Detekcja Anomalii

Inżynieria wiedzy i uczenie maszynowe

Laboratorium

mgr inż. Jan Garus
jan.garus@comarch.pl

Akademia Górniczo-Hutnicza im. Stanisława Staszica w Krakowie
AGH University of Science and Technology

13 grudnia 2021

Środowisko

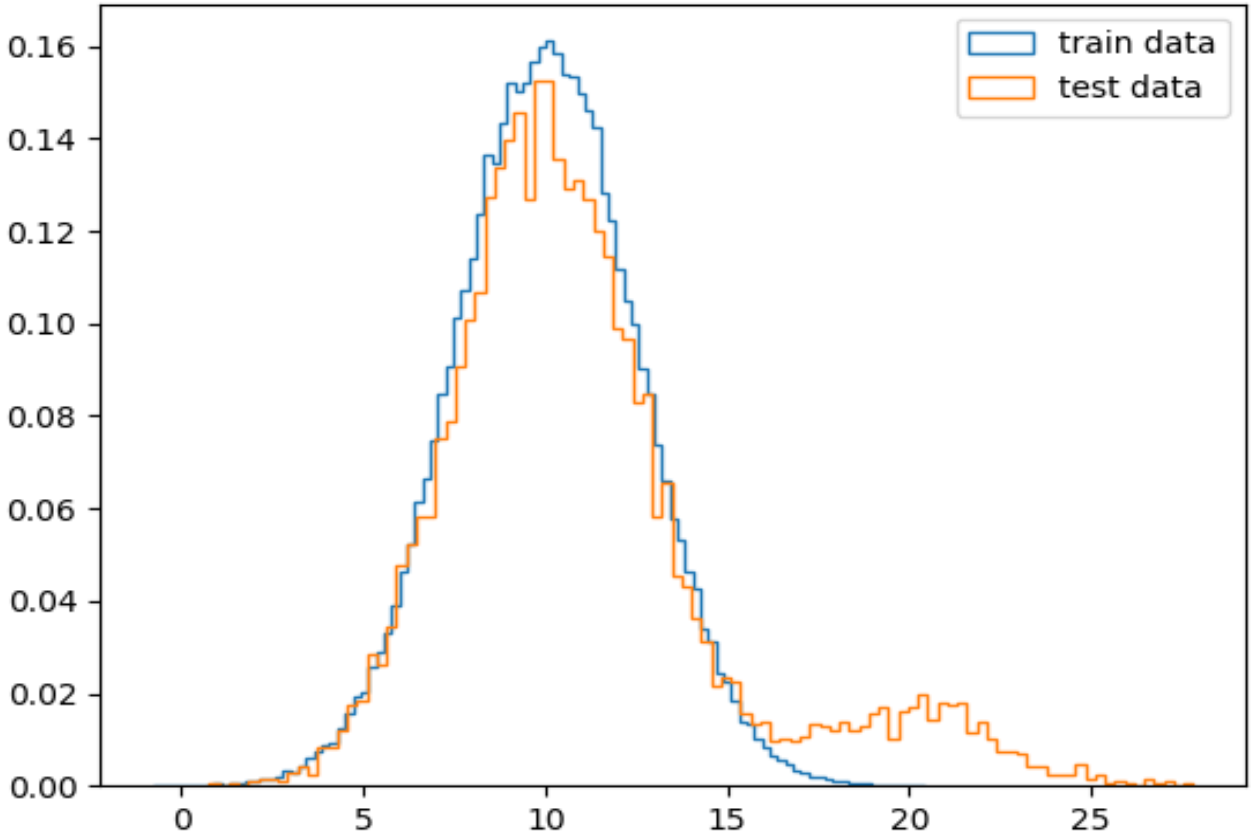
- Używamy Python 3.12.
- W pliku requirements.txt znajduje się lista wymaganych pakietów Python.
- Najlepiej (ale nie jest to konieczne) utworzyć środowisko Virtual Environment:
    - Utworzenie środowiska virtual environment:
      virtualenv -p /usr/bin/python3.12 iwum_anomaly
    - Aktywacja środowiska:
      source iwum_anomaly/bin/activate
- Instalacja pakietów:
  pip install -r requirements.txt

Informacje ogólne

- Kody źródłowe do kolejnych zadań znajdują się w katalogach: ex1, ex2, ex3, ex4 i ex5.
- Skrypty exN/scriptN.py to programy realizujące detekcję anomalii.
- Każdy z w/w skryptów wykorzystuje funkcje do napisania przez studentów - ich atrapy znajdują się w plikach exN/solutionN.py
- Anomaliom należy przypisać klasę 1, zaś przykładom normalnym klasę 0.
- W pliku utils.py znajduje się m.in. przydatna funkcja binary2neg_boolean do konwersji (z negacją) wyników binarnych -1/1 na 0/1.
- W celu zaliczenia zajęć należy przesłać via UPEL (Detekcja Anomalii - zadanie):
    - Uzupełnione pliki exN/solutionN.py
    - Raport zawierający:
        - uzyskane F1-score
        - otrzymane wykresy
        - wskazane obserwacje/wnioski/odpowiedzi na pytania.

Zadanie 1
Model statystyczny 1D

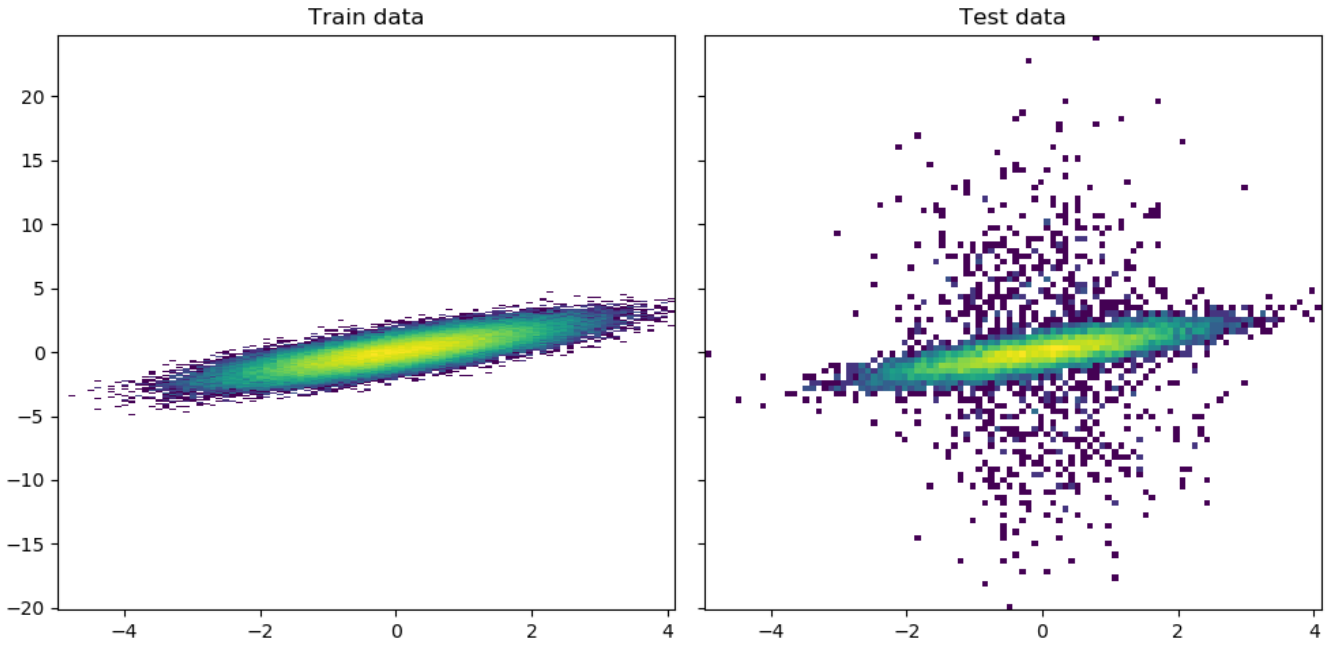
1) Założyć typ rozkładu statystycznego danych uczących
2) Wyestymować jego parametry na podstawie danych uczących
3) Określić próg detekcji anomalii w odniesieniu do obliczonych parametrów rozkładu statystycznego
4) Obliczyć wyniki detekcji dla danych testowych



Zadanie 2

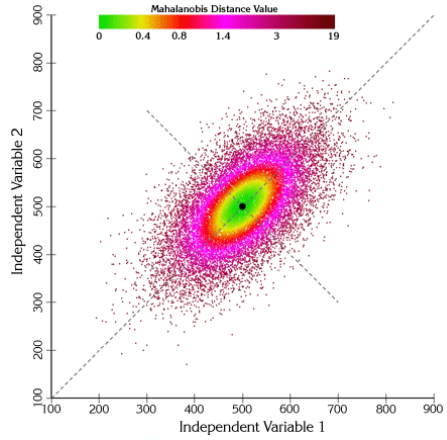
Model statystyczny 2D

1) Wyestymować macierz kowariancji rozkładu statystycznego danych uczących, np. przy pomocy pakietu sklearn.covariance.MinCovDet
2) Obliczyć maksymalną odległość przykładów uczących od wartości oczekiwanej (średniej) wg metryki Mahalanobisa - patrz: MinCovDet.mahalanobis()
3) Obliczyć odległości Mahalanobisa dla przykładów testowych i na tej podstawie określić czy są artefaktami.



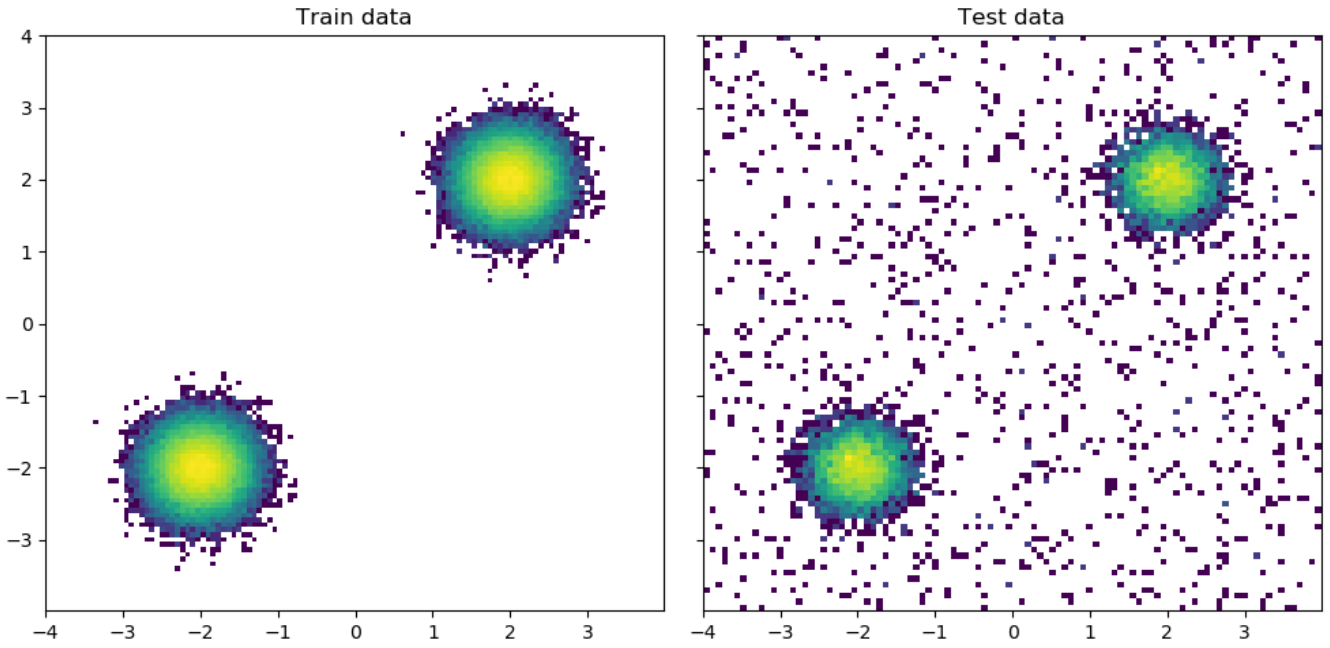
Odległość Mahalanobisa
Kolor wskazuje odległość od punktu na środku

Źródło:
http://www.jennessent.com/arcview/mahalanobis_description.htm



Zadanie 3
Model statystyczny vs OC-SVM

1) Przeprowadzić detekcję anomalii na zbiorze testowym wykorzystując odległość Mahalanobisa, jak w zadaniu 2 (użyć funkcji ponownie).
2) Wyniki porównać z wynikami na zbiorze testowym uzyskiwanymi przez algorytm OneClass-SVM: sklearn.svm.OneClassSVM Wykorzystać odpowiedni kernel.
3) Skomentować wyniki - jakie skłonności mają te algorytmy? tj. w jakich sytuacjach są odpowiednie?
4) Jaki wpływ na wyniki ma manipulacja parametrami algorytmu OC-SVM?

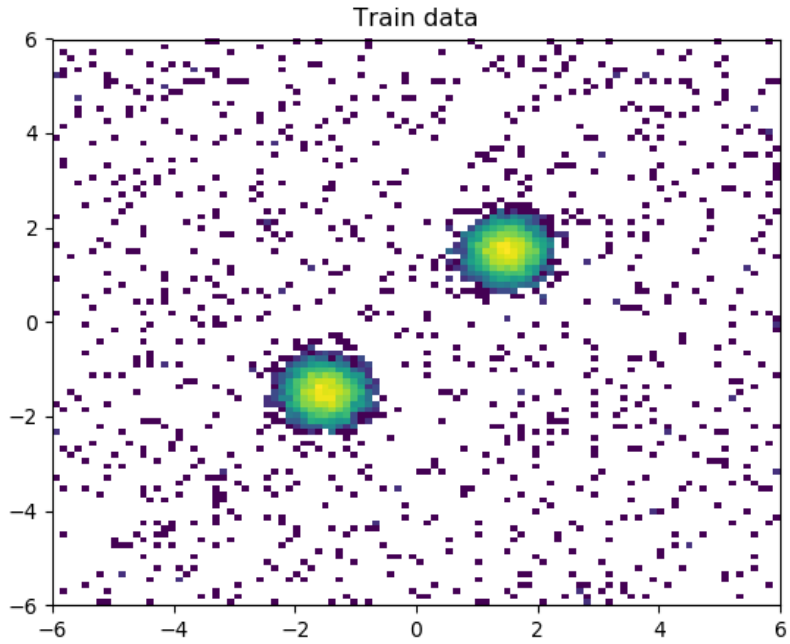


Zadanie 4
Uczenie w obecności anomalii

Porównanie wybranych algorytmów uczonych bez nadzorcy na zbiorze zawierającym anomalie.

Stosunek liczby anomalii do wielkości zbioru uczącego jest znany.

1) Estymacja kowariancji i odległość Mahalanobisa - można użyć gotowej implementacji: sklearn.covariance.EllipticEnvelope
2) OneClass-SVM: sklearn.svm.OneClassSVM
3) Isolation Forest sklearn.ensemble.IsolationForest
4) Local Outlier Factor sklearn.neighbors.LocalOutlierFactor



1) Użyć poszczególnych metod.
2) Skomentować dobór ich parametrów – co zmieniają?
3) Testy na zbiorze uczącym – podać wyniki.
4) Porównać zachowanie się poszczególnych metod. tj. jakie mają skłonności – do czego się nadają.

Zadanie 5
AutoEnkoder

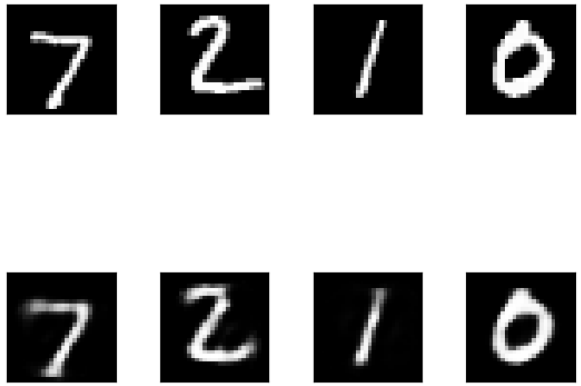
Jako przykłady normalne traktowane są rekordy z bazy pisanych odręcznie cyfr MNIST.

Zbiór testowy składa się z rekordów z MNIST (traktujemy jako klasę nominalną) oraz bazy Fashion MNIST zawierającej fotografie ubrań (traktujemy je jako anomalie nieznane na etapie uczenia).

1) Zaproponować miarę jakości rekonstrukcji (ex5/solution5/reconstruction_errors())
2) Zaproponować sposób wyznaczenia progu błędu rekonstrukcji oznaczającego anomalię (ex5/solution5/calc_threshold())
3) Wykryć anomalie na podstawie tego progu (ex5/solution5/predict())
4) Omówić histogramy błędów rekonstrukcji dla obu zbiorów.
5) Opisać inne spostrzeżenia dot. eksperymentu.

Dla chętnych:

Zbadać wpływ zwiększenia liczby neuronów w warstwie ukrytej (latent) na skuteczność detekcji anomalii.
Zbadać wpływ zwiększenia liczby warstw autoenkodera na skuteczność detekcji anomalii.



Przykłady danych wejściowych (górne wiersze) i ich rekonstrukcji (wiersze dolne) zwrócone przez AE uczony tylko na MNIST.

Przedstawiono wyniki dla zbiorów: MNIST (na górze strony↑) oraz Fashion-MNIST (na dole strony↓)