ML - przypomnienie

Podstawowa klasyfikacja metod 1

Ze względu na sposób uczenia:

- Uczenie nadzorowane (*ang.* supervised learning)
- Uczenie nienadzorowane (*ang.* unsupervised learning)
- Uczenie ze wzmocnieniem (*ang.* reinforcement learning)

Uczenie nadzorowane

Model uczy się odwzorowywać dane wejściowe na dane wyjściowe $X \rightarrow Y$.

W tym zagadnieniu dane są pary wartości wejściowych x_i oraz wyjściowych y_i.

Przykłady modeli i algorytmów:

- Parametryczne:
  - regresja liniowa/nieliniowa (*ang.* linear/non-linear regression),
  - logistyczna (*ang.* logistic regression)
- Nieparametryczne:
  - maszyna wektorów wspierających (*ang.* support vector machine SVM),
  - drzewa losowe (*ang.* random trees),
  - XGBoost

Uczenie nienadzorowane

Algorytm wyszukuje w danych wzorce (ang. pattern) nie mając informacji wyjściowej (Y). Wyszukiwanie wzorców odbywa się z wykorzystaniem optymalizacji (maksymalizacja funkcji celu lub minimalizacja funkcji kosztu).

Przykłady modeli i algorytmów (nieparametryczne):

- algorytm centroidów (*k*-means)
- t-distributed Stochastic Neighbor Embedding (t-SNE)
- Uniform Manifold Approximation and Projection for Dimension Reduction (uMAP)

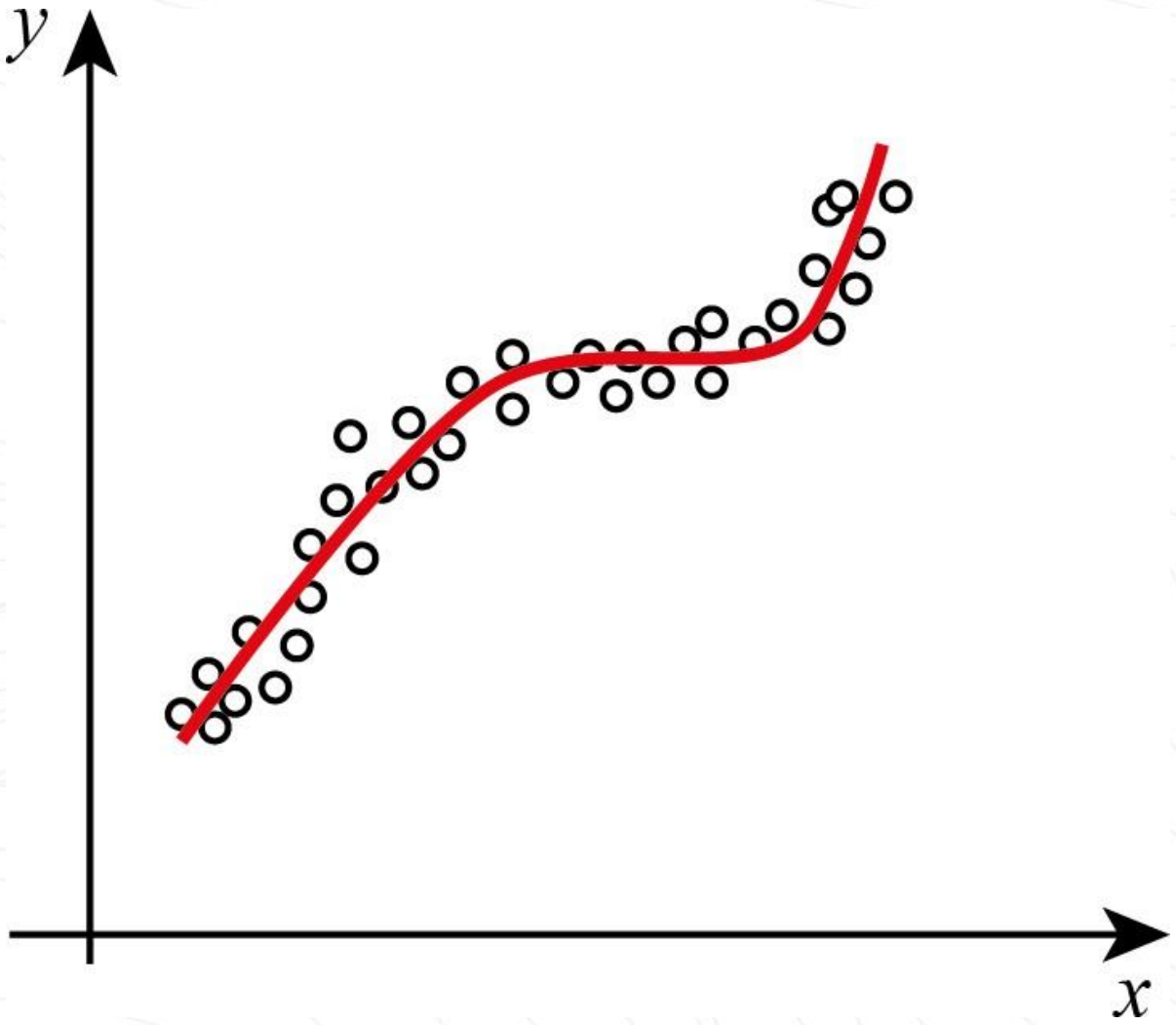
Podstawowa klasyfikacja metod 2

Ze względu na rodzaj zagadnienia:

- Klasyfikacja (*ang.* classification)
- Regresja (*ang.* regression)
- Klasteryzacja (*ang.* clustering)
- Inne:
  - Redukcja wymiarowości (*ang.* dimensionality reduction)
  - Uczenie zależności (*ang.* association rule mining)
  - ...

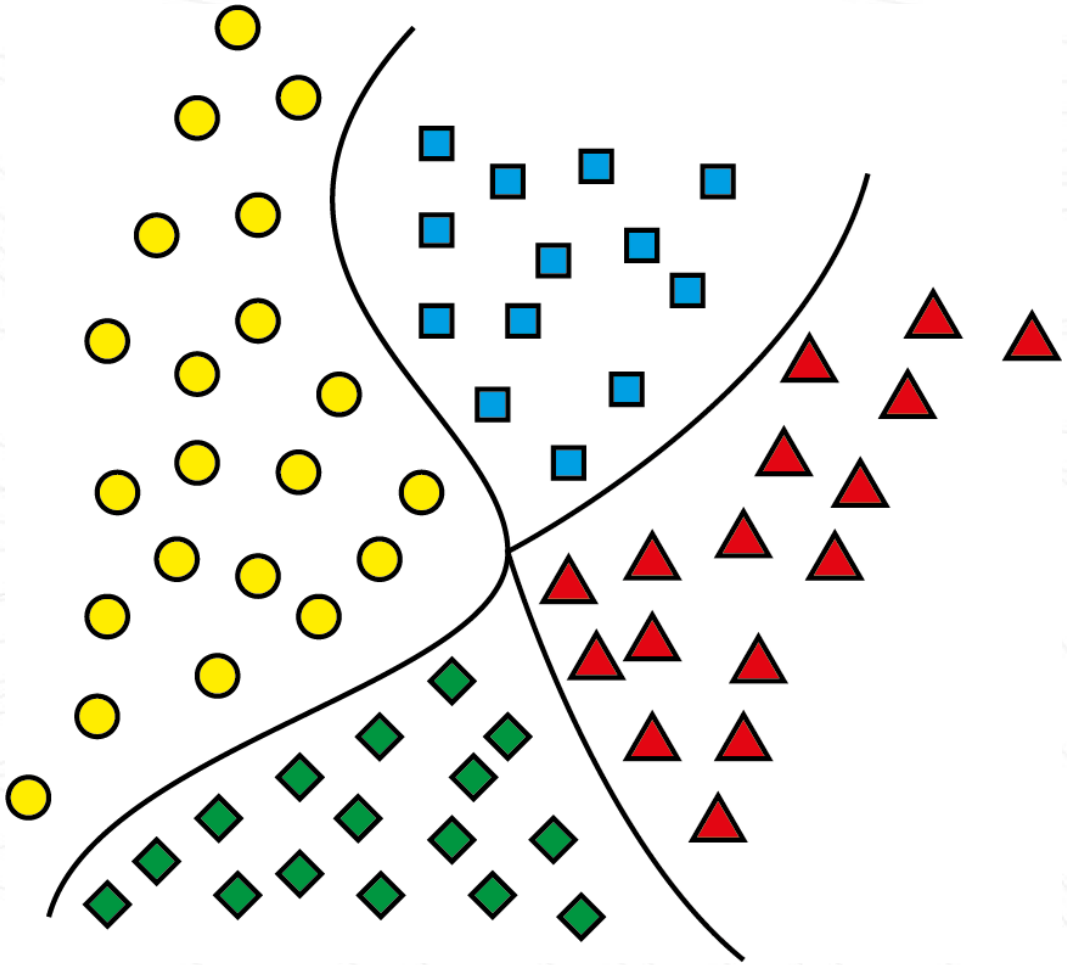
Regresja

Zmienna Y jest w założeniu ciągła.



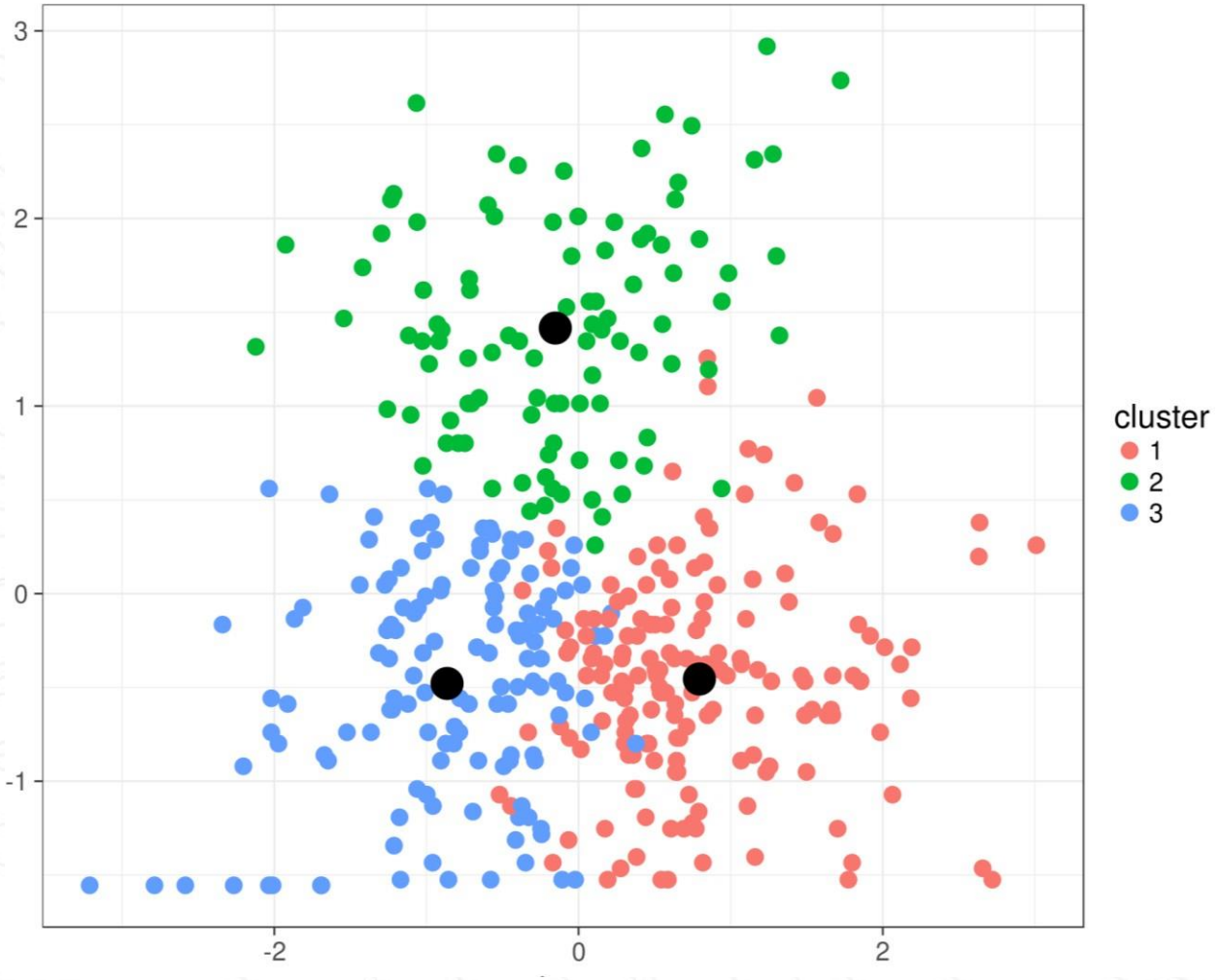
Klasyfikacja

Zmienna Y określa przynależność do klasy.



Klasteryzacja

Poszukiwanie wzorców w danych.



Podstawowa klasyfikacja metod 3

Ze względu na typ modelu:

- Parametryczne (*ang.* parametric)
- Nieparametryczne (*ang.* non-parametric)

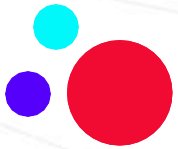
Modele parametryczne

Modele parametryczne mają z góry określoną postać funkcyjną, a uczenie modelu polega na określeniu parametrów. Nadają się raczej do nieskomplikowanych zagadnień.

Przykłady modeli parametrycznych:

- regresja liniowa i nieliniowa
- regresja logistyczna

Np. w przypadku regresji liniowej $y=a \cdot x+b$ parametrami są a i b.



Modele nieparametryczne

Modele nieparametryczne nie posiadają określonej formy funkcyjnej.

Są to zwykle bardzo elastyczne modele umożliwiające tworzenie bardzo skomplikowanych odwzorowań.

Metryki między obserwacjami

Metryki (ang. metrics, scoring) służą do oceny efektywności modelu.

W różnych zagadnieniach stosuje się różne metryki. Metryk jest bardzo wiele np.

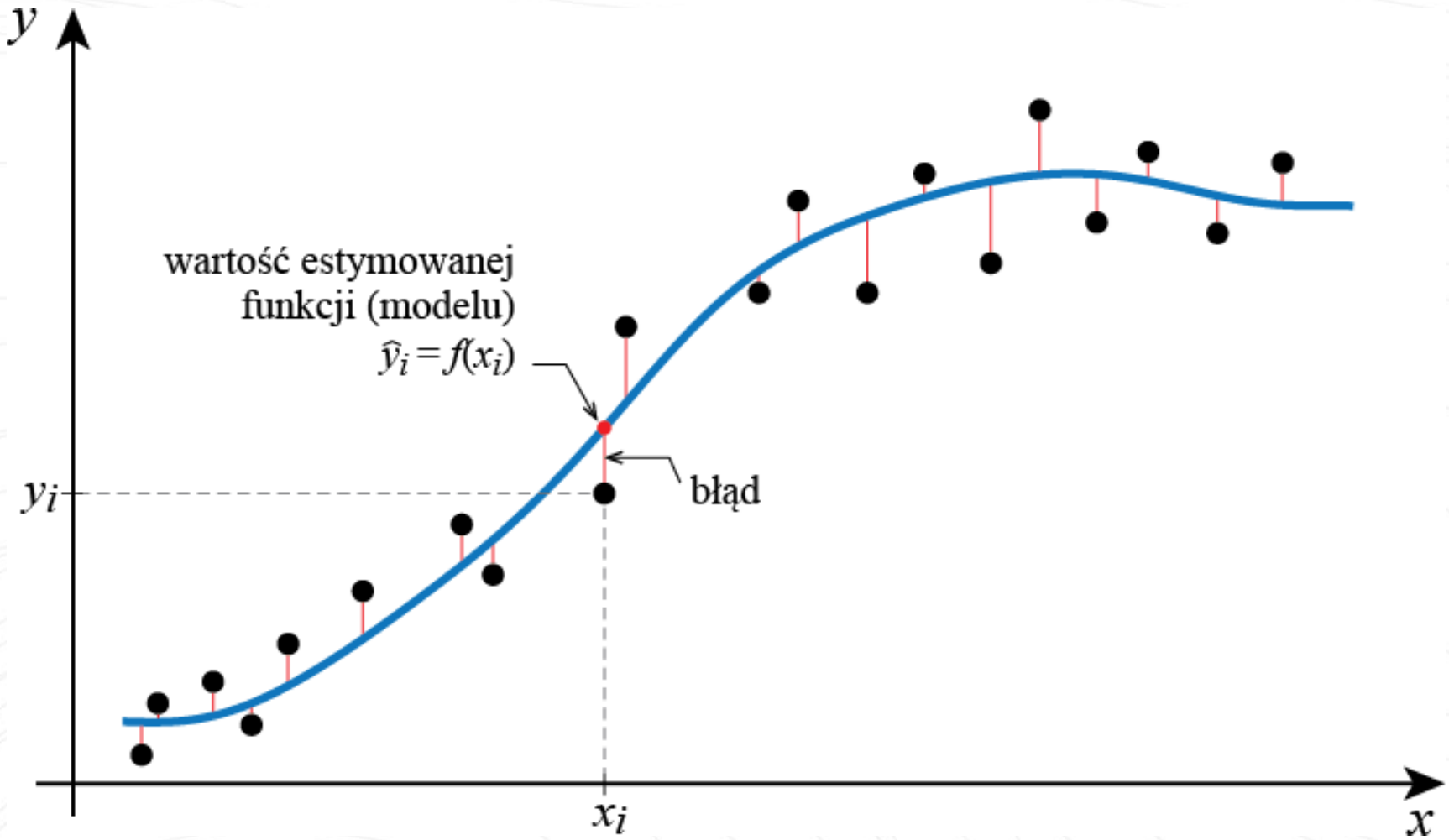
lista dostępnych metryk w scikit-learn.

W zagadnieniach klasyfikacji, regresji i klasteryzacji używa się różnych metryk.

Stosując różne metryki można uzyskać inne (lepsze/gorsze) wyniki dla danego modelu i identycznych danych.

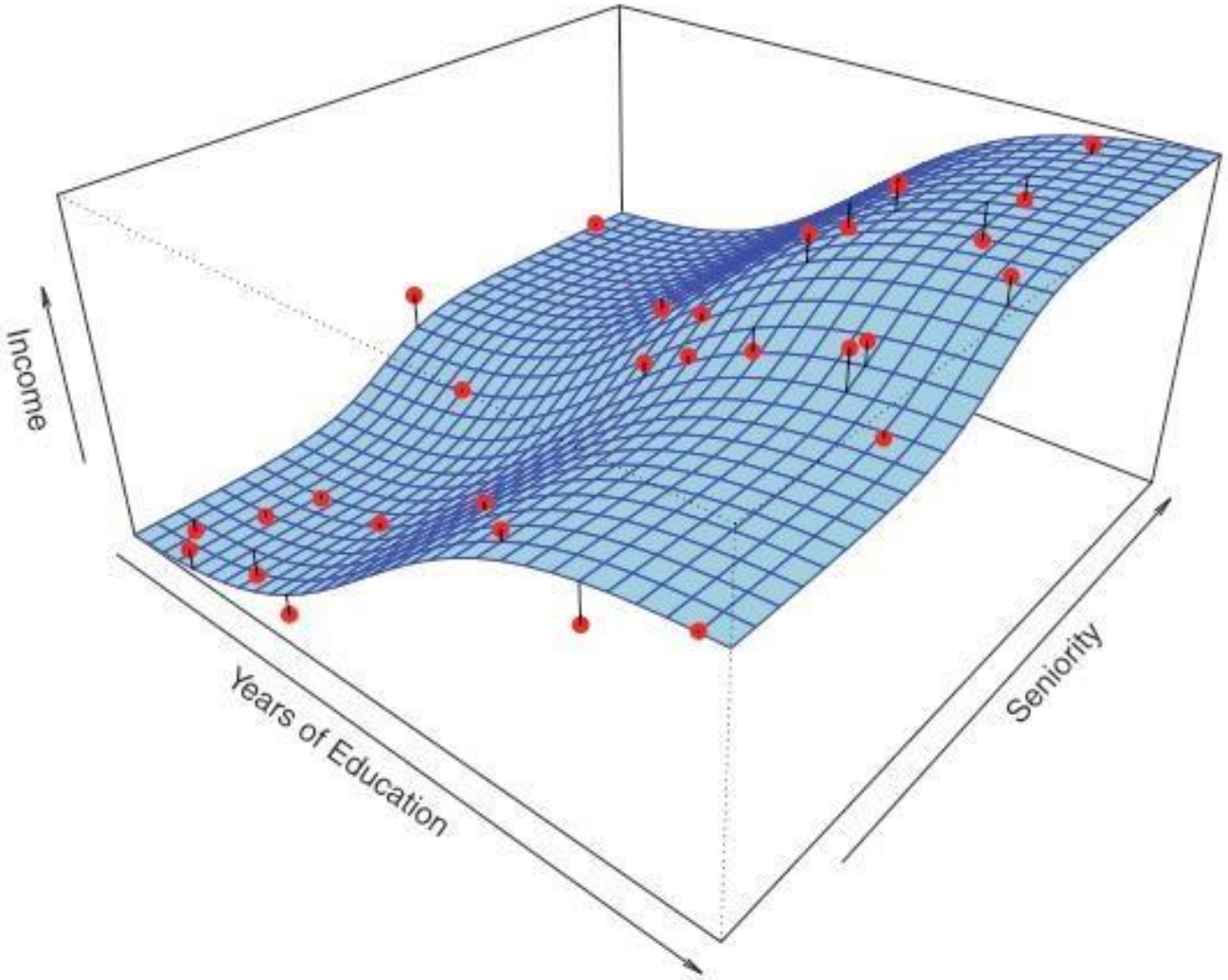
Regresja

Błąd w regresji.



Regresja

Błąd w regresji.



Regresja przykładowy wynik

indeks	obserwacje (dane)	model (predykcja)	błąd
i	y_i	$\hat{y}_i$	$y_i - \hat{y}_i$
1	2,4	2,5	-0,1
2	25,3	27,5	-2,2
3	4,5	3,8	0,7
...	...	...	...
1024	12,1	11,8	0,3

Regresja podstawowe metryki

Podstawowymi metrykami dopasowania modelu są

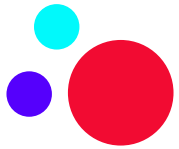
- średni błąd kwadratowy (mean square error)

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

- pierwiastek MSE (root mean square error) $RMSE = \sqrt{MSE}$

- średni błąd bezwzględny (mean absolute error)

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$



Praktyka ML

Budowanie modeli

infoShareAcademy.com

Uczenie maszynowe w praktyce

1. Analiza problemu
2. Analiza i eksploracja danych
3. Inżynieria cech (*ang.* feature engineering)
4. Wybór modelu
5. Wybór metryk
6. Trenowanie modelu
7. Testowanie
8. Analiza wyników

Uczenie maszynowe w praktyce

W praktyce uczenie maszynowe rozumie się jako zespół czynności składający się z kilku następujących po sobie elementów:

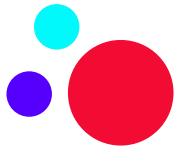
1. Analiza problemu, który ma zostać rozwiązany przy wykorzystaniu danych (np. oszacowanie liczby klientów w danym dniu roku, segmentacja klientów, analiza powiązań) i wybór odpowiedniej metody uczenia maszynowego do rozwiązania danego zagadnienia.
2. Analiza i eksploracja danych w celu wyboru modelu i możliwie najlepszego przygotowania danych (np. usuwanie obserwacji odstających, usuwanie danych skorelowanych, itp.).
3. Inżynieria cech (*ang.* feature engineering) – wprowadzenie dodatkowych danych (np. one-hot encoding, flagi klasteryzacji), redukcja wymiarowości itp.

Uczenie maszynowe w praktyce

4. Wybór modelu będzie zależał od charakteru danych i celu jaki należy osiągnąć. W niektórych zagadnieniach akceptowalna będzie niższa skuteczność modelu, jeśli model będzie oferował możliwość interpretacji.
5. Wybór metryk adekwatnych do modelu i zagadnienia. Błąd modelu można mierzyć na różne sposoby, ważne jest aby dokonać właściwego wyboru metryki. ogólnie rzecz biorąc nie ma metryk lepszych i gorszych, wszystko zależy od konkretnego problemu.
6. Trenowanie modelu. Polega na określeniu postaci estymatora (np. parametrów modelu) na podstawie danych.
7. Testowanie modelu. Polega na sprawdzeniu efektywności modelu na danych, które nie zostały wykorzystane do uczenia (trenowania).

Uczenie maszynowe w praktyce

8. Analiza wyników. Stwierdzenie czy model jest skuteczny (tj. pozwala na osiągnięcie założonego na początku celu), sprawdzenie czy model nie jest obciążony (*ang.* bias) oraz czy nie jest nadmiernie dopasowany (*ang.* overfitting).



scikit-learn API

Uczenie maszynowe w Pythonie

infoShareAcademy.com

scikit-learn API

Scikit-learn oferuje bardzo spójne API.

Przepływ pracy jest następujący:

1. Utworzenie obiektów (encoder, scaler, classifier, regressor, itp.)
2. Użycie metody fit z argumentem w postaci danych
3. Użycie metody transform lub predict w celu obliczenia transformacji lub predykcji
4. (użycie pipeline)

Biblioteki ML Python

- scikit-learn, statsmodels, XGBoost, uMAP
- TPOT
- pyCARET

Literatura

- *Mistrz analizy danych*. Foreman. Helion
- *Python. Podstawy nauki o danych.* Boschetti, Massaron, Helion
- *Szeregi czasowe.* Nielsen. Helion.
- *Uczenie maszynowe z użyciem Scikit-Learn i Tensorflow.* Géron. Helion.
- *Zaawansowane uczenie maszynowe z językiem Python.* Hearty. Helion.
- The elements of statistical learning. Hastie, Tibshirani, Friedman. Springer. (PDF dostępny w sieci)
- *An introduction to statistical learning with applications in R*. James, Witten, Hastie, Tibshirani. Springer. (PDF dostępny w sieci)

Dzięki!