Moduł 11
Klasyfikacja przy użyciu BiG Data

iBigWorld:
Innovations for Big Data in a Real World

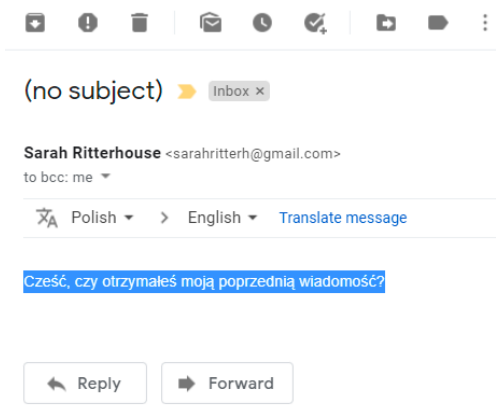
Zespół UBB

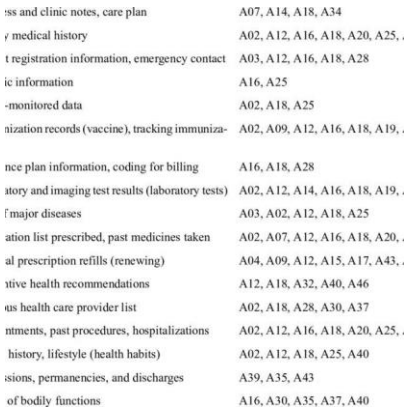
iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

Klasyfikacja w uczeniu maszynowym

- klasyfikacja jest uważana za metodę uczenia nadzorowanego
- rozważa również problem modelowania predykcyjnego, gdzie dla danego przykładu przewidziana jest etykieta klasy [1].
- matematycznie odwzorowuje funkcję (f) ze zmiennych wejściowych (X) na zmienne wyjściowe (Y) jako cel, etykietę lub kategorie.
- Klasyfikacja może być realizowana na podstawie danych ustrukturyzowanych lub nieustrukturyzowanych.
- **spam** **dane medyczne**





ss and clinic notes, care plan	A07, A14, A18, A34
medical history	A02, A12, A16, A18, A20, A25, .
t registration information, emergency contact	A03, A12, A16, A18, A28
ic information	A16, A25
-monitored data	A02, A18, A25
nization records (vaccine), tracking immuniza-	A02, A09, A12, A16, A18, A19, .
nce plan information, coding for billing	A16, A18, A28
atory and imaging test results (laboratory tests)	A02, A12, A14, A16, A18, A19, .
major diseases	A03, A02, A12, A18, A25
ation list prescribed, past medicines taken	A02, A07, A12, A16, A18, A20, .
al prescription refills (renewing)	A04, A09, A12, A15, A17, A43, .
ative health recommendations	A12, A18, A32, A40, A46
us health care provider list	A02, A18, A28, A30, A37
ntments, past procedures, hospitalizations	A02, A12, A16, A18, A20, A25, .
history, lifestyle (health habits)	A02, A12, A18, A25, A40
sions, permanencies, and discharges	A39, A35, A43
of bodily functions	A16, A30, A35, A37, A40

iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

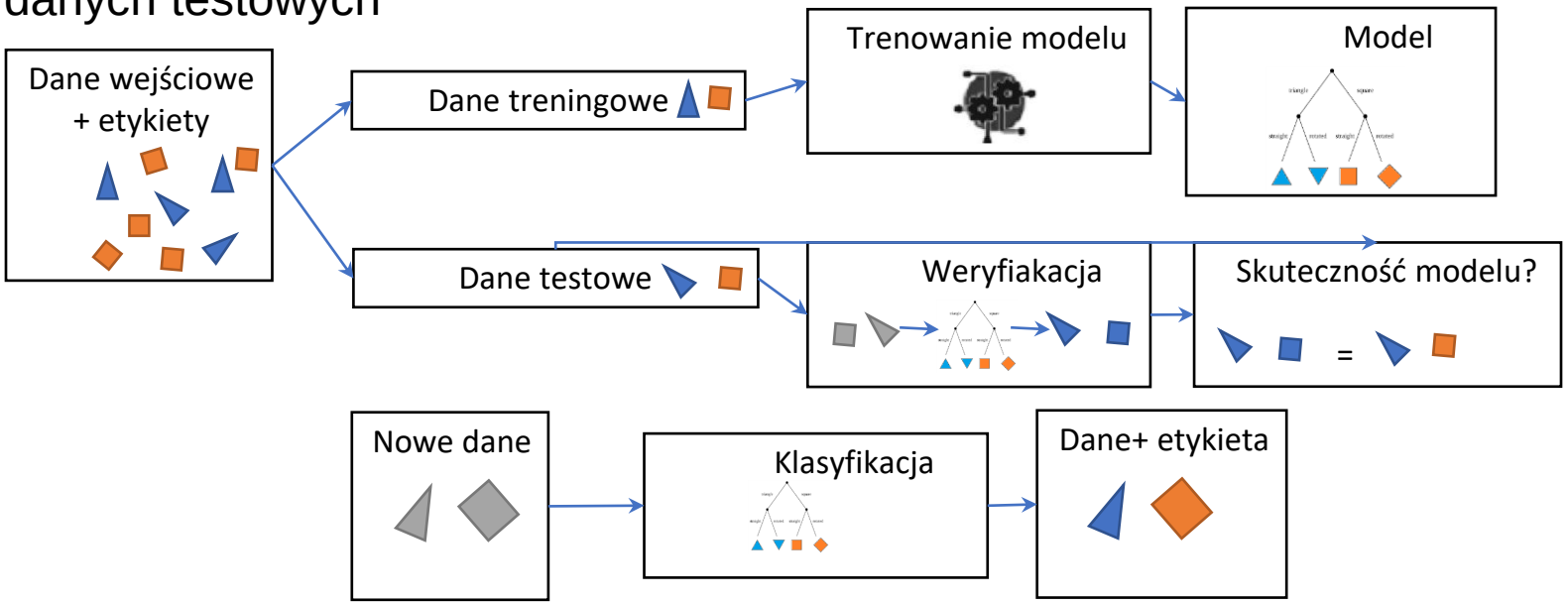
Problem klasyfikacji

- *Klasyfikacja binarna:* odnosi się do zadań klasyfikacyjnych mających dwie klasy, takie jak „prawda i fałsz” lub „tak i nie” [1]. Przykład: wiadomość e-mail „spam” i „nie spam” w powyższym przykładzie dostawców usług pocztowych jest traktowana jako klasyfikacja binarna.
- *Klasyfikacja wieloklasowa:* tradycyjnie dotyczy to zadań klasyfikacyjnych, które mają więcej niż dwie klasy. Przykładowe ataki sieciowe: DoS (atak odmowy usługi), U2R (atak użytkownika na root), R2L (atak z roota na lokalny) i atak sondujący.
- *Klasyfikacja z wieloma etykietami:* w uczeniu maszynowym klasyfikacja z wieloma etykietami jest ważnym czynnikiem, gdy klasyfikowany element jest powiązany z kilkoma klasami lub etykietami. Na przykład wiadomości Google mogą być prezentowane w kategoriach „nazwa miasta”, „technologia” lub „najnowsze wiadomości”.

Co-funded by the Erasmus+ Programme of the European Union

Klasyfikacja krok po kroku

- Klasyfikacja zwykle obejmuje dwa etapy. Poznanie relacji na podstawie danych wejściowych i wyjściowych, a następnie wykorzystanie tego modelu do nowych danych.
- Istnieje ryzyko wykorzystania nieprzetestowanego modelu, dlatego zanim model będzie mógł zostać zastosowany, zazwyczaj jest on weryfikowany na danych testowych



iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the
Erasmus+ Programme
of the European Union

Klasyfikacja w Big Data

- opracowanie skalowalnych algorytmów ML, które są w stanie obsłużyć duże zbiory danych
- Dwa podejścia [2]:
  - równoległość danych: wykorzystanie istniejącej architektury big data, partycjonowanie danych wejściowych w pionie, poziomie, a nawet arbitralnie na łatwe do zarządzania fragmenty, a następnie jednoczesne przetwarzanie na wszystkich podzbiorach,
  - Równoległość parametrów modelu: tworzenie zrównoleglonych wersji algorytmów uczenia się, najpierw dzieląc uczący się model/parametry, a następnie obliczając jednocześnie na każdym bloku strukturalnym.
- Spark MLlib i Mahout to dwa reprezentatywne projekty/pakiety typu open source, które obsługują wiele skalowalnych algorytmów uczenia się.
- MapReduce może zaimplementować wiele algorytmów ML, w tym regresję liniową, kmeans, regresję logistyczną, Naive Bayes, SVM, ICA, PCA, EM, sieć neuronową itp.

Co-funded by the Erasmus+ Programme of the European Union

Algorytmy klasyfikacji w Spark MLlib

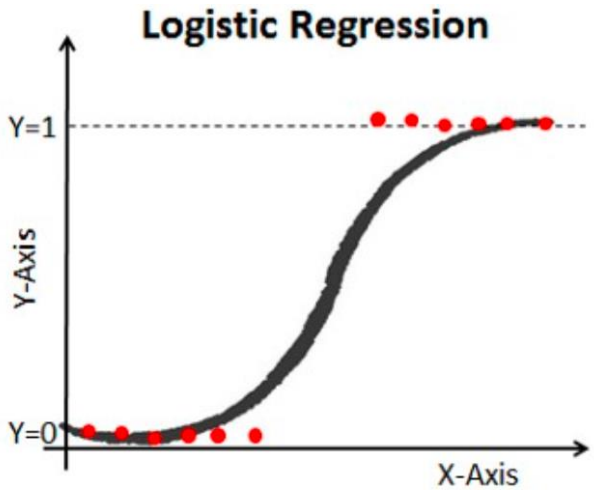
- **Regresja logistyczna**
- **Drzewa decyzyjne**
- **Losowe lasy**
- **Wielowarstwowy klasyfikator perceptronowy (analizowany w sekcji NN)**
- **Liniowa maszyna wektorów nośnych**
- **Klasyfikator jeden kontra reszta**
- **Naiwny Bayes**
- **Klasyfikator maszyn do faktoryzacji**

iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

Regresja logistyczna

- Regresja logistyczna jest popularną metodą przewidywania odpowiedzi kategorycznej.
- Jest to szczególny przypadek uogólnionych modeli liniowych, które przewidują prawdopodobieństwo wyników.
- Wielomianowa regresja logistyczna może być wykorzystana do klasyfikacji binarnej poprzez ustawienie parametru rodziny na „wielomianowy”. Wygeneruje dwa zestawy współczynników i dwa wyrazy wolne.



iBigWorld:2020-1-PL01-KA203-082197

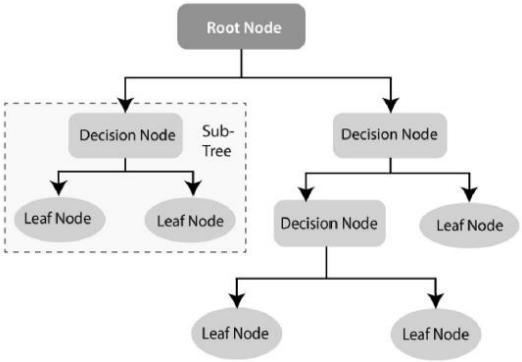
Co-funded by the
Erasmus+ Programme
of the European Union

Drzewo decyzyjne (DT)

- jest dobrze znaną nieparametryczną metodą nadzorowanego uczenia się. Metody uczenia DT są wykorzystywane zarówno do klasyfikacji i zadania regresji.
- ID3, C5.0 i CART [3] są dobrze znane z algorytmów DT. Algorytmy BehavDT [4] i IntrudTree [5] są skuteczne w odpowiednich dziedzinach aplikacji: analityki behawioralnej i cyberbezpieczeństwa.
- Instancje są klasyfikowane przez sprawdzenie atrybutu zdefiniowanego przez ten węzeł, zaczynając od węzła głównego drzewa, a następnie przechodząc w dół gałęzi drzewa odpowiadającej wartości atrybutu.
- W przypadku podziału najbardziej popularnymi kryteriami są algorytmy „gini” oraz „entropia”.

$$\text{Entropy} : H(x) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i)$$

$$\text{Gini}(E) = 1 - \sum_{i=1}^{c} p_i^2.$$

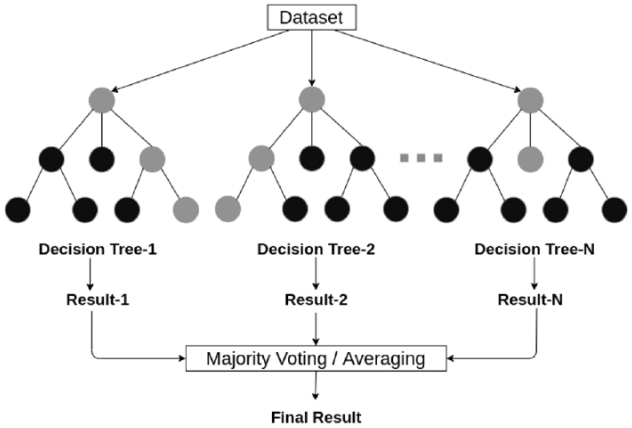


iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

Losowe lasy

- Losowy klasyfikator lasu jest dobrze znany jako technika klasyfikacji zespołowej, która jest stosowana w dziedzinie uczenia maszynowego i nauki o danych w różnych obszarach zastosowań.
- Metoda ta wykorzystuje „składanie równoległe”, które dopasowuje równolegle kilka klasyfikatorów drzew decyzyjnych, na różnych podzbiorach danych
- wykorzystuje głosowanie większościowe lub średnie dla wyniku lub wyniku końcowego.
- W ten sposób minimalizuje problem nadmiernego dopasowania i zwiększa dokładność przewidywania oraz kontrolę [6].

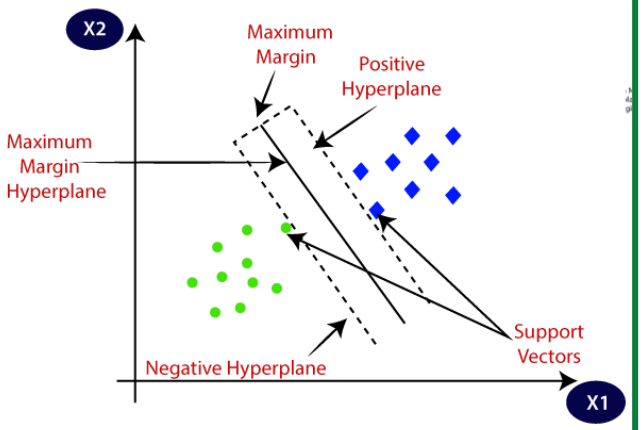


iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

Liniowa maszyna wektorów nośnych

- Inną powszechną techniką, którą można wykorzystać do klasyfikacji, regresji lub innych zadań, jest maszyna wektorów nośnych (SVM) [7].
- W przestrzeni wielowymiarowej lub nieskończenie wymiarowej maszyna wektorów nośnych konstruuje hiperpłaszczyznę lub zbiór hiperpłaszczyzn.
- Intuicyjnie, hiperpłaszczyzna, która ma największą odległość od najbliższych punktów treningowych w dowolnej klasie, osiąga silną separację, ponieważ ogólnie rzecz biorąc, im większy margines, tym niższy błąd uogólnienia klasyfikatora.
- Jest skuteczny w przestrzeniach wielowymiarowych i może zachowywać się inaczej w oparciu o różne funkcje matematyczne znane jako jądro: liniowa, wielomianowa, radialna funkcja bazowa (RBF), sigmoidalna itp.,

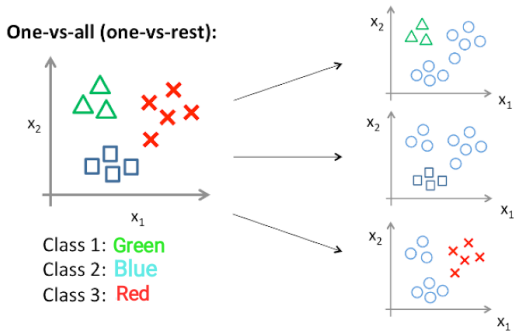


iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the
Erasmus+ Programme
of the European Union

Jeden kontra reszta

- OneVsRest to przykład redukcji uczenia maszynowego do wykonywania klasyfikacji wieloklasowej, biorąc pod uwagę klasyfikator podstawowy, który może wydajnie wykonywać klasyfikację binarną. Znany jest również jako „jeden przeciwko wszystkim”.
- OneVsRest jest zaimplementowany jako estymator. W przypadku klasyfikatora podstawowego przyjmuje wystąpienia klasyfikatora i tworzy problem klasyfikacji binarnej dla każdej z k klas. Klasyfikator klasy jest szkolony w celu przewidywania, czy jedna klasa odróżnia się od wszystkich innych klas.



Co-funded by the
Erasmus+ Programme
of the European Union

Naiwny Bayes (NB)

- Naiwny algorytm Bayesa opiera się na twierdzeniu Bayesa przy założeniu niezależności każdej pary cech [8]. Działa dobrze i może być używany zarówno w przypadku kategorii binarnych, jak i wieloklasowych w wielu rzeczywistych sytuacjach, takich jak klasyfikacja dokumentów lub tekstu, filtrowanie spamu itp.
- Kluczową korzyścią jest to, że w porównaniu z bardziej wyrafinowanych podejść, wymaga niewielkiej ilości danych uczących, aby szybko oszacować niezbędne parametry [9].
- Jednak na jego wydajność może mieć wpływ silne założenia dotyczące niezależności cech.
- Gaussowski, wielomianowy, dopełniający, Bernoulli i kategoryczny są powszechnymi wariantami klasyfikatora NB [9].
- MLIB Spark obsługuje: Bayes naiwny wielomianowy, Bayes naiwny uzupełniający, Bayes naiwny Bernoulliego i Bayes naiwny Gaussa.

Co-funded by the Erasmus+ Programme of the European Union

Naiwny klasyfikator faktoryzacji

- Maszyny do faktoryzacji są w stanie oszacować interakcje między funkcjami nawet w przypadku problemów z ogromną rzadkością (takich jak system reklam i rekomendacji). Implementacja spark.ml obsługuje maszyny do faktoryzacji do klasyfikacji binarnej i regresji. Faktoryzacja odbywa się wg formuły:

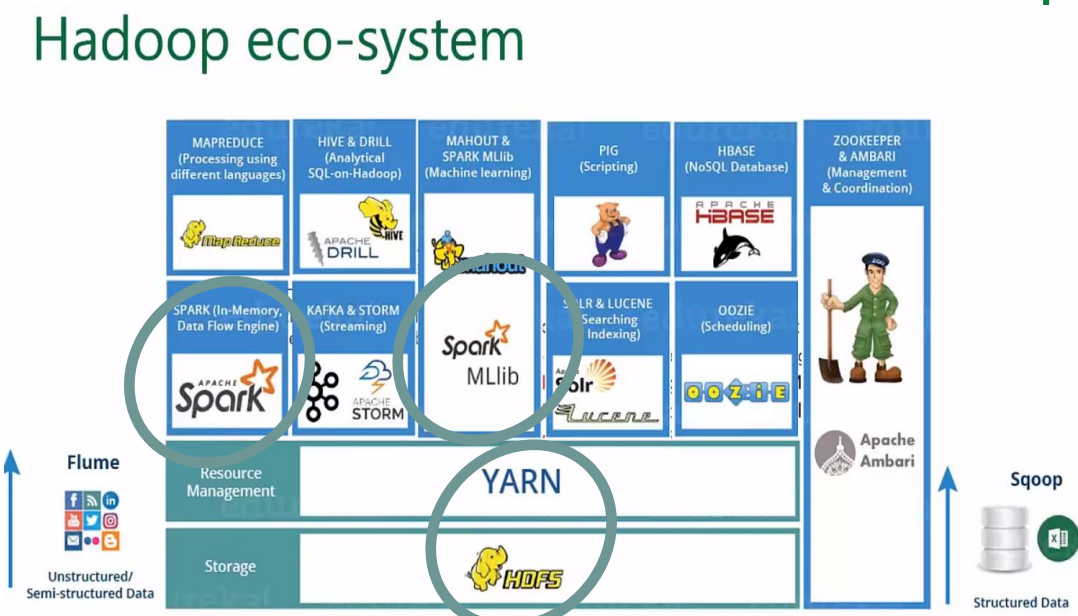
$$\hat{y} = w_0 + \sum_{i-1}^{n} w_i x_i + \sum_{i=1}^{n} \sum_{j=i+1}^{n} \langle v_i, v_j \rangle x_i x_j$$

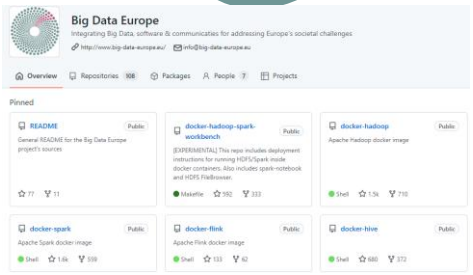
iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the
Erasmus+ Programme
of the European Union

Haadop a klasyfikacja?

- HDFS — system plików dla BIG Data
- Yarn - wbudowany menedżer zasobów i harmonogram zadań
- **MapReduce — przetwarzanie równoległe**
- **Spark - silnik przetwarzania danych**
- Oozie - kalendarz zarządzania strumieniami
- Ambari - zarządzanie i monitoring
- Sqoop - transfer między Hadoop a relacyjnym BD
- Hbase - losowy dostęp do danych w czasie rzeczywistym
- Kafka - platforma do przesyłania wiadomości między aplikacjami a strumieniem danych





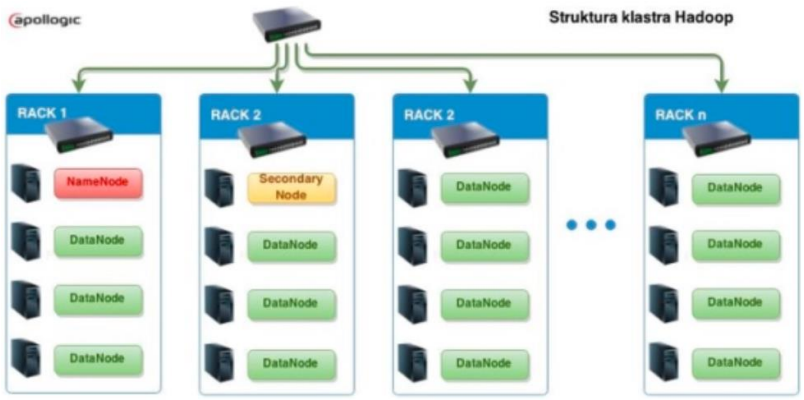
Implementacje:
https://github.com/big-data-europe

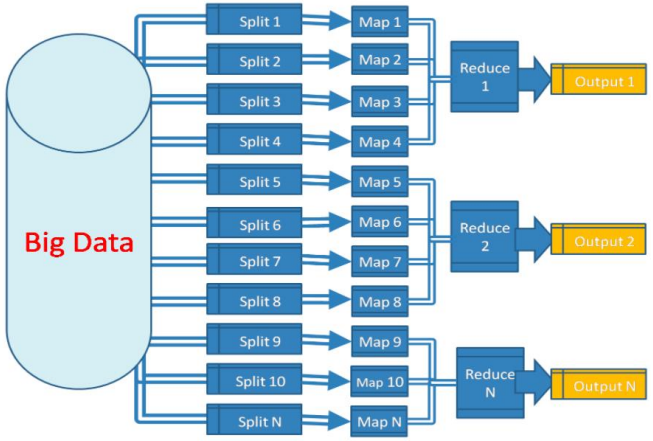
iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

HDFS/MapReduce

- Struktura umożliwiająca przechowywanie i zarządzanie rozproszonym zbiorem danych

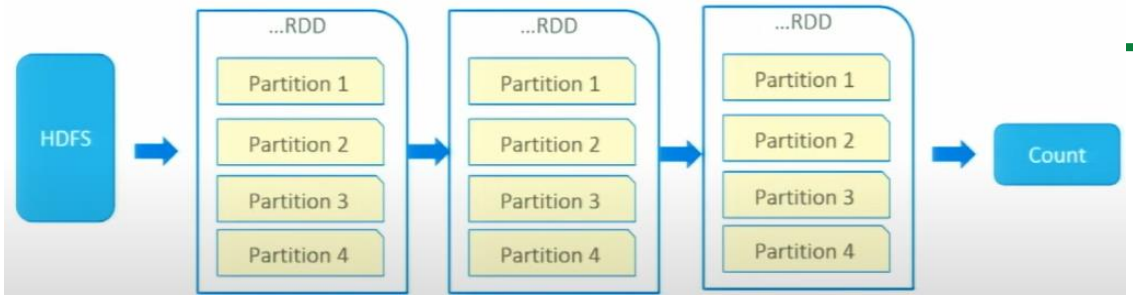




iBigWorld:2020-1-PL01-KA203-082197

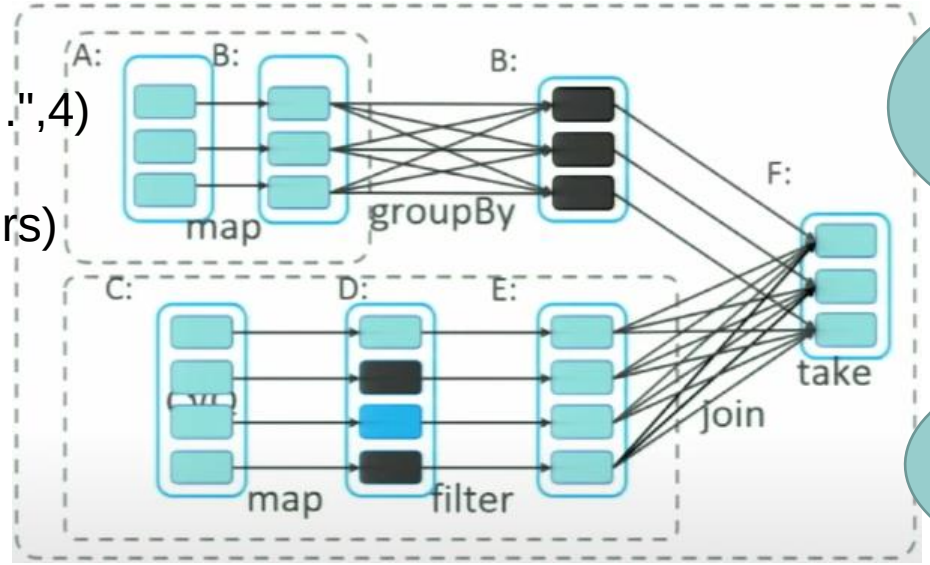
Co-funded by the Erasmus+ Programme of the European Union

MapReduce kontra Spark



Wyniki operacji są zapisywane po każdej transformacji

```
sc.textFile("hdfs://...",4)
  .map(to_series)
  .filter(has_outlayers)
  .count()
```



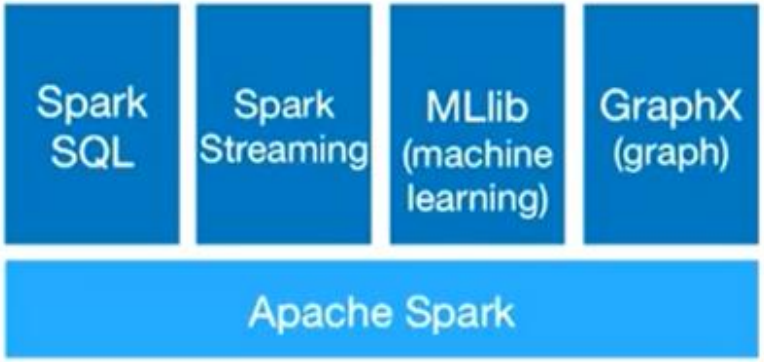
Wyniki operacji przechowywane w pamięci (jeśli to możliwe)

Wyróżnianie działań i transformacji

[Cloudera Sumit 2016 Juliet Hougland, Senior Data Scientist, Cloudera]

Co-funded by the Erasmus+ Programme of the European Union

Przetwarzanie



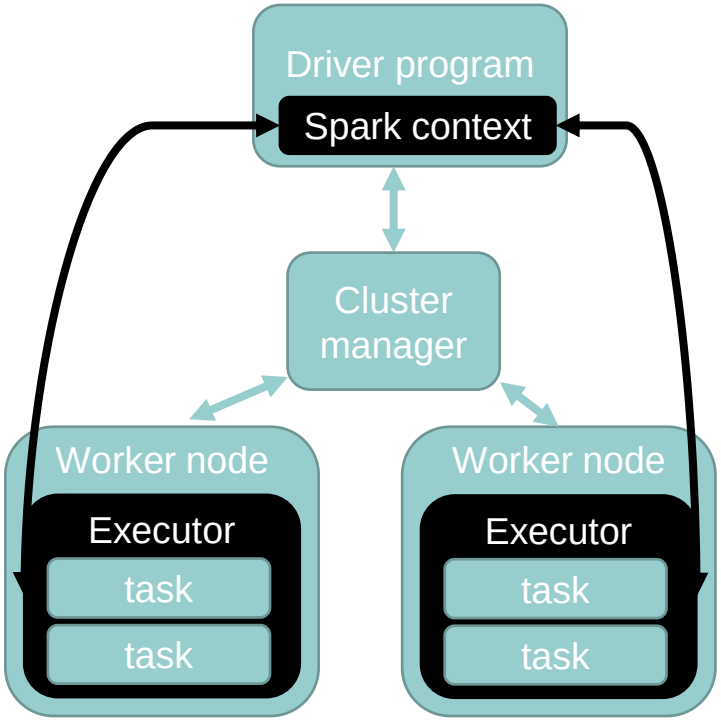
- Spark „core” — baza Spark z obsługą podstawowej abstrakcji danych RDD
- Spark SQL - komponent, który pozwala manipulować danymi za pomocą poleceń SQLSpark
- ML - Komponent zawierający algorytmy ML dostępne w Spark
- Spark Streaming - moduł pozwalający na pracę ze strumieniami danych
- Spark GraphX - komponent do pracy z wykresami

iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

Architektura Spark

- Sterownik — proces, który uruchamia główną aplikację i tworzy Spark
- ContextExecutor — proces uruchomiony dla aplikacji w węźle roboczym, który uruchamia zadania i przechowuje dane w pamięci lub na dysku.
- Każda aplikacja ma swojego wykonawcę (executorów)
- Menedżer klastra - dostępne opcje:
  - Yarn,Mesos,Kurernettes,Standalone



iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

Pracuj ze Sparkiem

- SparkContext:
  - Punkt wejścia do pracy z SparkCoordinates procesów w klastrze
    - SparkContext == nazywana najczęściej „sc”
  - SparkSession: wprowadzona w Spark 2.0
    - Zawiera:
      - SparkContext,
      - SQL context
      - HiveContext
    - nazywana „spark”

iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

Jednostki do przetwarzania w Spark

- RDD
  - Basic abstention at Spark
  - R - resilient
  - D - distributed
  - D - dataset
  - Immutable
  - In-memory
  - Lazy evaluated
  - Parallel
  - Two types: actions and transformations

```
>>> distFile = sc.textFile("data.txt")
```

```
distFile.map(lambda s: len(s)).reduce(lambda a, b: a + b)
```

- DataFrame
  - Data abstraction (SQL module)
  - Immutable
  - In-memmory
  - Resilient
  - Distributed
  - Parallel
  - Stores information about the structure (schema)
  - A distributed collection of rows with names and columns
  - Optimized by Catalyst Optimizer
  - It allows you to work with data via SQL

```
root
 |-- name: struct (nullable = true)
 |    |-- firstname: string (nullable = true)
 |    |-- middlename: string (nullable = true)
 |    |-- lastname: string (nullable = true)
)
```

iBigWorld:2020-1-PL01-KA203-082197

Co-funded by the Erasmus+ Programme of the European Union

Referencje

[1] Han J, Pei J, Kamber M. Data mining: concepts and techniques. Amsterdam: Elsevier; 2011.
[2] Lina Zhoua,∗, Shimei Pana, Jianwu Wanga, Athanasios V. Vasilakosb Machine learning on big data: Opportunities and challenges Neurocomputing 237 (2017) 350–361
[3] Breiman L, Friedman J, Stone CJ, Olshen RA. Classification and regression trees. CRC Press; 1984.
[4] Sarker IH, Alan C, Jun H, Khan AI, Abushark YB, Khaled S. Behavdt: a behavioral decision tree learning to build user-centric context-aware predictive model. Mob Netw Appl. 2019; 1–11. 101.
[5] Sarker IH, Abushark YB, Alsolami F, Khan A. Intrudtree: a machine learning based cyber
[6] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, et al. Scikit-learn: machine earning in python. J Mach Learn Res. 2011;12:2825–30.
[7] Keerthi SS, Shevade SK, Bhattacharyya C, Radha Krishna MK. Improvements to platt's smo algorithm for svm classifier design. Neural Comput. 2001;13(3):637–49.
[8] John GH, Langley P. Estimating continuous distributions in bayesian classifiers. In: Proceedings of the Eleventh conference on Uncertainty in artificial intelligence, Morgan Kaufmann Publishers Inc. 1995; 338–345
[9] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, et al. Scikit-learn: machine learning in python. J Mach Learn Res. 2011;12:2825–30.

Co-funded by the Erasmus+ Programme of the European Union