POLITECHNIKA WROCŁAWSKA
WYDZIAŁ INFORMATYKI I TELEKOMUNIKACJI

METODY ANALIZY I EKSPLORACJI DANYCH

Wykład 9 - Predykcja: Regresja, Walidacja krzyżowa

DR INŻ. AGATA MIGALSKA

01

PROBLEM REGRESJI

02

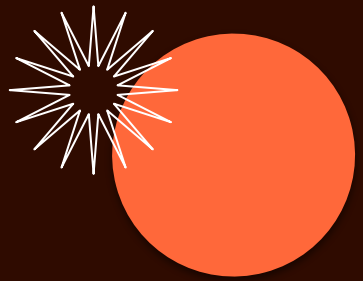
REGRESJA LINIOWA I REGULARYZACJA

03

DOBROĆ REGRESORA

04

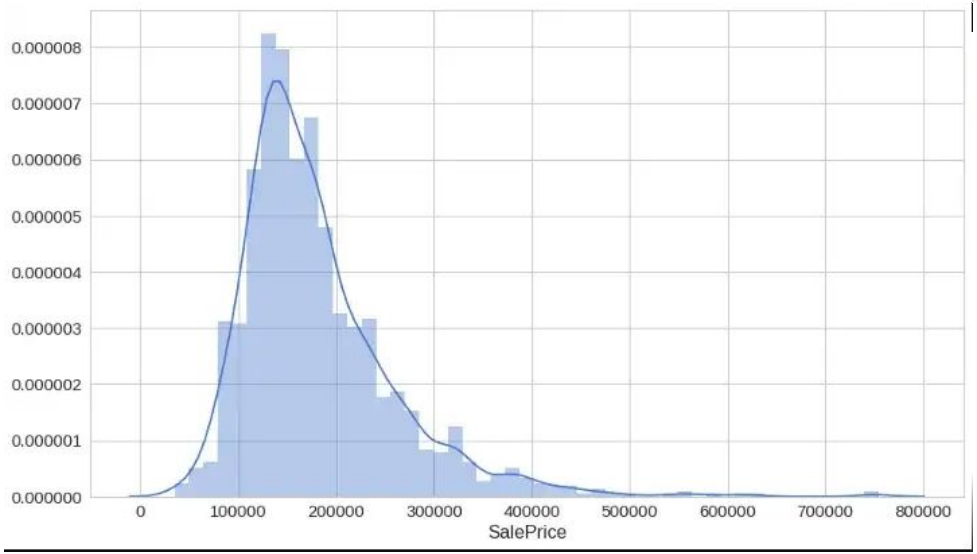
WALIDACJA KRZYŻOWA



01

PROBLEM REGRESJI

PROBLEM REGRESJI



Data fields

Here's a brief version of what you'll find in the data description file.

- **SalePrice** - the property's sale price in dollars. This is the target var
- **MSSubClass**: The building class
- **MSZoning**: The general zoning classification
- **LotFrontage**: Linear feet of street connected to property
- **LotArea**: Lot size in square feet
- **Street**: Type of road access
- **Alley**: Type of alley access
- **LotShape**: General shape of property
- **LandContour**: Flatness of the property
- **Utilities**: Type of utilities available

Źródło: https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques/overview

PRZYPOMNIENIE: KLASYFIKATOR K-NN

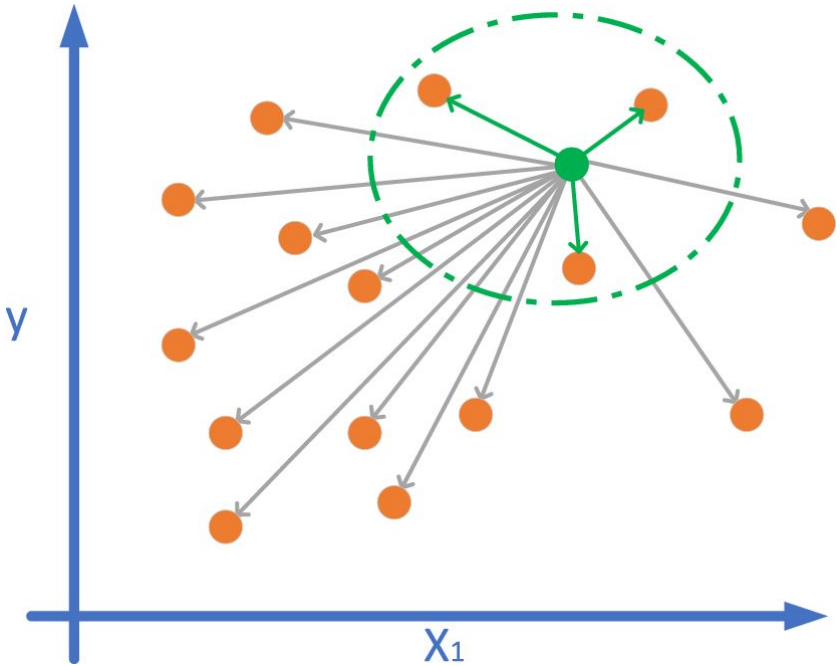
Dane:

- Zbiór uczący X_{train} i zbiór odpowiadających mu etykiet y_{train}.
- Nowa obserwacja x_{test}, która ma być zaklasyfikowana.

Algorytm:

1. Znajdź w X_{train} k obserwacji, które są najbardziej podobne do x_{test} - nazwijmy ten zbiór X_{NN}.
2. Pobierz etykiety y_{NN} odpowiadające zbiorowi X_{NN}.
3. Wyznacz etykietę dla obserwacji x_{test} na podstawie etykiet y_{NN} np. poprzez głosowanie większościowe.

REGRESOR K-NAJBLIŻSZYCH SĄSIADÓW



- Analogicznie do klasyfikatora, do predykcji wartości dla nowej obserwacji patrzymy na k najbliższych sąsiadów.
- Predykowana wartość jest średnią z wartości sąsiadów.
- Wprowadzając wagi (np. w zależności od odległości sąsiada od nowej obserwacji) dostajemy ważony regresor.

Źródło grafiki: https://www.researchgate.net/figure/k-Nearest-neighbors-regression_fig3_358473767

SUPPORT VECTOR REGRESSOR

1.2. *The basic idea*

Suppose we are given training data $\{(x_1, y_1), \ldots, (x_\ell, y_\ell)\} \subset \mathcal{X} \times \mathbb{R}$, where $\mathcal{X}$ denotes the space of the input patterns (e.g. $\mathcal{X} = \mathbb{R}^d$). These might be, for instance, exchange rates for some currency measured at subsequent days together with corresponding econometric indicators. In ε-SV regression (Vapnik 1995), our goal is to find a function $f(x)$ that has at most ε deviation from the actually obtained targets y_i for all the training data, and at the same time is as flat as possible. In other words, we do not

For pedagogical reasons, we begin by describing the case of linear functions f, taking the form

$$f(x) = \langle w, x \rangle + b \text{ with } w \in \mathcal{X}, b \in \mathbb{R} \tag{1}$$

where $\langle \cdot, \cdot \rangle$ denotes the dot product in $\mathcal{X}$. *Flatness* in the case of (1) means that one seeks a small w. One way to ensure this is to minimize the norm,[3] i.e. $\|w\|^2 = \langle w, w \rangle$. We can write this problem as a convex optimization problem:

$$\begin{aligned} &\text{minimize} && \tfrac{1}{2}\|w\|^2 \\ &\text{subject to} && \begin{cases} y_i - \langle w, x_i \rangle - b \leq \varepsilon \\ \langle w, x_i \rangle + b - y_i \leq \varepsilon \end{cases} \end{aligned} \tag{2}$$

Źródło: Smola, A. J., & Schölkopf, B. (2004). A tutorial on support vector regression. Statistics and computing, 14(3), 199-222.

SUPPORT VECTOR REGRESSOR

want to allow for some errors. Analogously to the "soft margin" loss function (Bennett and Mangasarian 1992) which was used in SV machines by Cortes and Vapnik (1995), one can introduce slack variables ξ_i, ξ_i^* to cope with otherwise infeasible constraints of the optimization problem (2). Hence we arrive at the formulation stated in Vapnik (1995).

$$\begin{aligned} \text{minimize} \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{\ell}(\xi_i + \xi_i^*) \\ \text{subject to} \quad & \begin{cases} y_i - \langle w, x_i\rangle - b \le \varepsilon + \xi_i \\ \langle w, x_i\rangle + b - y_i \le \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \ge 0 \end{cases} \end{aligned} \tag{3}$$

The constant $C > 0$ determines the trade-off between the flatness of f and the amount up to which deviations larger than ε are tolerated. This corresponds to dealing with a so called ε-insensitive loss function $|\xi|_\varepsilon$ described by

$$|\xi|_\varepsilon := \begin{cases} 0 & \text{if } |\xi| \le \varepsilon \\ |\xi| - \varepsilon & \text{otherwise.} \end{cases} \tag{4}$$

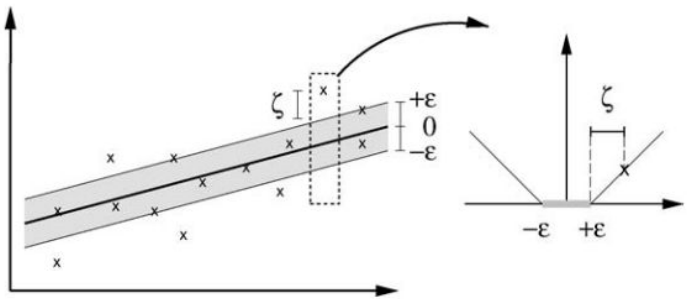


Fig. 1. *The soft margin loss setting for a linear SVM (from Schölkopf and Smola, 2002)*

Źródło: Smola, A. J., & Schölkopf, B. (2004). A tutorial on support vector regression. Statistics and computing, 14(3), 199-222.

DRZEWA REGRESJI

9.2.2 *Regression Trees*

We now turn to the question of how to grow a regression tree. Our data consists of p inputs and a response, for each of N observations: that is, (x_i, y_i) for $i = 1, 2, \ldots, N$, with $x_i = (x_{i1}, x_{i2}, \ldots, x_{ip})$. The algorithm needs to automatically decide on the splitting variables and split points, and also what topology (shape) the tree should have. Suppose first that we have a partition into M regions $R_1, R_2, \ldots, R_M$, and we model the response as a constant c_m in each region:

$$f(x) = \sum_{m=1}^{M} c_m I(x \in R_m). \tag{9.10}$$

If we adopt as our criterion minimization of the sum of squares $\sum(y_i - f(x_i))^2$, it is easy to see that the best $\hat{c}_m$ is just the average of y_i in region R_m:

$$\hat{c}_m = \text{ave}(y_i | x_i \in R_m). \tag{9.11}$$

Now finding the best binary partition in terms of minimum sum of squares is generally computationally infeasible. Hence we proceed with a greedy algorithm. Starting with all of the data, consider a splitting variable j and split point s, and define the pair of half-planes

$$R_1(j, s) = \{X | X_j \leq s\} \quad \text{and} \quad R_2(j, s) = \{X | X_j > s\}. \tag{9.12}$$

Then we seek the splitting variable j and split point s that solve

$$\min_{j,\, s} \left[\min_{c_1} \sum_{x_i \in R_1(j,s)} (y_i - c_1)^2 + \min_{c_2} \sum_{x_i \in R_2(j,s)} (y_i - c_2)^2 \right]. \tag{9.13}$$

For any choice j and s, the inner minimization is solved by

$$\hat{c}_1 = \text{ave}(y_i | x_i \in R_1(j, s)) \quad \text{and} \quad \hat{c}_2 = \text{ave}(y_i | x_i \in R_2(j, s)). \tag{9.14}$$

For each splitting variable, the determination of the split point s can be done very quickly and hence by scanning through all of the inputs, determination of the best pair (j, s) is feasible.

Having found the best split, we partition the data into the two resulting regions and repeat the splitting process on each of the two regions. Then this process is repeated on all of the resulting regions.

Źródło: T. Hastie, R. Tibshirani and J. Friedman. "Elements of Statistical Learning", Springer, 2009.

SIECI NEURONOWE

```
NN_model = Sequential()
# The Input Layer :
NN_model.add(Dense(128, kernel_initializer='normal',input_dim = train.shape[1], activation='relu'))
# The Hidden Layers :
NN_model.add(Dense(256, kernel_initializer='normal',activation='relu'))
NN_model.add(Dense(256, kernel_initializer='normal',activation='relu'))
NN_model.add(Dense(256, kernel_initializer='normal',activation='relu'))
# The Output Layer :
NN_model.add(Dense(1, kernel_initializer='normal',activation='linear'))
# Compile the network :
NN_model.compile(loss='mean_absolute_error', optimizer='adam', metrics=['mean_absolute_error'])
NN_model.summary()
```

TAKE-AWAY MESSAGE

Większość modeli uczenia maszynowego służących do predykcji potrafi rozwiązywać zarówno problemy klasyfikacji, jak i problemy regresji.

02

REGRESJA LINIOWA I REGULARYZACJA

MODEL LINIOWY

- Model liniowy to suma ważona (kombinacja liniowa) zmiennych, która przewiduje wartość zmiennej tłumaczonej (zależnej, wynikowej) na podstawie wartości zmiennych tłumaczących (niezależnych, wejściowych).
- Przykład: przewidywanie cen domów
- Zmienne:
  - wiek domu w latach X_{AGE}
  - wysokość rocznego podatku od nieruchomości X_{TAX}
- Model:

$$\widehat{Y_{PRICE}} = 212000 + 109X_{TAX} - 2000X_{AGE}$$

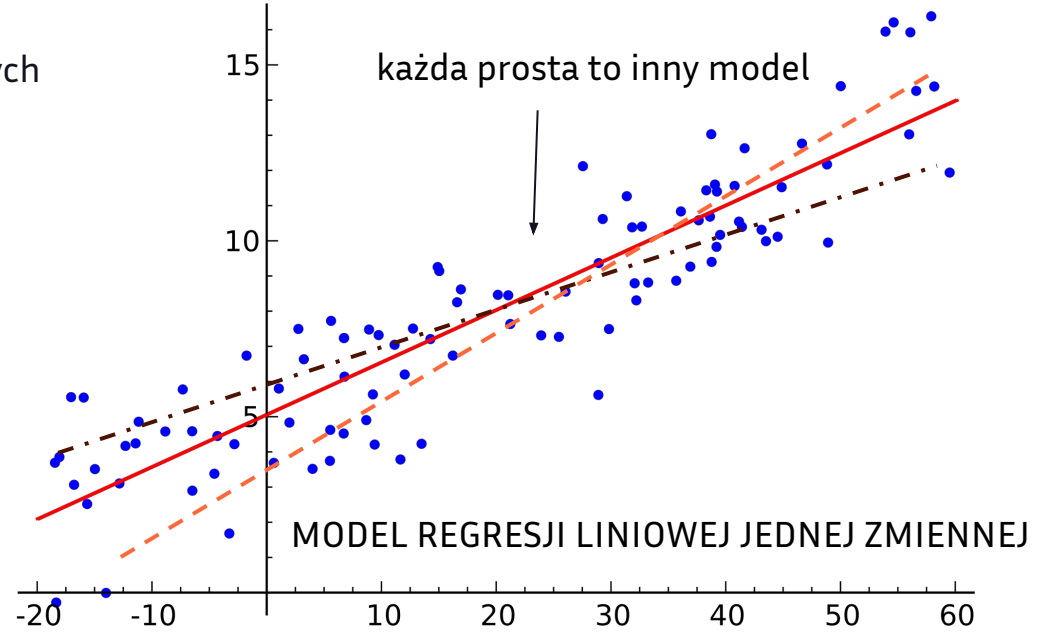
- Przykładowo dla domu opisanego przez krotkę zmiennych tłumaczących
  $(X_{AGE}, X_{TAX}) = (75, 10000)$
  predykowana wartość domu wynosi $\widehat{Y_{PRICE}} = 212000 + 109 \cdot 10000 - 2000 \cdot 75 = 1152000.$

MODEL REGRESJI LINIOWEJ

- w - wagi / współczynniki
- x - wektor zmiennych tłumaczących
- w_0 - wyraz wolny, błąd (ang. bias)

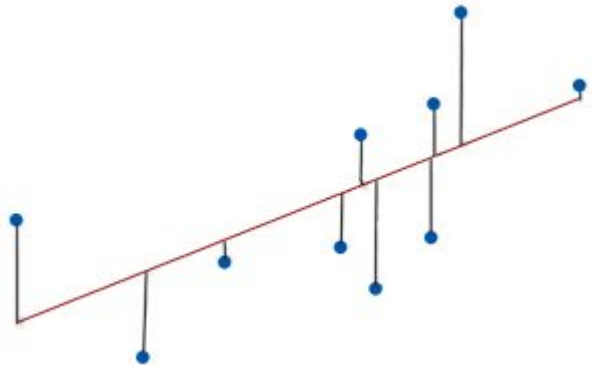
$$\mathbf{x} = (x_1, \ldots, x_n)$$
$$\hat{y} = w_0 + w_1 x_1 + \cdots + w_n x_n$$

$$\mathbf{x} = (x_0, x_1, \ldots, x_n) \text{ gdzie } x_0 = 1$$
$$\hat{y} = w_0 x_0 + w_1 x_1 + \cdots + w_n x_n$$



METODA NAJMNIEJSZYCH KWADRATÓW

- ang. Ordinary Least-Squares (OLS)
- Znajduje wagi w oraz bias b, które minimalizują błąd średniokwadratowy modelu - czyli sumę kwadratów różnic pomiędzy predykowanymi i obserwowanymi (prawdziwymi) wartościami.
  - RSS - residual sum of squares
- Brak parametrów kontrolujących złożoność modelu



$$RSS(\mathbf{w}, b) = \sum_{i=1}^{N} (\hat{y}_i - (\mathbf{w}x_i + b))^2$$

Źródło grafiki: https://statisticsbyjim.com/glossary/ordinary-least-squares/

REGULARYZACJA

- Modele złożone dobrze dopasowują się do danych wyjściowych, ale charakteryzują się dużą zmiennością wartości wyjściowych. Ryzykiem jest nadmierne dopasowanie (overfitting).
- Modele prostsze są obciążone dużym błędem systematyczny (bias) i ich zastosowanie niesie ryzyko niewystarczającego dopasowania (underfitting).
- Regularyzacja przeciwdziała nadmiernemu dopasowaniu, poprzez penalizowanie nadmiernie skomplikowanych modeli.
- Wpływ regularyzacji R jest kontrolowany przez parametr λ.
- Im wyższe λ tym silniejsza regularyzacja i prostszy model.
- Modele z regularyzacją wymagają przeskalowania zmiennych do przedziału [0,1] (MinMaxScaler).

$$RSS(\mathbf{w}, b) = \sum_{i=1}^{N} (\hat{y}_i - (\mathbf{w}x_i + b))^2 + \lambda R$$

REGRESJA LINIOWA Z REGULARYZACJĄ

$$RSS(\mathbf{w}, b) = \sum_{i=1}^{N} (\hat{y}_i - (\mathbf{w}x_i + b))^2 + \lambda R$$

Regresja Grzbietowa
ang. ridge regression
regularyzacja L_2

$$RSS(\mathbf{w}, b) = \sum_{i=1}^{N} (\hat{y}_i - (\mathbf{w}x_i + b))^2 + \lambda \sum_{j=1}^{p} w_j^2$$

Regresja LASSO
ang. LASSO regression
regularyzacja L_1

$$RSS(\mathbf{w}, b) = \sum_{i=1}^{N} (\hat{y}_i - (\mathbf{w}x_i + b))^2 + \lambda \sum_{j=1}^{p} |w_j|$$

03

DOBROĆ REGRESORA

WSPÓŁCZYNNIK DETERMINACJI R^2

- Współczynnik determinacji R^2 – miara jakości dopasowania modelu do danych uczących.
- Informuje o tym, jaka część zmienności (wariancji) zmiennej objaśnianej w próbie pokrywa się z korelacjami ze zmiennymi zawartymi w modelu.
- Współczynnik R^2 ma jasną interpretację tylko wtedy, gdy współczynniki modelu regresji zostały wyestymowane metodą najmniejszych kwadratów i w modelu występuje wyraz wolny. Wówczas $0<=R^2<=1$ i R^2 można interpretować jako miarę dopasowania modelu do danych.

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=1}^{N} (y_i - \hat{y_i})^2}{\sum_{i=1}^{N} (y_i - \bar{y_i})^2}$$

BŁĄD ŚREDNIOKWADRATOWY

- Błąd średniokwadratowy penalizuje obserwacje leżące daleko od krzywej regresji.
- Jednostką jest kwadrat jednostki podstawowej, co może być problematyczne w interpretacji.
- ang. Mean Squared Error

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

PIERWIASTEK BŁĘDU ŚREDNIOKWADRATOWEGO

- Jednostka błędu jest spójna z jednostką podstawową.
- RMSE to pierwiastek z MSE
- ang. Root Mean Squared Error

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}$$

ŚREDNI BŁĄD BEZWZGLĘDNY

- Jednostka błędu jest spójna z jednostką podstawową.
- Metryka bardziej odporna na obserwacje odstające niż MSE (czy RMSE).
- ang. Mean Absolute Error

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

04

WALIDACJA KRZYŻOWA

PODZIAŁ ZBIORU DANYCH

	fruit_label	fruit_name	fruit_subtype	mass	width	height	color_score
0	1	apple	granny_smith	192	8.4	7.3	0.55
1	1	apple	granny_smith	180	8.0	6.8	0.59
2	1	apple	granny_smith	176	7.4	7.2	0.60
3	2	mandarin	mandarin	86	6.2	4.7	0.80
4	2	mandarin	mandarin	84	6.0	4.6	0.79
...	...	...	...	...	...	...	...
54	4	lemon	unknown	116	6.1	8.5	0.71
55	4	lemon	unknown	116	6.3	7.7	0.72
56	4	lemon	unknown	116	5.9	8.1	0.73
57	4	lemon	unknown	152	6.5	8.5	0.72
58	4	lemon	unknown	118	6.1	8.1	0.70

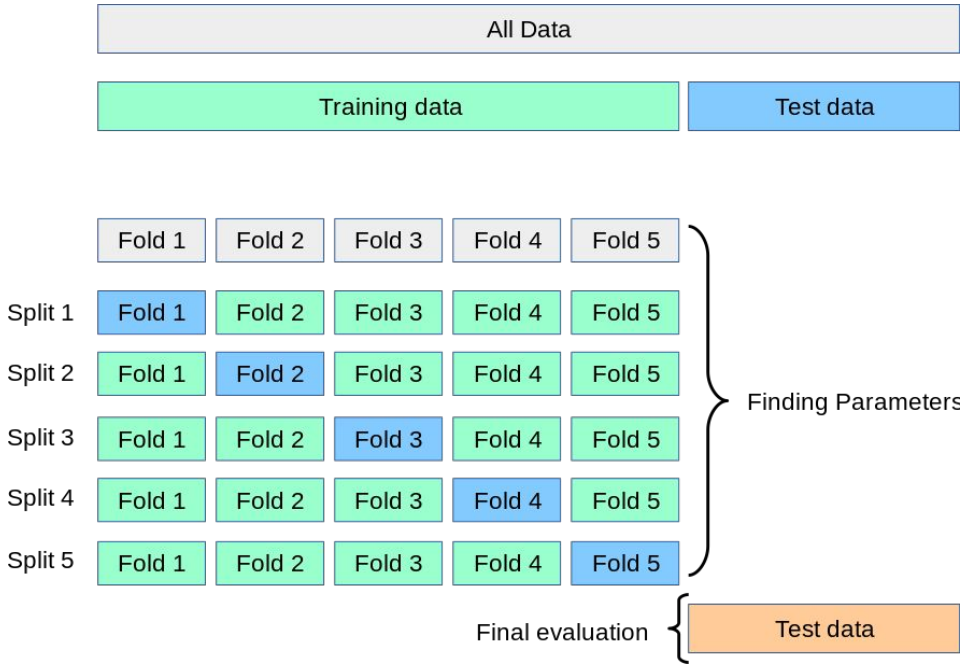
ZBIÓR TESTOWY

- Uczenie się parametrów funkcji przewidywania i testowanie jej na tych samych danych jest błędem metodologicznym
    - model, który po prostu odtwarzałby etykiety próbek, które właśnie zobaczył, miałby doskonały wynik, ale nie przewidziałby niczego użytecznego na niewidzianych wcześniej danych.
- Aby tego uniknąć, powszechną praktyką podczas przeprowadzania (nadzorowanego) eksperymentu uczenia maszynowego jest przechowywanie części dostępnych danych jako zestawu testów X_test, y_test.

ZBIÓR WALIDACYJNY

- Podczas oceny różnych hiperparametrów (ustawień) dla estymatorów, nadal istnieje ryzyko przeuczenia zestawu testowego, ponieważ parametry można modyfikować, aż estymator będzie działał optymalnie.
    - W ten sposób wiedza o zestawie testów może „przeciekać” do modelu, a metryki ewaluacyjne nie informują już o wydajności generalizacji.
- Aby rozwiązać ten problem, jeszcze jedną część zbioru danych można przedstawić jako tak zwany „zbiór walidacyjny”:
    - uczenie przebiega na zbiorze uczącym,
    - ocena przeprowadzana jest na zbiorze walidacyjnym,
    - kiedy eksperyment wydaje się pomyślny, ostatecznej oceny można dokonać na zestawie testowym.
- Dzieląc dostępne dane na trzy zestawy, drastycznie zmniejszamy liczbę próbek.

WALIDACJA KRZYŻOWA (CROSSVALIDATION)



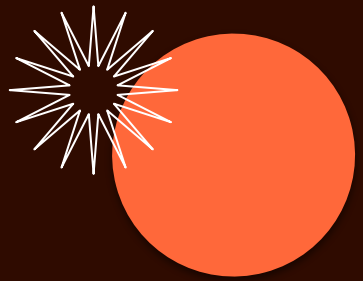
WALIDACJA KRZYŻOWA

- Zbiór testowy nadal powinien być trzymany do ostatecznej oceny.
- Zbiór do walidacji nie jest już potrzebny podczas tworzenia CV (walidacji krzyżowej).
- W walidacji krzyżowej:
  - model jest uczony przy użyciu k-1 podzbiorów danych treningowych
  - wynikowy model jest weryfikowany na pozostałej części danych (tj. jest używany jako zestaw testów do obliczania miary wydajności, takiej jak dokładność).
- Wyjściowa miara jakości modelu z k-krotnej walidacji krzyżowej jest średnią z k wartości.

WYCIEKI DANYCH

- W 2013 roku w konkursie uczenia maszynowego przyznano nagrodę za najdokładniejsze wykrywanie odgłosów wieloryba biskajskiego na podstawie danych dźwiękowych.
- Organizatorzy wkrótce odkryli problemy z wyciekiem danych w pierwszej wersji zestawu danych.
- Artykuł wyjaśnia, co się stało. Ten krótki, ale interesujący artykuł, stanowi doskonały przykład tego, jak może nastąpić wyciek danych i jak można sobie z nim poradzić.

https://www.kaggle.com/c/the-icml-2013-whale-challenge-right-whale-redux/discussion/4865#25839#post25839



05

DEMO

THANKS!

DZIĘKUJĘ ZA UWAGĘ