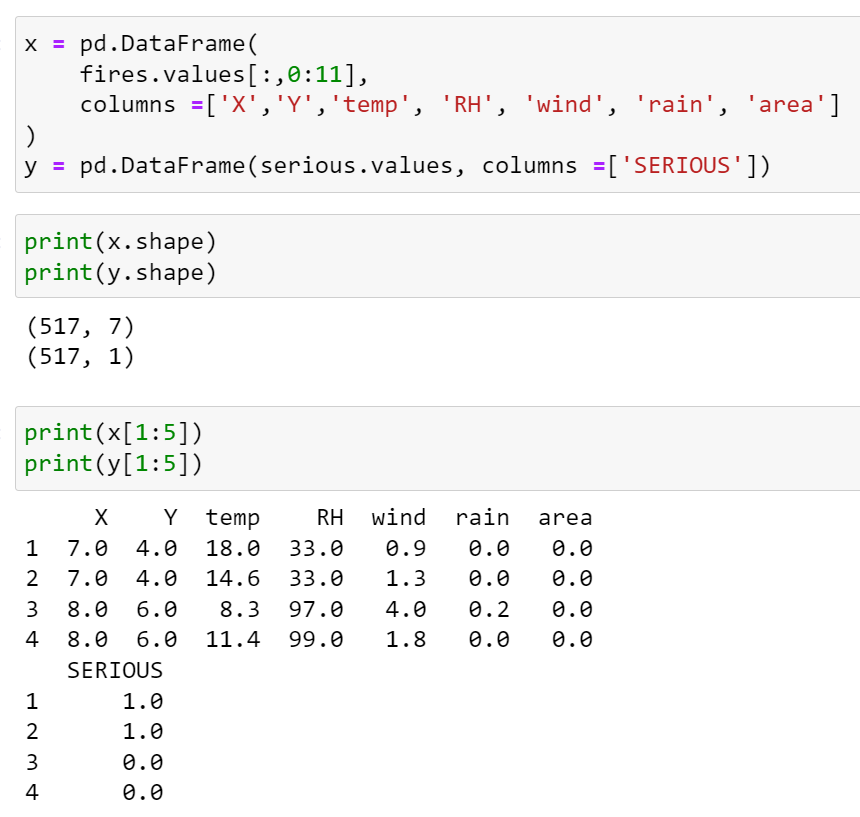
#### Zbiór danych



Obraz 1: Ładowanie bazy danych z pliku

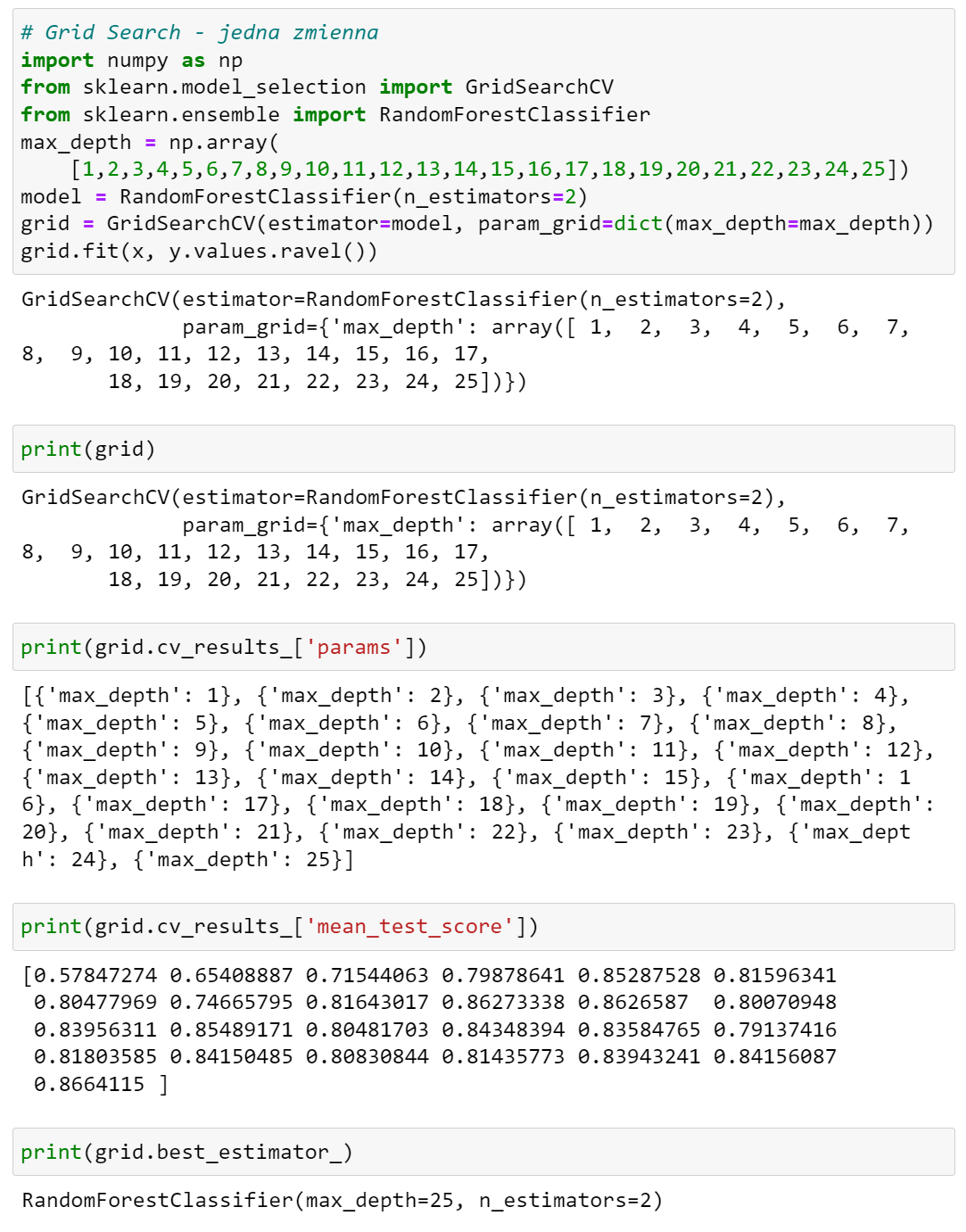


Obraz 2: Algorytm klasyfikujący

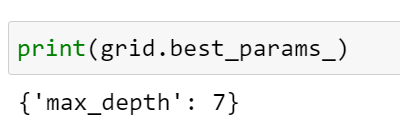


Obraz 3: Docelowy zbiór danych

#### Optymalizacja 1 parametru



Obraz 4: Optymalizacja modelu - maksymalna głębokość drzew decyzyjnych

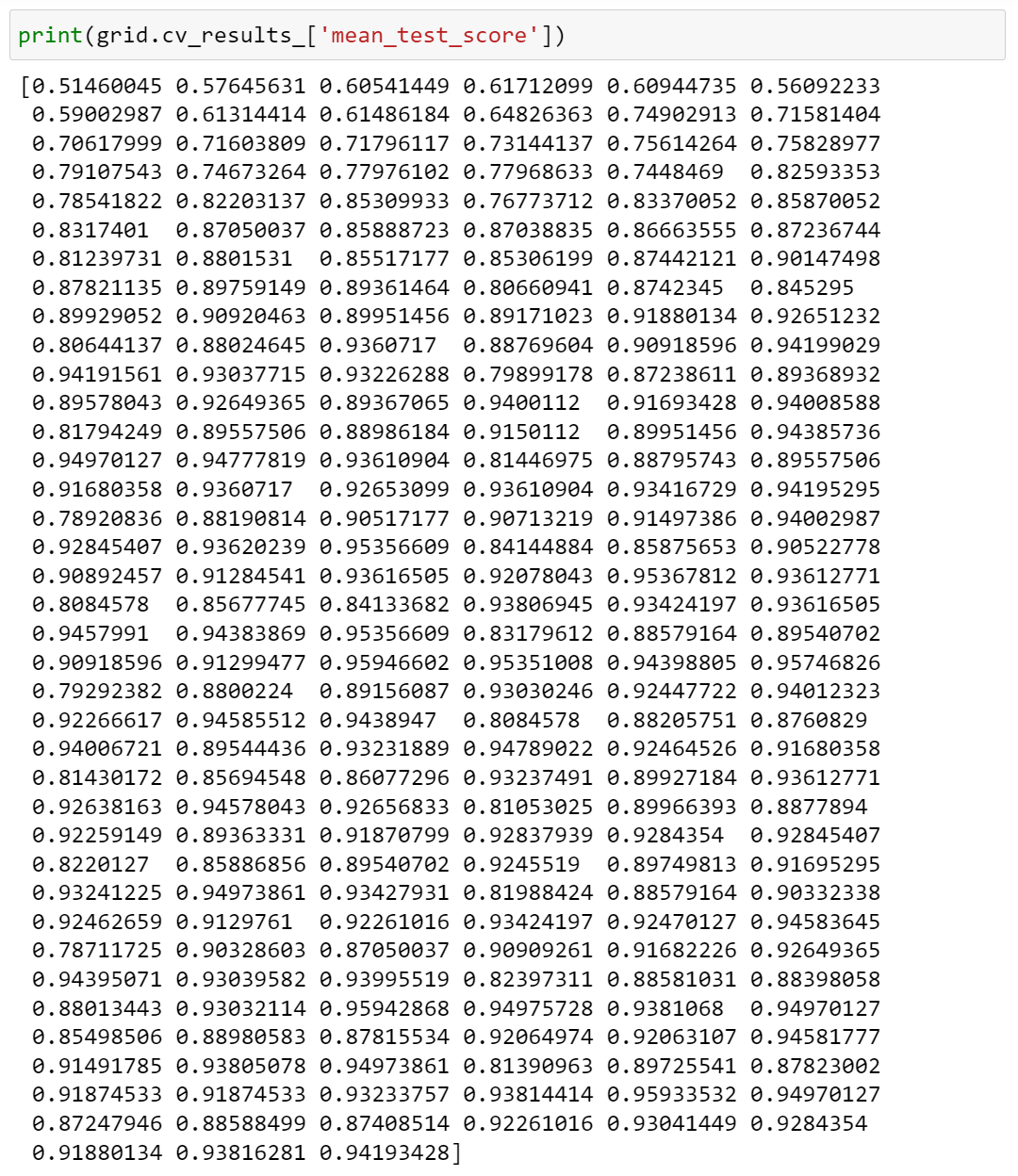


Obraz 5: Wynik optymalizacji modelu dla maksymalnej głębokości drzew decyzyjnych - najlepsze rozwiązanie

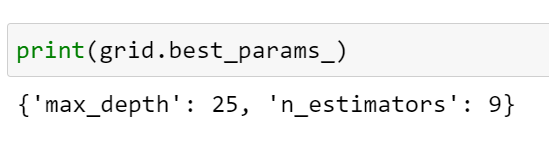
#### Optymalizacja 2 parametrów



Obraz 6: Optymalizacja modelu - maksymalna głębokość drzew decyzyjnych oraz ilość drzew decyzyjnych

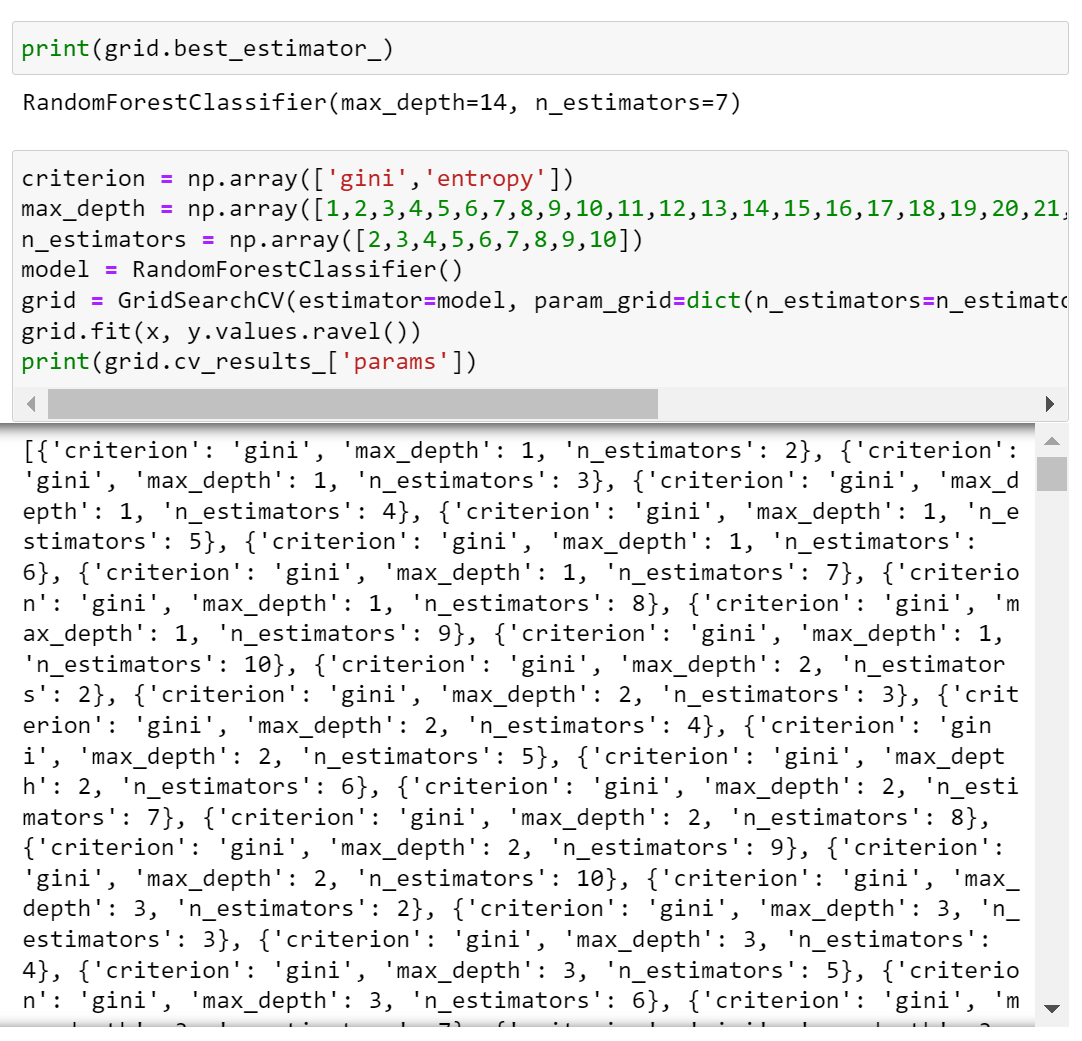


Obraz 7: Wyświetlenie średnich dopasowań modelu dla wszystkich kombinacji

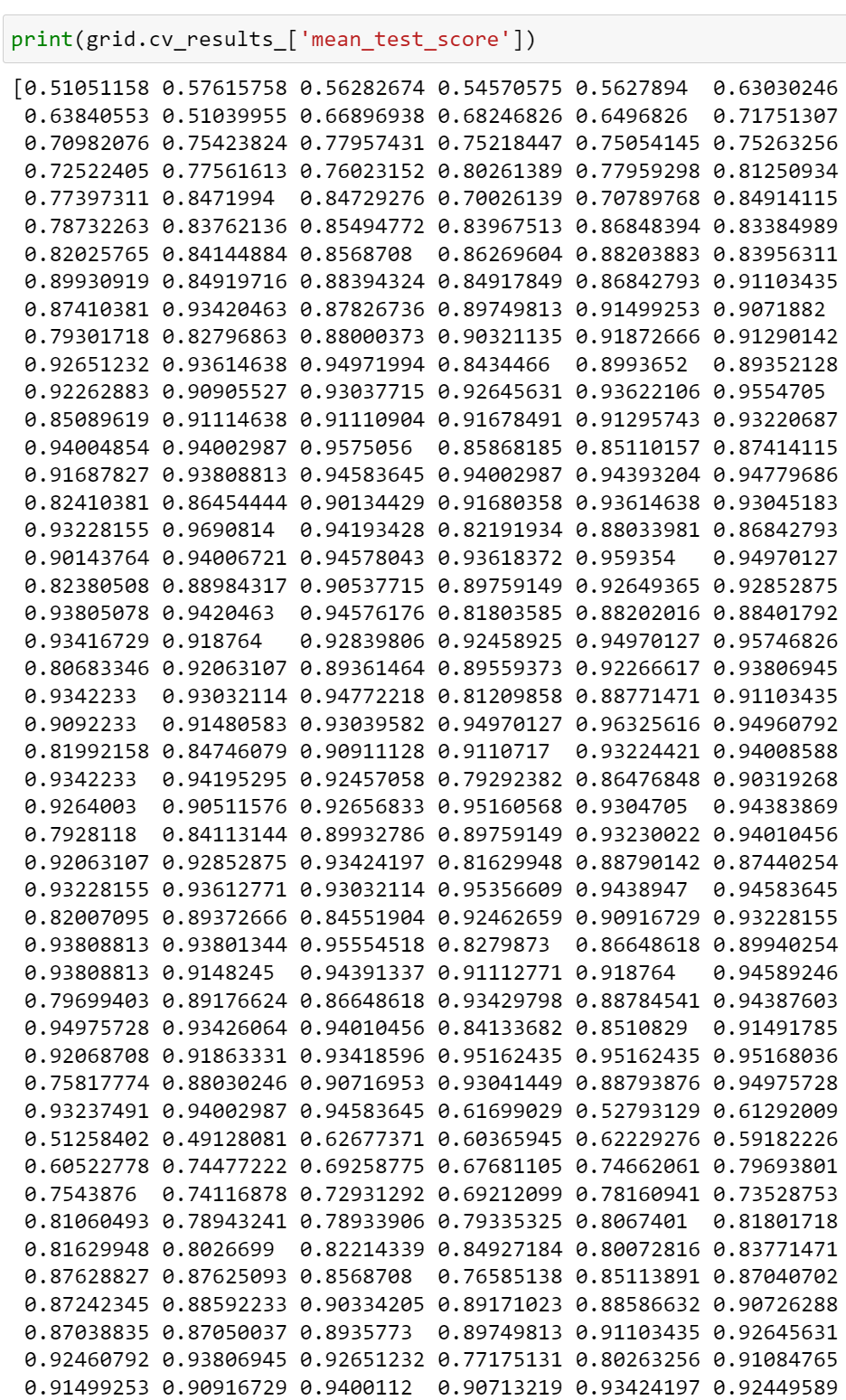


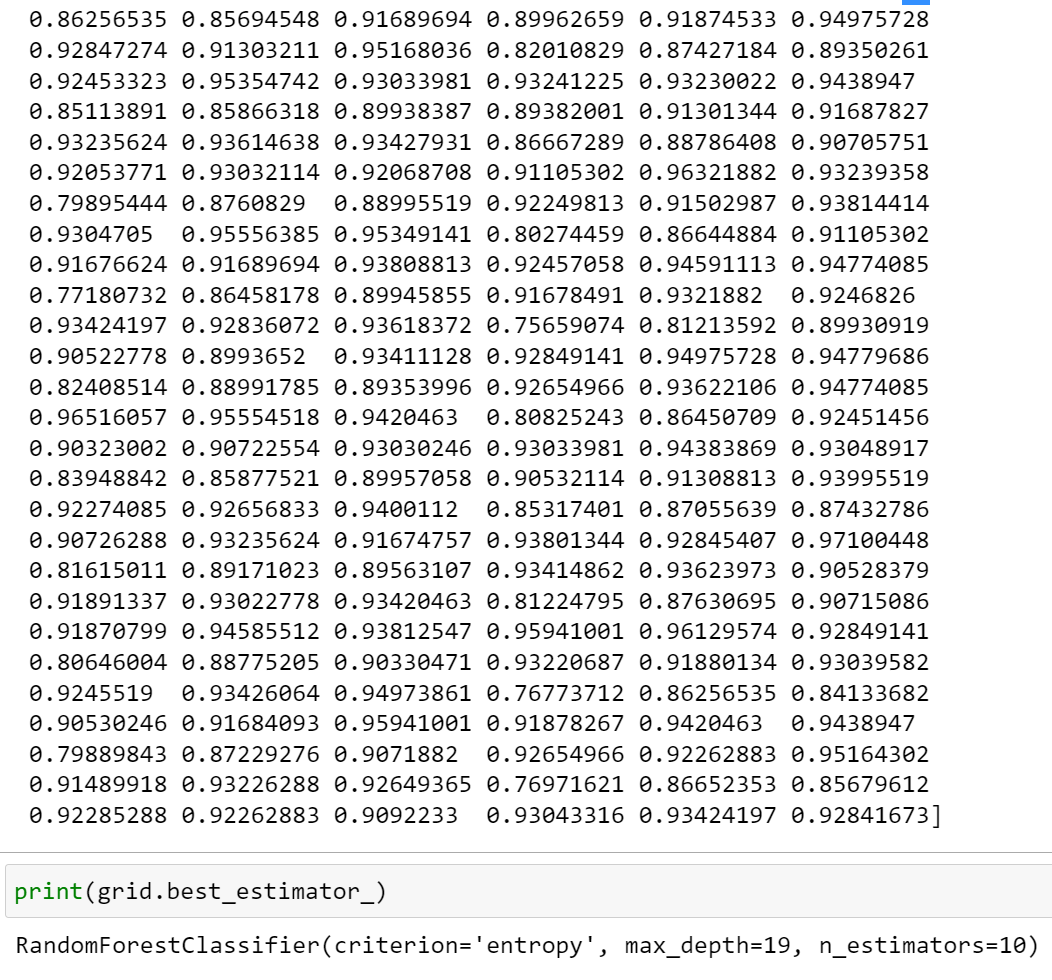
Obraz 8: Wynik optymalizacji modelu dla maksymalnej głębokości drzew decyzyjnych oraz ilości drzew   
- najlepsze rozwiązanie

#### Optymalizacja 3 parametrów

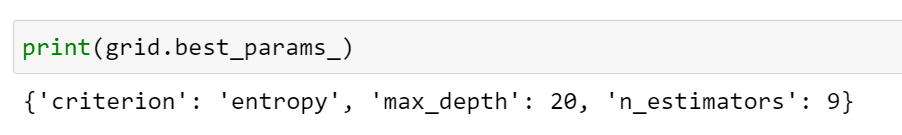


Obraz 9: Optymalizacja modelu - maksymalna głębokość drzew decyzyjnych, ilość drzew oraz sposób sprawdzania dopasowania podczas podziału w drzewach decyzyjnych





Obraz 10: Średnie wyniki dla wszystkich kombinacji podczas optymalizacji modelu



Obraz 11: Wynik optymalizacji modelu dla maksymalnej głębokość drzew decyzyjnych, ilości drzew oraz sposobu sprawdzania dopasowania podczas podziału w drzewach decyzyjnych - najlepsze rozwiązanie

#### Najlepsze parametry dla wybranego modelu - zestawienie wyników optymalizacji modelu

Tabela 1: Zestawienie najlepszych parametrów w zależności od rodzaju optymalizacji modelu

|  |  |  |  |
| --- | --- | --- | --- |
| **Parametry optymalizacji** | **Najelpszy wynik: max\_depth** | **Najelpszy wynik: n\_estimators** | **Najlepszy wynik: criterion** |
| *max\_depth* | 7 | *default (2)* | *default („gini”)* |
| *max\_depth, n\_estimators* | 25 | 9 | *default („gini”)* |
| *max\_depth, n\_estimators, criterion* | 20 | 9 | *„entropy”* |

Dla najprostszego modelu wystarczy las drzew decyzyjnych z dwoma drzewami o maksymalnej głębokości równej 7. Dla najbardziej dokładnego modelu potrzeba las drzew decyzyjnych z 9 drzewami o głębokości maksymalnej równej 20 oraz z kryterium dopasowywania ustawionym na „entropy”.

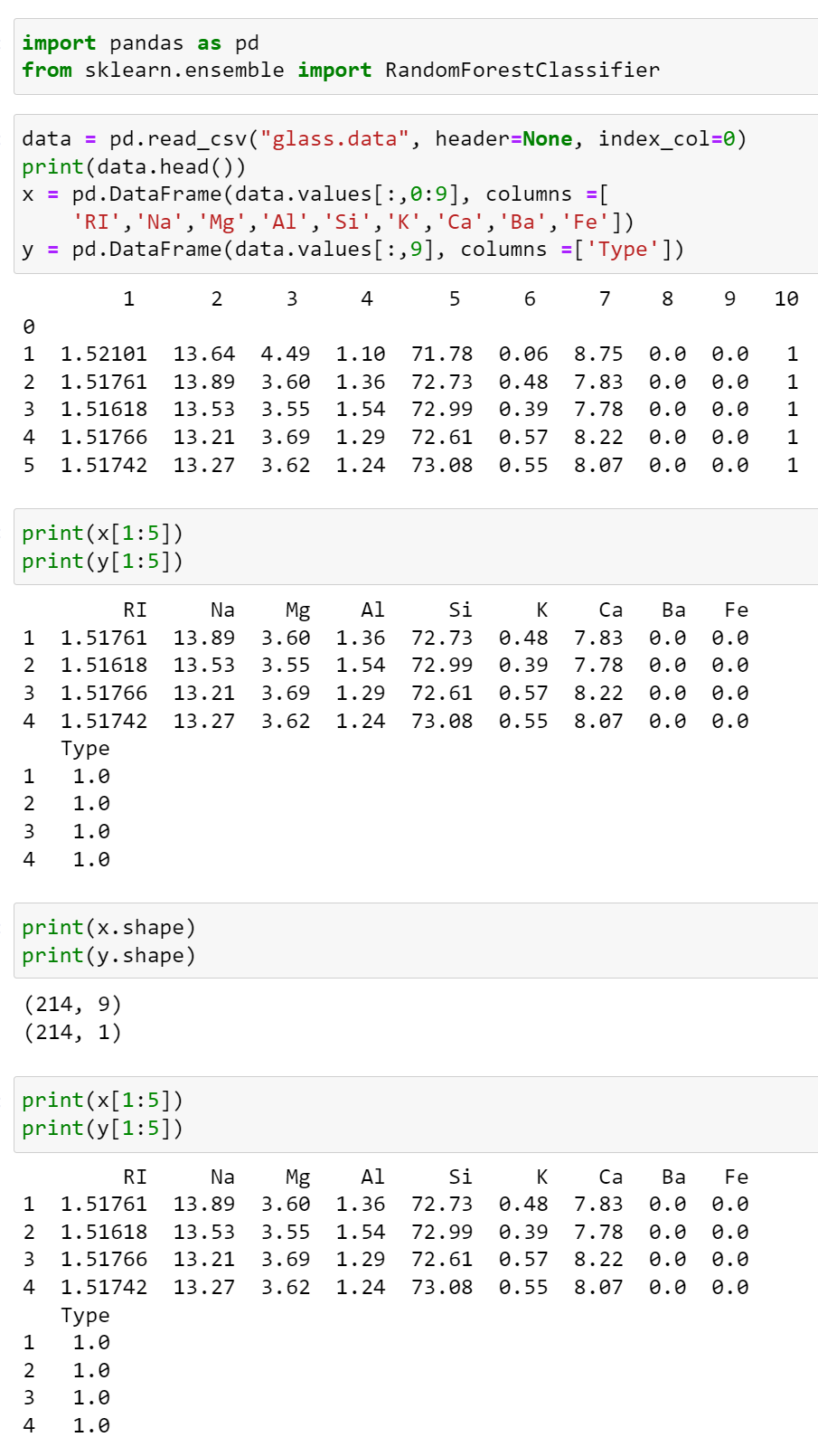
# Przetrenowanie

## Zadanie #2

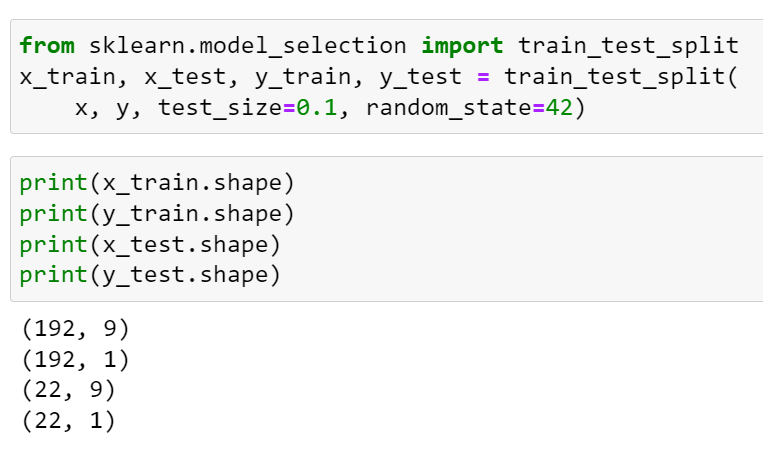
### Treść polecenia

Proszę zaprezentować zjawisko overfitingu na klasyfikatorze random forest.

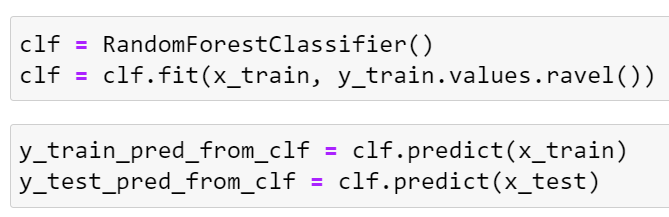
### Rozwiązanie



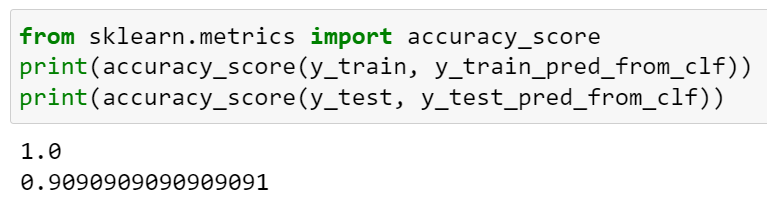
Obraz 12: Ładowanie bazy danych



Obraz 13: Podział bazy danych na zbiór trenujący i zbiór testujący



Obraz 14: Trenowanie modelu na klasyfikatorze random forest



Obraz 15: Wynik dokładności modelu na zbiorze trenującym i na zbiorze testującym

### Wnioski

Bardzo dokładne trenowanie modelu nie jest wskazane, ponieważ może prowadzić do przetrenowania - model będzie się idealnie sprawdzał na danych ze zbioru trenującego, jednak będzie niewystarczająco skuteczny na danych ze zbioru testującego. Lepszym rozwiązaniem jest mniejsza dokładność na modelu trenującym, aby dokładność na danych testujących była zbliżona.

# Analiza tekstu

## Zadanie #3

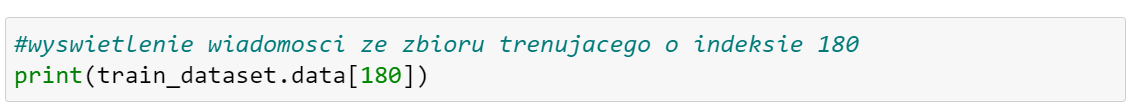
### Treść polecenia

Proszę powtórzyć ćwiczenia dla innego podzbioru kategorii, np. rec.autos, rec.motorcycles, rec.sport.baseball, rec.sport.hockey.

### Rozwiązanie



Obraz 16: Załadowanie tekstów wiadomości e-mail



Obraz 17: Wyświetlenie wiadomości o indeksie 180

*From: ajjb@adam4.bnsc.rl.ac.uk (Andrew Broderick)*

*Subject: Re: Solar Sail Data*

*Keywords: Solar Sail*

*Organization: Rutherford Appleton Lab, UK*

*Lines: 79*

*In article <1993Apr15.051746.29848@news.duc.auburn.edu> snydefj@eng.auburn.edu writes:*

*>*

*>I am looking for any information concerning projects involving Solar*

*> Sails*

*I was at an interesting seminar at work (UK's R.A.L. Space Science*

*Dept.) on this subject, specifically on a small-scale Solar Sail*

*proposed as a student space project. The guy giving the talk was keen to*

*generate interest in the project. I'll typein the handout he gave out at*

*the meeting. Here goes :*

*The Microlight Solar Sail*

*-------------------------*

*1. Introduction*

*The solar sail is a well-established concept. Harnessing the pressure of*

*sunlight, a spacecraft would have unlimited range. In principle, such a*

*vehicle could explore the whole Solar System with zero fuel consumption.*

*However it is more difficult to design a practical solar sail than most*

*people realize. The pressure of sunlight is only about one kilogram per*

*square kilometer. Deploying and controlling the large area of aluminized*

*fabric which would be necessary to transport a 'conventional' type*

*spacecraft is a daunting task. This is why, despite the potential of hte*

*idea, no such craft has actually been launched to date.*

*2.Design*

*Recent advances in microelectronics make possible a different concept: a*

*tiny sail just a few metres in diameter which could be controlled purely*

*be electronics, with no mechanical parts. Several attitude control*

*methods are feasible: for example the pressure sunlight exerts on a*

*panel of solar cells varies according to whether power is being drawn.*

*The key components of the craft will be a minute CCD camera developed at*

*Edinburgh University which can act as both attitude sensor and data*

*gathering device; solar cells providing ~1 watt power for control and*

*communication; and a directional radio antenna etched onto the surface*

*of the sail itself. Launched as a piggyback payload, the total cost of*

*the mission can be limited to a few tens of thousands of dollars.*

*3.Missions*

*The craft would be capable of some ambitious missions. For example:*

*a) It could rendezvous with a nearby asteroid from the Apollo or Amor*

*groups. Closeup pictures could be transmitted back to Earth at a low bit*

*rate.*

*b) It could be steered into a lunar polar orbit. Previously unobserved*

*areas around the lunar poles could be viewed. By angling the sail to*

*reflect sunlight downwards, polar craters whose bases never receive*

*sunlight could be imaged. Bright reflections would confirm that*

*volatiles such as water ice have become trapped in these*

*locations.[Immensely valuable information for setting up a manned lunar*

*base, BTW]*

*c) It could be sent to rendezvous with a small asteroid or comet*

*nucleus. Impacting at low speed, a thin wire probe attached to the craft*

*causes it to rebound while capturing a tiny sample is a sharp-edged*

*tube, like performing a biopsy. Returning to Earth, the sail acts as an*

*ideal re-entry parachute: load per unit area 20 gm/m2 ensures that heat*

*is reradiated so efectively that the sail temperature cannot exceed ~300*

*deg C. The material sample is recovered, enclosed in a small insulating*

*container.*

*Contact: Colin Jack Tel. 0865-200447*

*Oxford Mathematical Designs, 131 High Street, Oxford OX1 4DH, England*

*--------------------------------*

*This guy would love to hear from anyone interested in this project or*

*seeking details or anything, and would be most happy to send you more*

*information.*

*Andy*

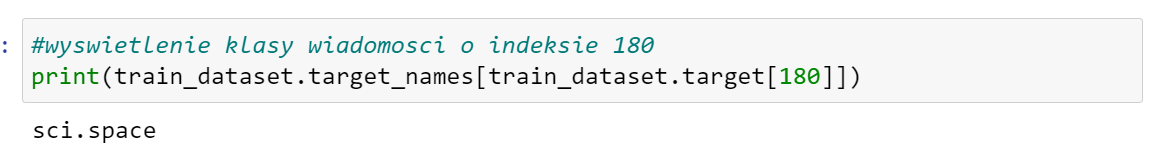
*--*

*-----------------------------------*

*Andy Jonathan J. Broderick, | "I have come that they might have |*

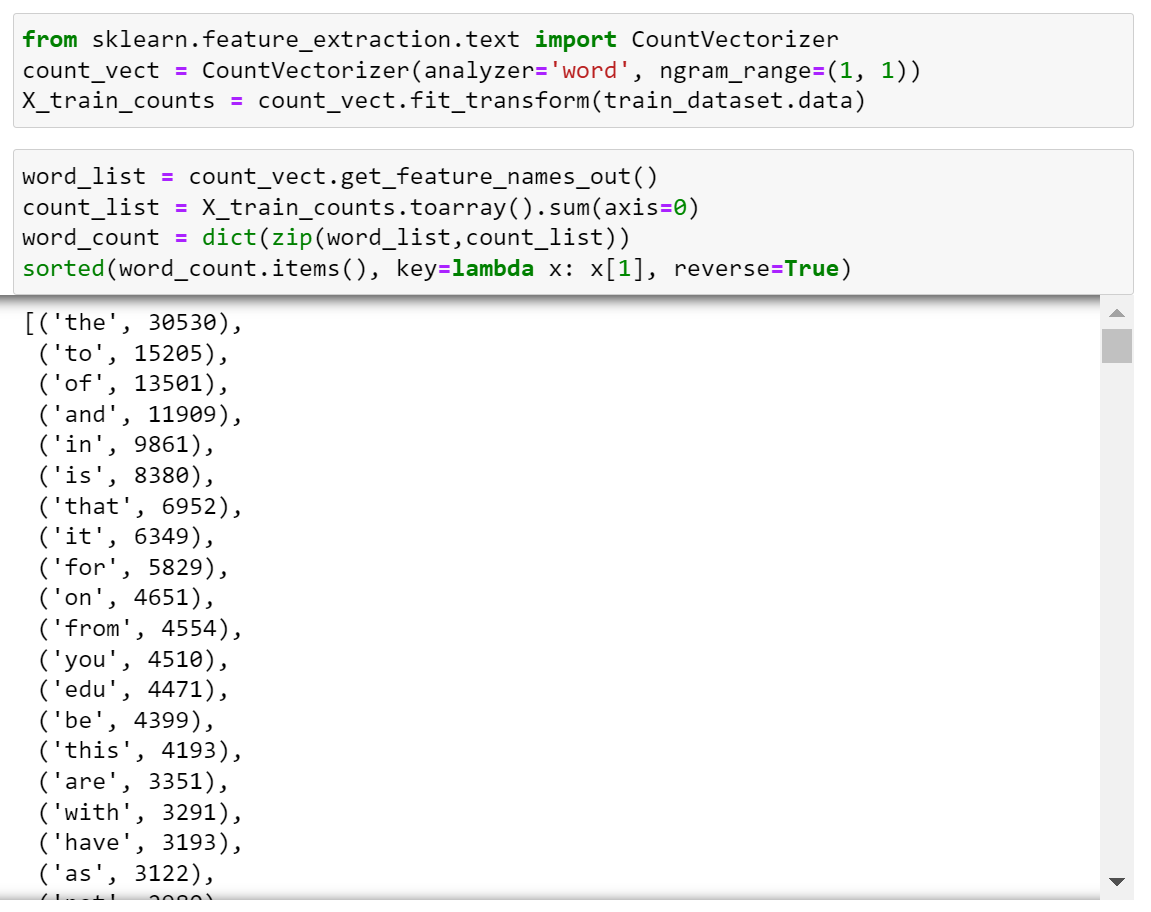
*Rutherford Lab., UK | life, and have it to the full" |*

*Mail : ajjb@adam2.bnsc.rl.ac.uk | - Jesus Christ |*

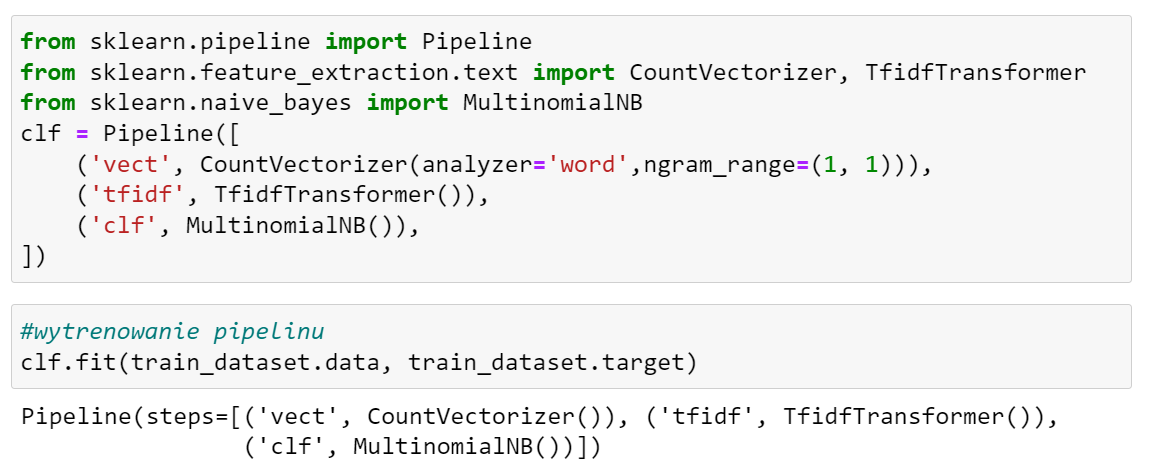


Obraz 18: Dopasowana kategoria wiadomości o indeksie 180

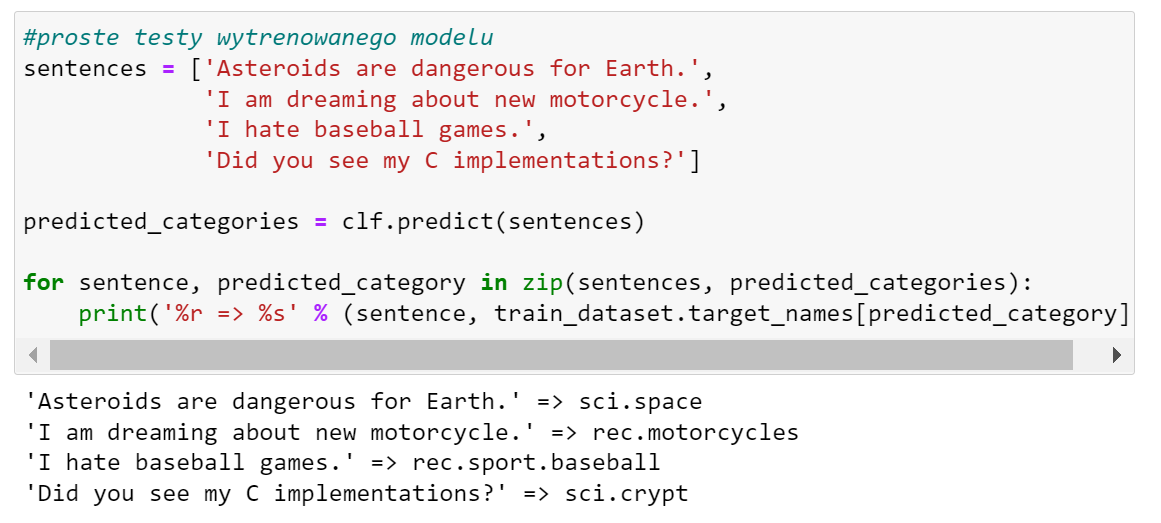
Wybrana wiadomość została przydzielona do kategorii „sci.space”, ponieważ zawiera słowa kluczowe typu „science”, „space” czy „Earth”.



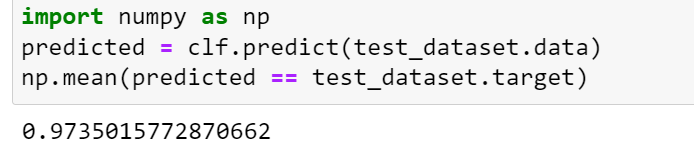
Obraz 19: Wyświetlenie najczęściej występujących słów we wszystkich wiadomościach



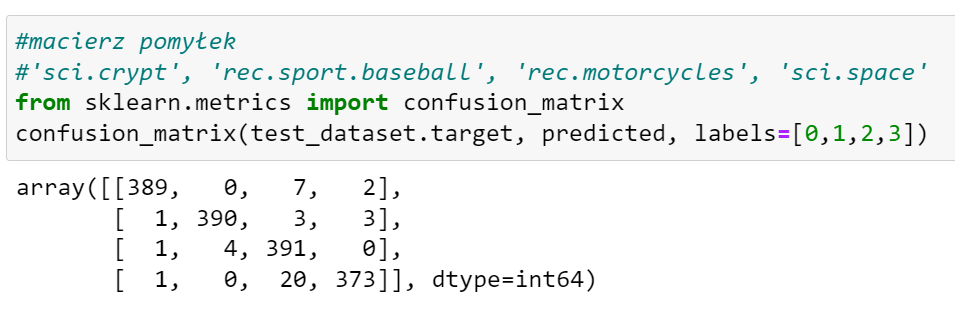
Obraz 20: Trenowanie modelu za pomocą naiwnego klasyfikatora bayesowskiego



Obraz 21: Proste testy wytrenowanego modelu



Obraz 22: Dokładność wytrenowanego modelu



Obraz 23: Macierz pomyłek dla wytrenowanego modelu

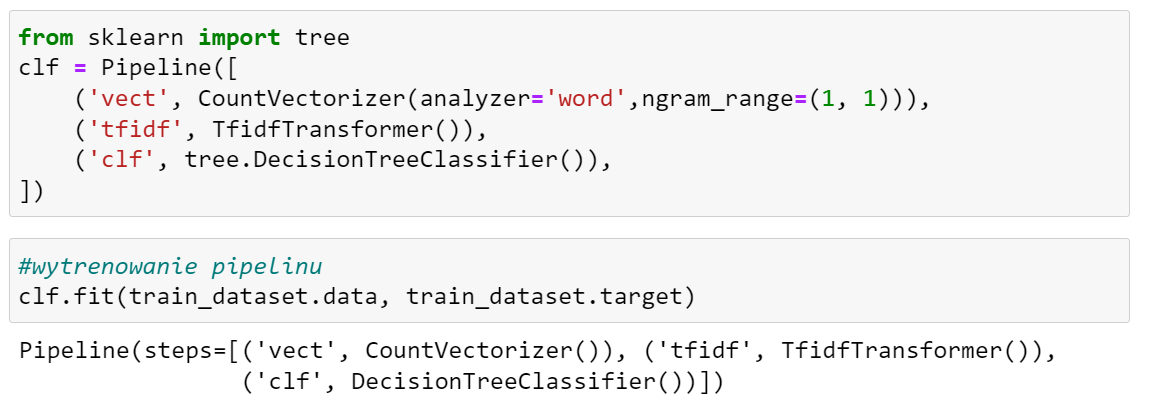
Na podstawie macierzy pomyłek widzimy, że model bardzo dobrze radzi sobie z dopasowywaniem kategorii do wiadomości; najbardziej myli „sci.crypt” z „rec.motorcycles” oraz „sci.space” z „rec.motorcycles”. Kategoria „rec.motorcycles” jest kategorią, którą najczęściej model błędnie wybiera.

## Zadanie #4

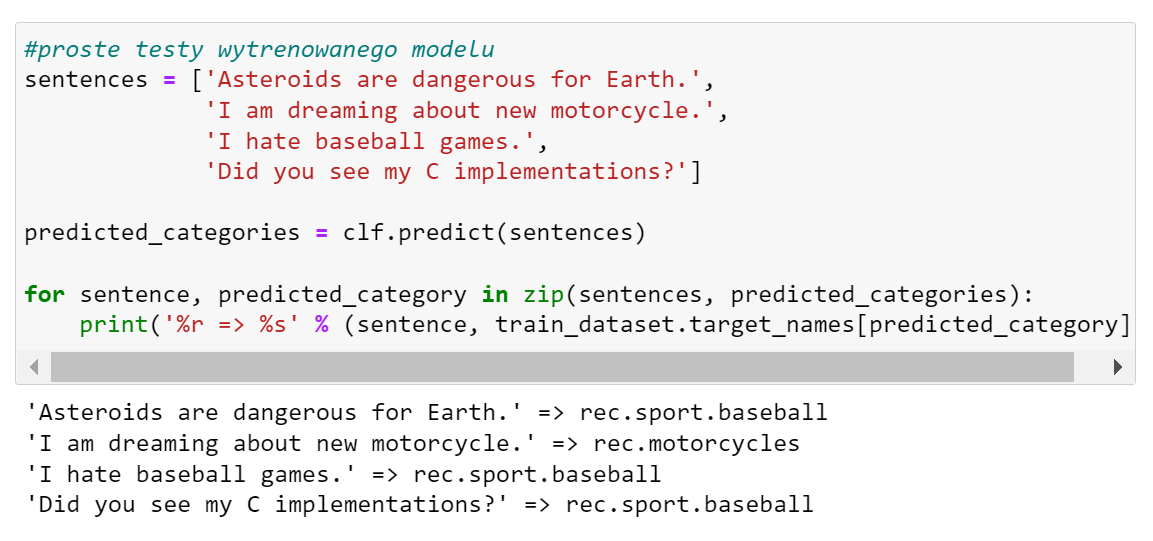
### Treść polecenia

Proszę zmienić użyty klasyfikator na drzewo decyzyjne i zbadać jakość wyników klasyfikacji.

### Rozwiązanie

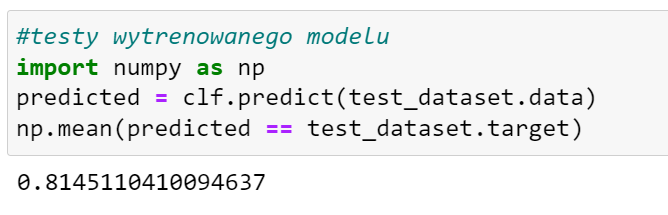


Obraz 24: Trenowanie pipelinu z klasyfikatorem drzewa decyzyjnego - porównywanie na raz jednego słowa z jednym słowem



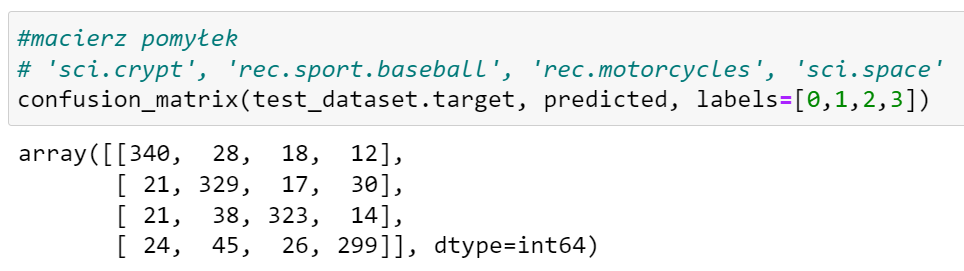
Obraz 25: Proste testy wytrenowanego modelu

Warto zauważyć, że przy zmianie klasyfikatora na drzewo decyzyjne model inaczej przypisuje kategorie już dla prostych testów wytrenowanego modelu - dwie z czterech wiadomości błędnie przypisuje do „rec.sport.baseball”.



Obraz 26: Dokładność wytrenowanego modelu po zmianie klasyfikatora

Wyraźnie spada też dokładność wytrenowanego modelu.



Obraz 27: Macierz pomyłek dla wytrenowanego modelu po zmianie klasyfikatora

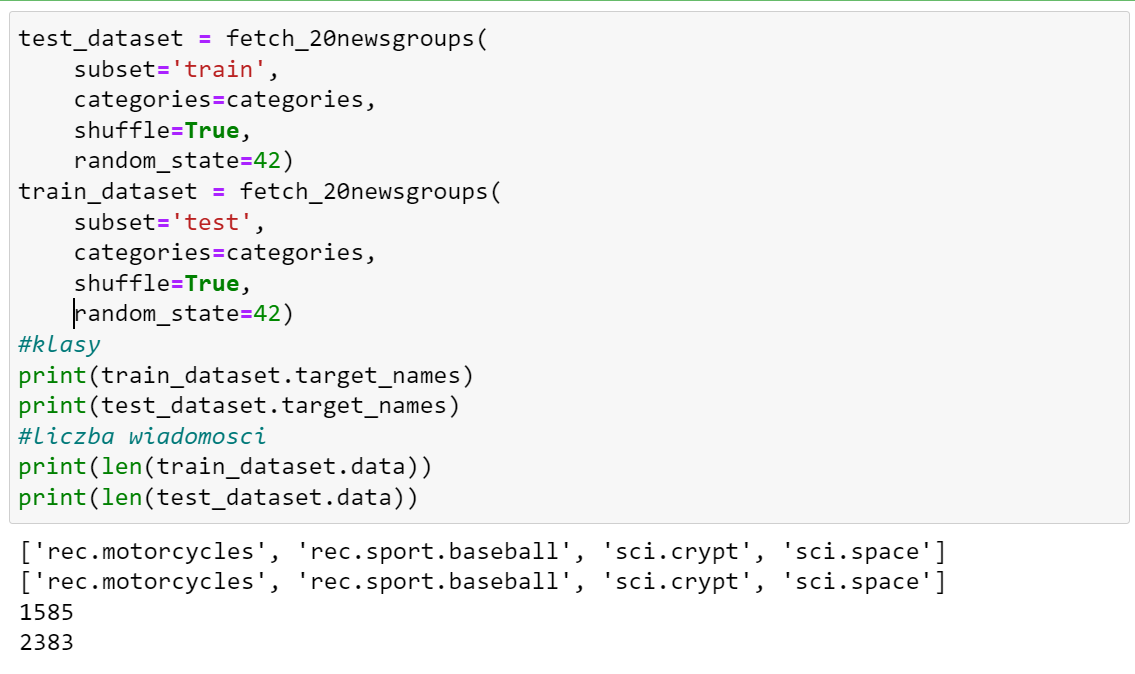
Macierz pomyłek również wyraźnie się różni - model dużo częściej błędnie przypisuje kategorie - najczęściej myli się w wiadomościach z kategorii „sci.space”, a najczęściej błędnie przypisywaną kategorią jest „rec.sport.baseball”.

## Zadanie #5

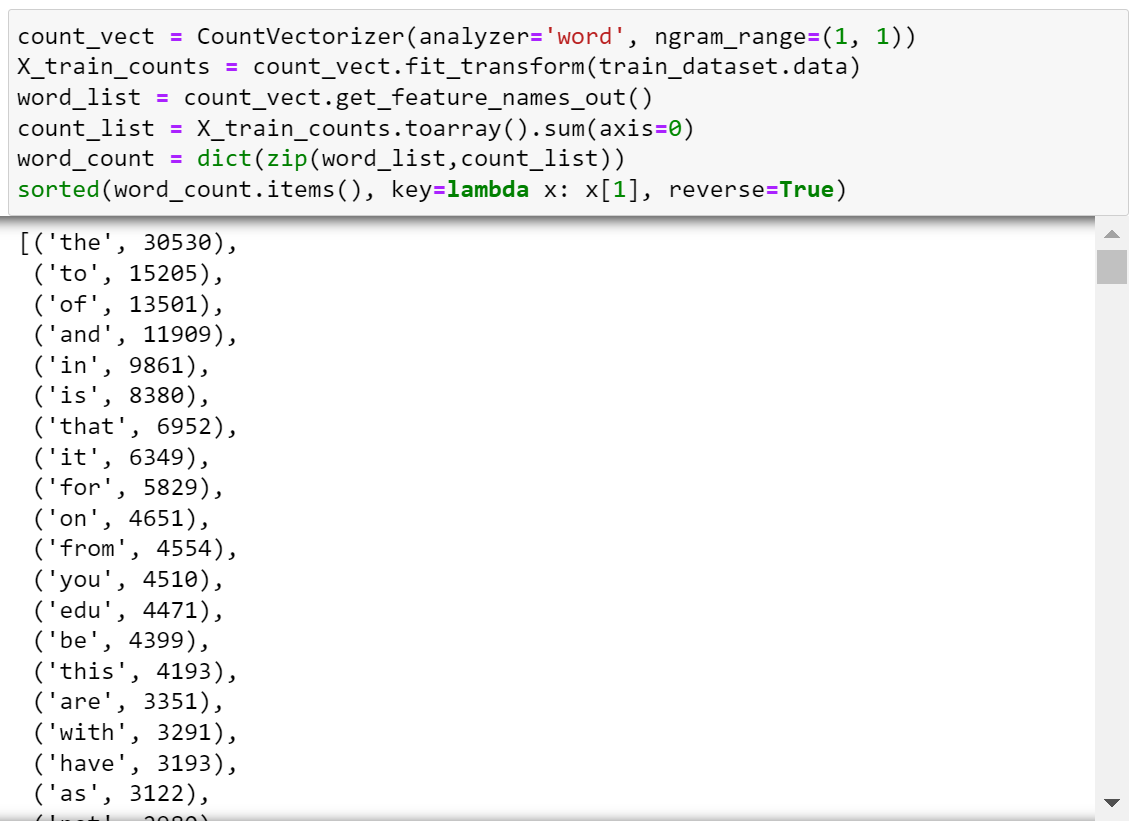
### Treść polecenia

Proszę wytrenować klasyfikator na zbiorze testującym a przetestować na zbiorze trenującym.

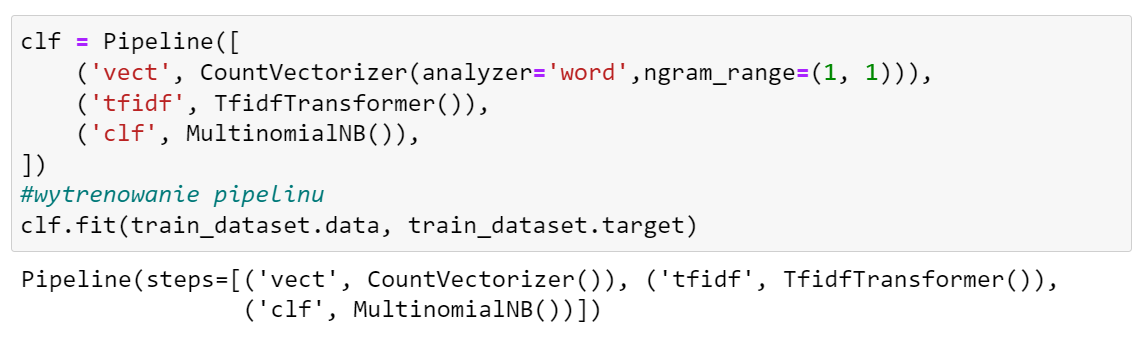
### Rozwiązanie



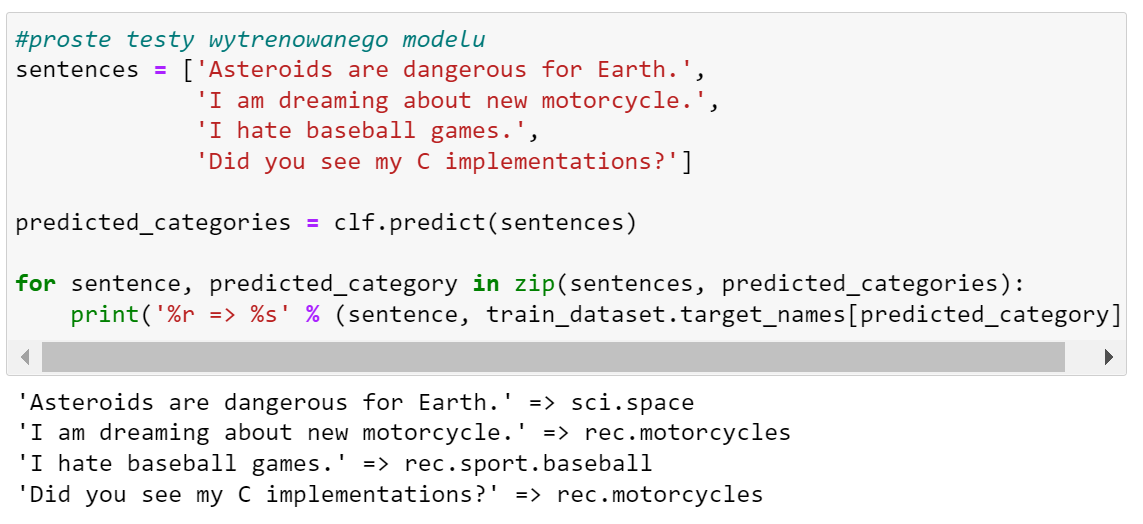
Obraz 28: Trenowanie modelu na zbiorze testującym a testowanie na zbiorze trenującym



Obraz 29: Najczęściej pojawiające się słowa we wszystkich wiadomościach

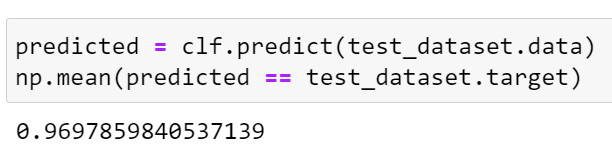


Obraz 30: Trenowanie pipelinu - porównywanie na raz jednego słowa z jednym słowem



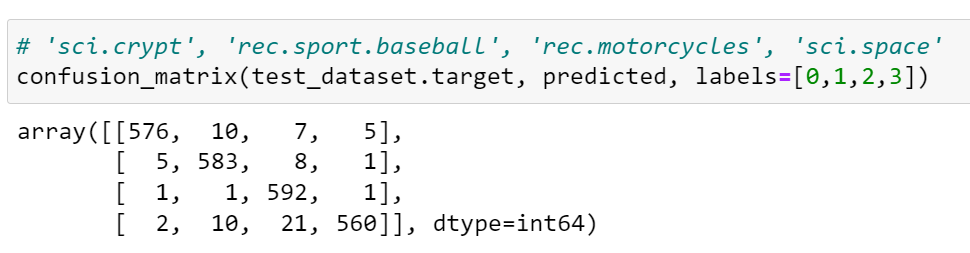
Obraz 31: Proste testy wytrenowanego modelu

Model przypisał błędnie tylko 1 wiadomość w prostych testach - zamiast „sci.crypt” oznaczył kategorię „rec.motorcycles”.



Obraz 32: Dokładność wytrenowanego modelu

Model mimo wszystko posiada wysoką dokładność predykcji.



Obraz 33: Macierz pomyłek wytrenowanego modelu

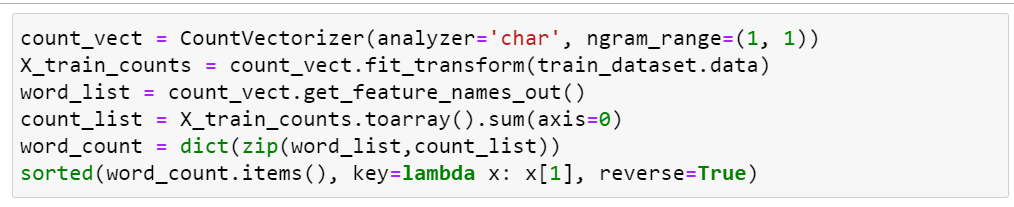
Model najczęściej myli „sci.space” z „rec.motorcycles” oraz „sci.crypt” z „rec.sport.baseball”. Kategoria „rec.motorcycles” jest najczęściej błędnie przypisywana.

## Zadanie #6

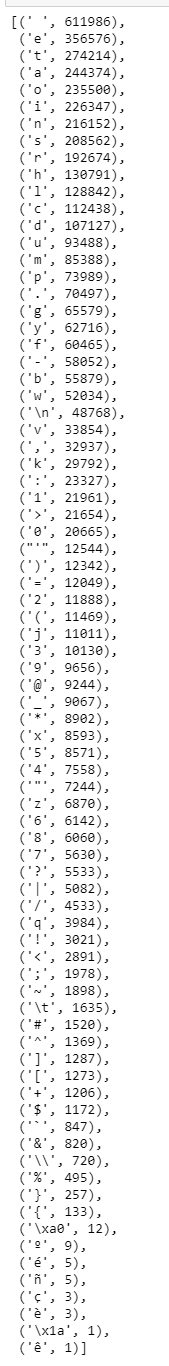
### Treść polecenia

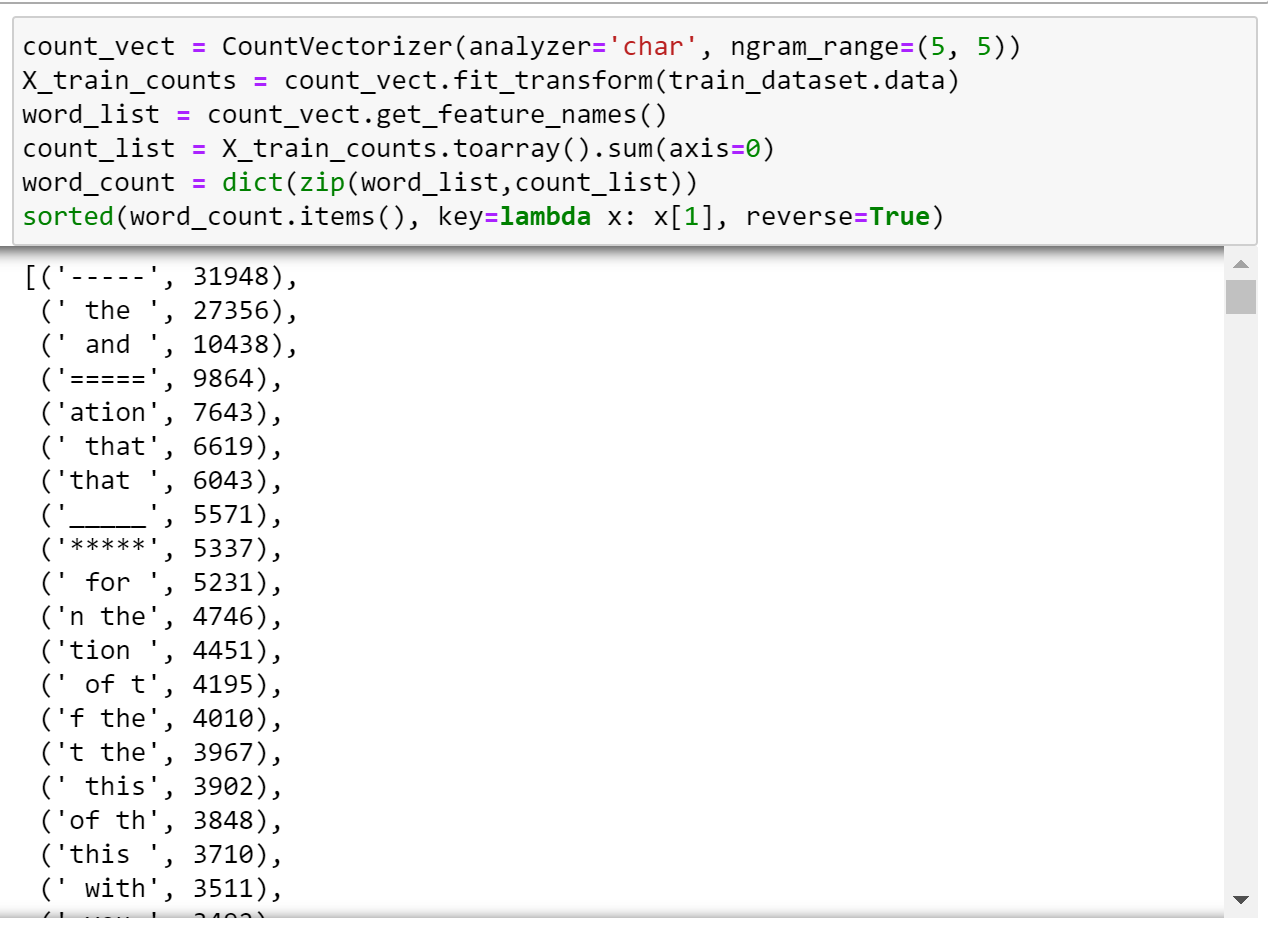
Proszę ustawić w odpowiednim miejscu parametr analyzer='char' i sprawdzić jak zmieniają się wyniki.

### Rozwiązanie

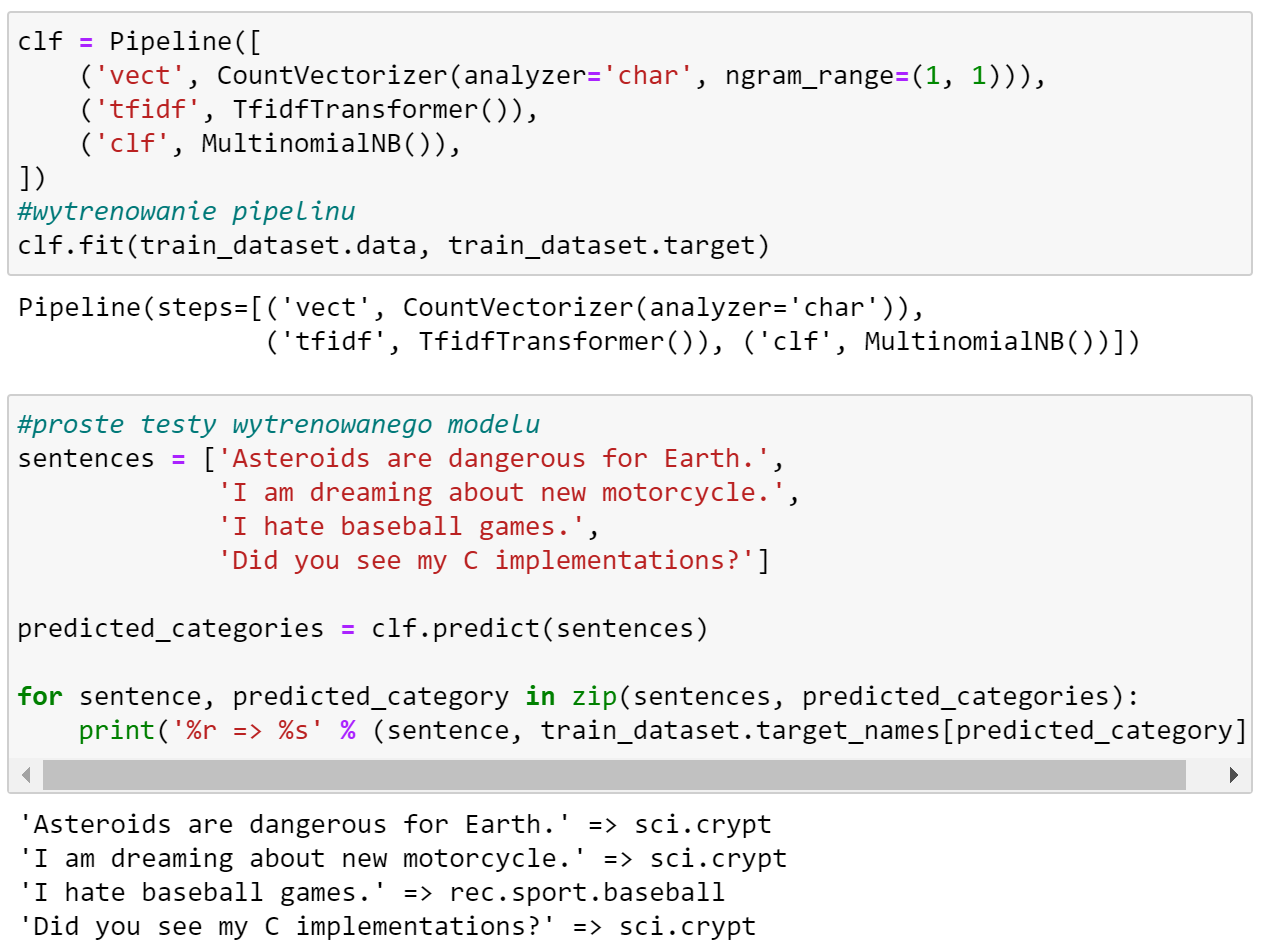


Obraz 34: Najczęściej pojawiające się znaki we wszystkich wiadomościach



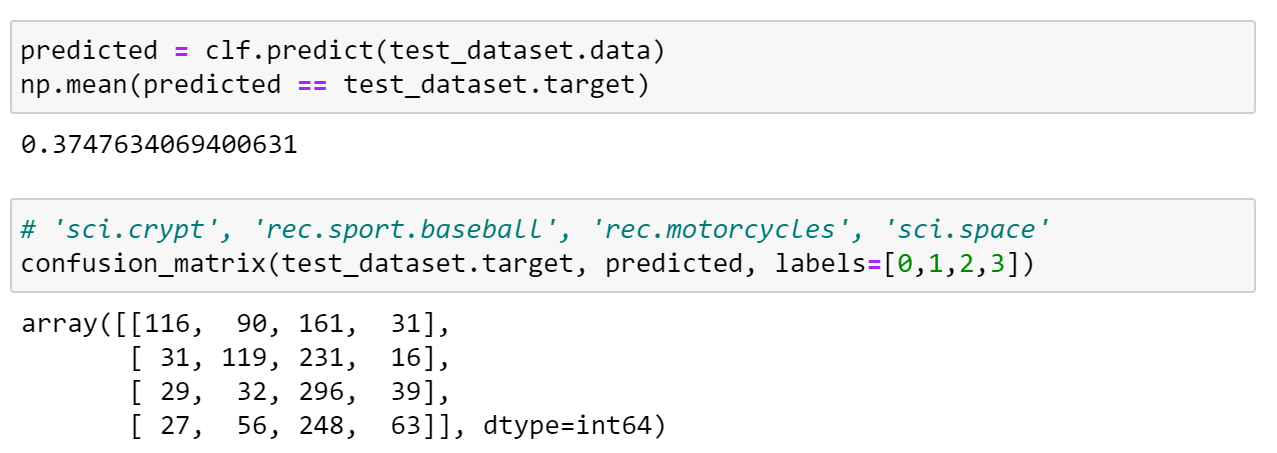


Obraz 35: Najczęściej pojawiające się 5 znakowe ciągi we wszystkich wiadomościach



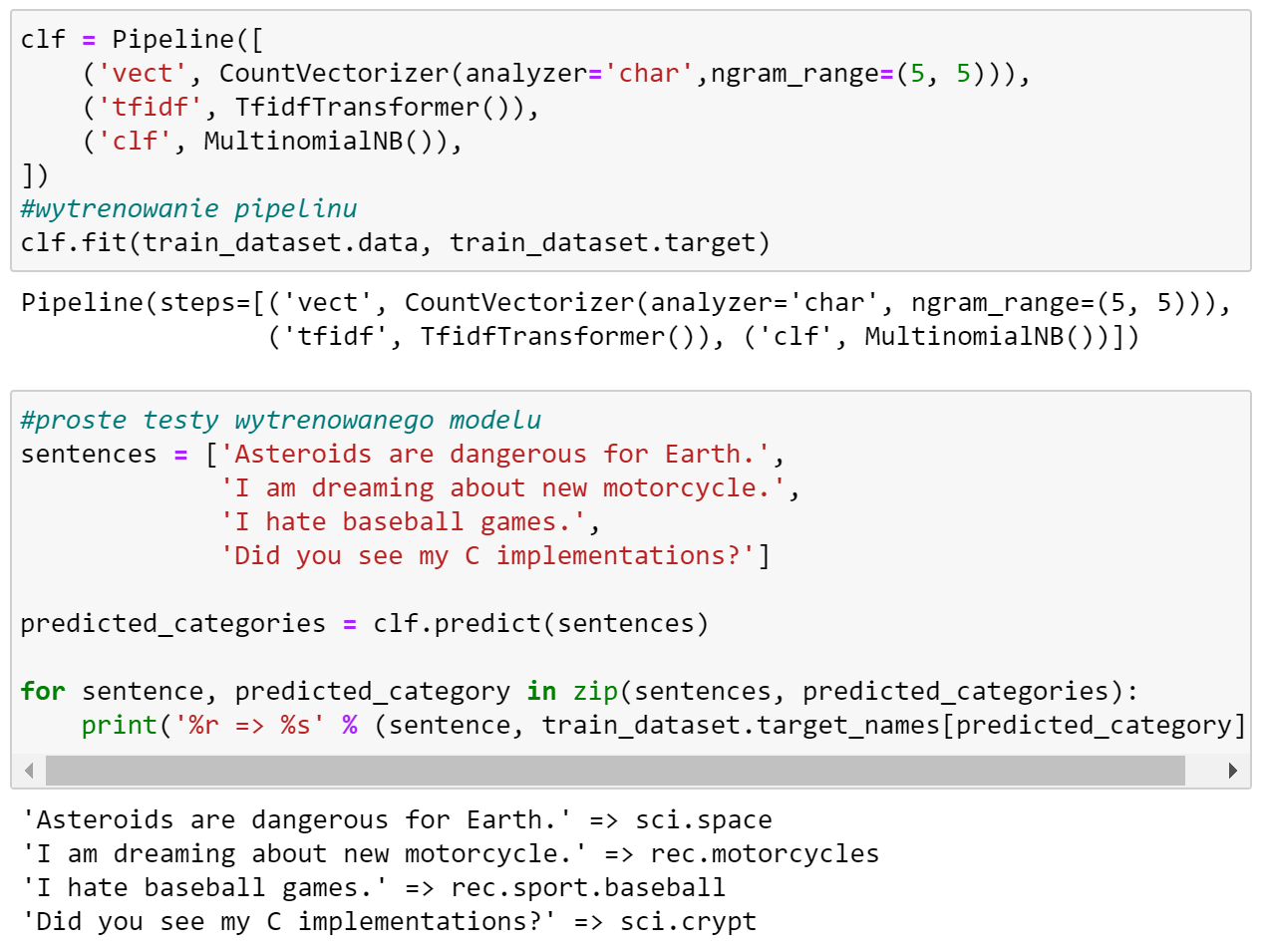
Obraz 36: Wytrenowanie pipelinu dla porównania pojedynczych znaków

Przy porównaniu pojedynczych znaków model w prostych testach przypisał prawie wszystkie wiadomości do kategorii „sci.crypt”.



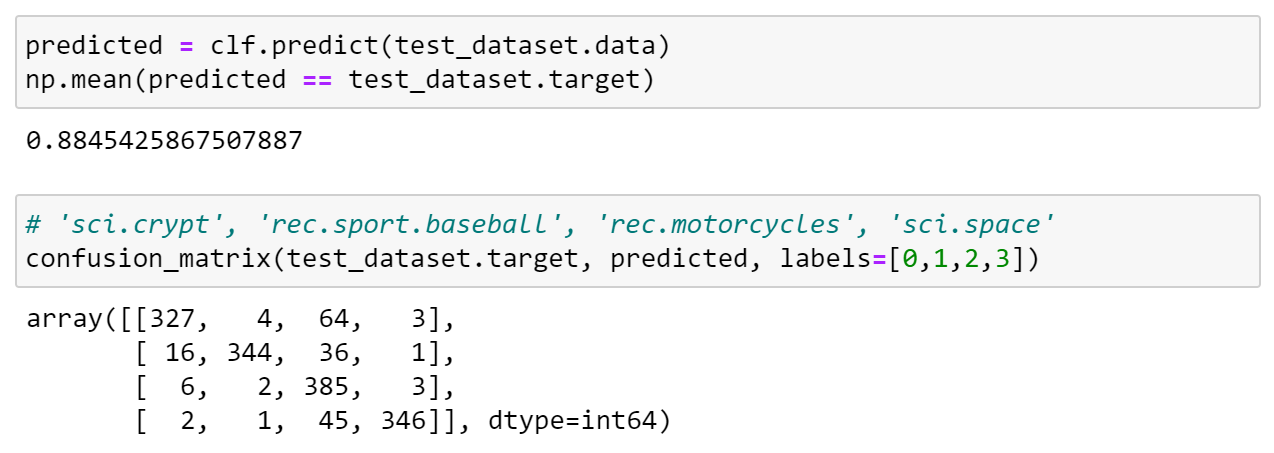
Obraz 37: Dokładność i macierz pomyłek dla modelu porównywającego pojedyncze znaki

Dokładność modelu jest bardzo niska. Model najczeście myli „sci.crypt” z „rec.motorcycles” a kategorię „sci.space” prawie zawsze oznacza jako „rec.motorcycles”.



Obraz 38: Trenowanie pipelinu w modelu porównującym 5 znakowe ciągi

Model przypisał poprawnie wszystkie wiadomości w prostych testach.



Obraz 39: Dokładność i macierz pomyłek dla modelu porównującego 5 znakowe ciągi

Model ma dość wysoką dokładność, jednak nie tak wysoką jak model porównujący ze sobą pojedyncze znaki. Najczęściej błędnie przypisuje „sci.crypt” jako „rec.motorcycles” oraz „sci.space” jako „rec.motorcycles”.