Metody eksploracji danych
3. Ocena modeli

Piotr Szwed
Katedra Informatyki Stosowanej AGH
2017

Zagadnienie regresji

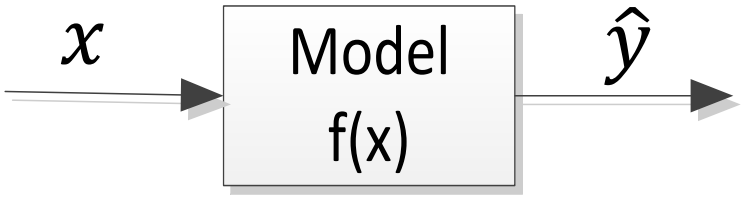
Dane:

- Zbiór uczący: $\mathrm{D} = \{(x_i, y_i)\}_{i=1,m}$
- Obserwacje: (x_i, y_i), wektor cech $x_i \in \boldsymbol{R}^n$
- Wartość wyjściowa jest skalarem $y_i \in \boldsymbol{R}$

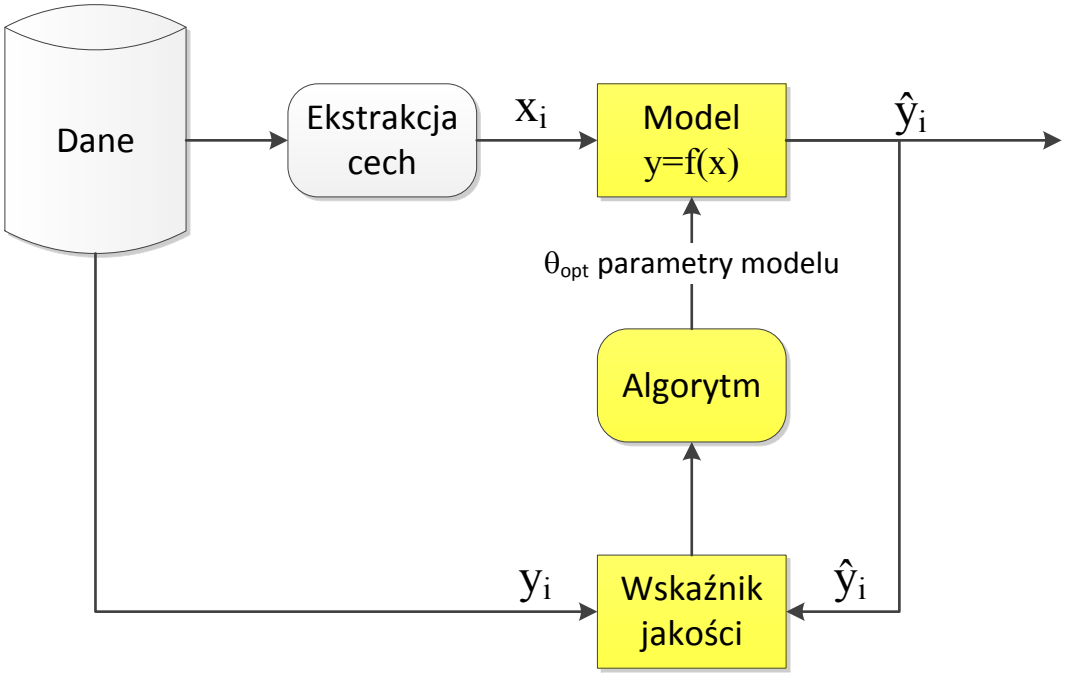
Zadanie: dobór funkcji

$f(x)$: $\boldsymbol{R}^n \rightarrow \boldsymbol{R}$,

która pozwoli **przewidzieć** wartość wyjściową $\hat{y} = f(x)$ odpowiadającą x



Przebieg procesu uczenia



- Wskaźnik jakości służy do porównania:
  - y_i- wartości wyjściowej zapisanych w zbiorze uczącym z
  - $\hat{y}_i$- wartości przewidywanej przez model dla x_i
- Wskaźnik jakości steruje przebiegiem algorytmu lub jest wykorzystywany w jego konstrukcji

Wielkości statystyczne

- Wartość średnia dla próby losowej $Y = \{y_1, y_2, \dots, y_m\}$
  $\bar{y} = \frac{1}{m}\sum_{i=1}^{m} y_i$. Kiedy $m \to \infty, \bar{y} \to E[Y]$
- Wariancja dla próby losowej

$$\sigma^2(Y) = \frac{1}{m}\sum_{i=1}^{m}(y_i - \bar{y})^2$$

$$E(Y - E[Y])^2 = E[Y^2] - E[Y]^2$$

- Kowariancja dla dwóch prób losowych $X = \{x_1, x_2, \dots, x_m\}$ i $Y = \{y_1, y_2, \dots, y_m\}$

$$\sigma(X, Y) = \frac{1}{m}\sum_{i=1}^{m}(x_i - \bar{x})(y_i - \bar{y})$$

$$E(X - E[X])(Y - E[Y]) = E[XY] - E[X]E[Y]$$

Współczynnik korelacji Pearsona

$$r(X,Y) = \frac{\sigma(X,Y)}{\sigma(X)\sigma(Y)} = \frac{\sum_{i=1}^{m}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{m}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{m}(y_i - \bar{y})^2}}$$

Współczynnik korelacji Persona r ma następującą własność:

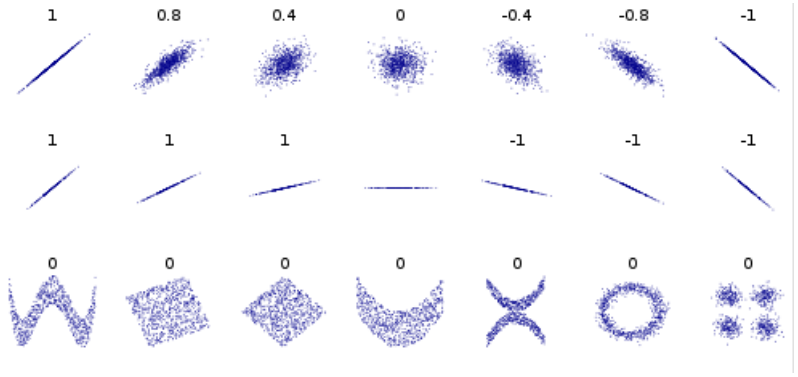
$$r(X,Y) = r(aX + b, cY + d)$$

czyli skalowanie lub przesuwanie wartości nie zmienia korelacji próbek.

Jeżeli próbki są przesunięte tak, aby $\bar{x} = 0$ i $\bar{y} = 0$, wówczas $r(X,Y)$ może być interpretowane jako iloczyn skalarny dwóch m-wymiarowych wektorów x i y podzielonych przez ich długości (normy).

$$\cos\varphi = \frac{x \cdot y}{|x||y|}$$

Odpowiada to kosinusowi kąta pomiędzy wektorami. Wartość 1 lub -1 oznacza, że próbki są silnie skorelowane. Wartość bliska 0 – brak korelacji.

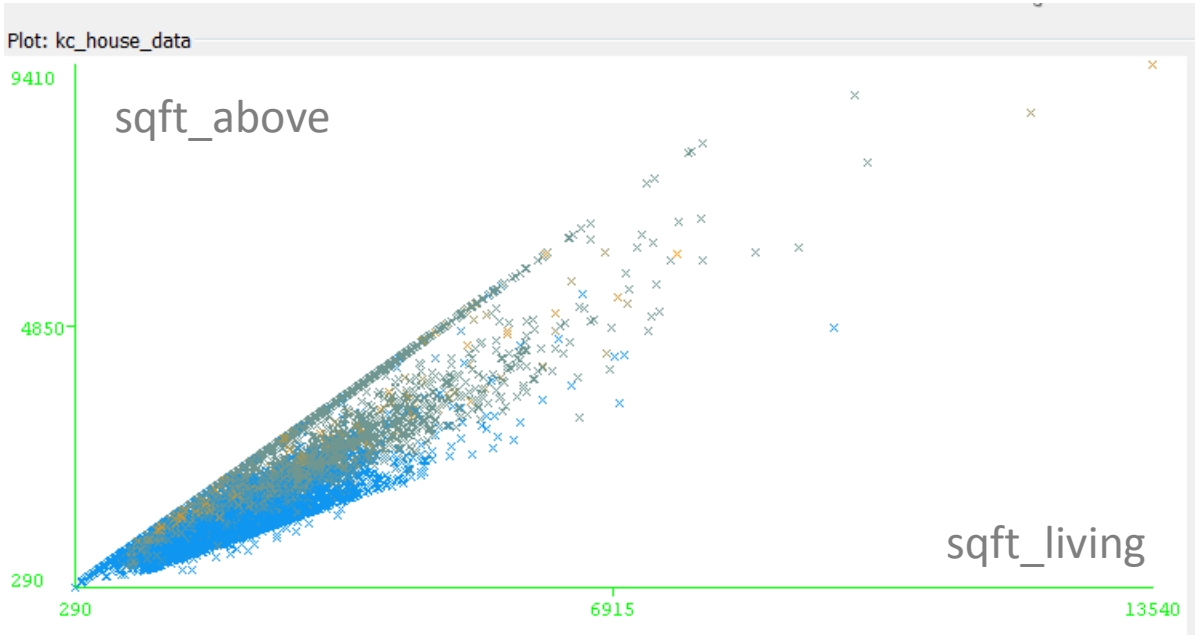


Przykłady korelacji [https://en.wikipedia.org/wiki/Pearson_product-moment_correlation_coefficient]

Współczynnik korelacji Pearsona - zastosowania

- Jeżeli dwa atrybuty A_k i A_l odpowiadające zmiennym $x[k]$ i $x[l]$ zbioru uczącego są silnie skorelowane, wówczas zapewne jedną z nich można odrzucić i zmniejszyć złożoność modelu.
- Idealna korelacja 1/-1 mogłaby być przyczyną błędów numerycznych
- Przykład: dla zbioru kc_house_data

$$r(sqft\_living, sqft\_above) = 0.8765$$

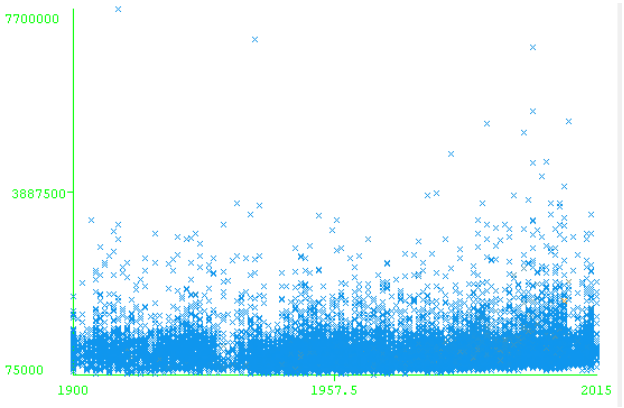


sqft_living - Square footage of the apartments interior living space

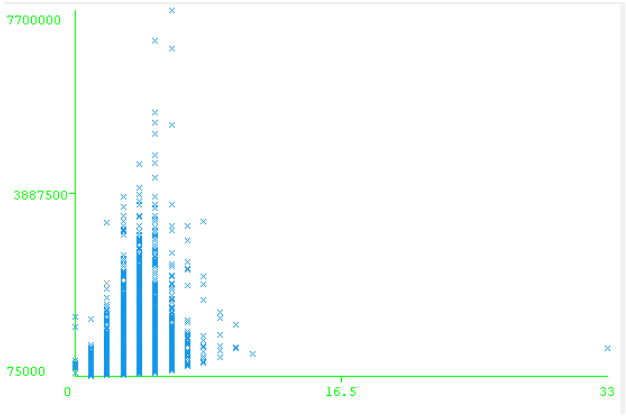
sqft_above - The square footage of the interior housing space that is above ground level

Współczynnik korelacji Pearsona - zastosowania

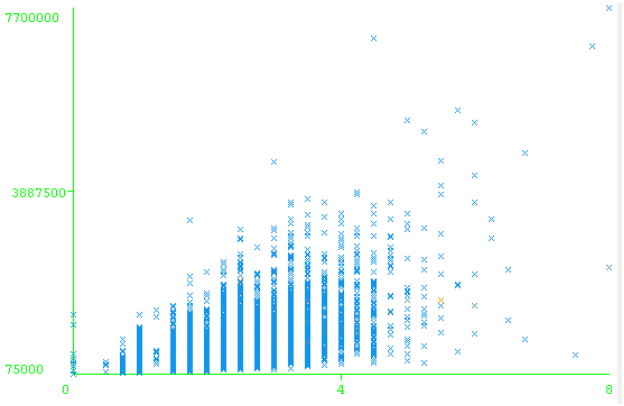
Korelacja pomiędzy atrybutem A_k (odpowiadającym zmiennej $x[k]$) i zmienną wyjściową y uzasadnia użycie atrybutu w modelu



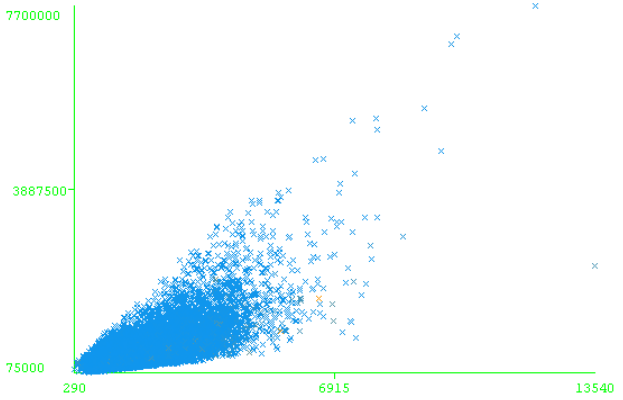
$r(yr\_built, price) = 0.054$



$r(bedrooms, price) = 0.308$



$r(bathrooms, price) = 0.525$



$r(sqft\_living, price) = 0.70$

Ocena jakości predykcji

Dla zbioru uczącego $\mathrm{D} = \{(x_i, y_i)\}_{i=1,m}$ i modelu $f(x)$, niech $\hat{y}_i = f(x)$.

Współczynnik Pearsona może zostać obliczony dla próbek y i $\hat{y}$ dla oceny korelacji pomiędzy obserwacjami zmiennych wyjściowych y oraz $\hat{y}$.

Współczynnik determinacji (ang. coefficient of determination)

W praktyce do oceny jakości modelu zamiast $r(y, \hat{y})$ używa się $r^2(y, \hat{y})$. Współczynnik może być obliczony przez podniesienie $r(y, \hat{y})$ do kwadratu lub wprost ze wzoru:

$$r^2 = \frac{\sum_{i=1}^{m}(\hat{y}_i - \bar{y})^2}{\sum_{i=1}^{m}(y_i - \bar{y})^2}$$

- Współczynnik determinacji mierzy jakość dopasowania regresji.
  - $r^2 \approx 1$ – dobre dopasowanie
  - $r^2 \approx 0.5$ – słabe dopasowanie
  - $r^2 \approx< 0.5$ – dopasowanie niezadawalające
- Wartość $1 - r^2$ określa jaka część zmienności wartości wyjściowej (zmiennej objaśnianej) nie jest „wyjaśniona” przez atrybuty użyte w regresji (zmienne objaśniające).
- Więcej informacji: Daniel T. Larose Data Mining Methods And Models/Metody i modele eksploracji danych

Python

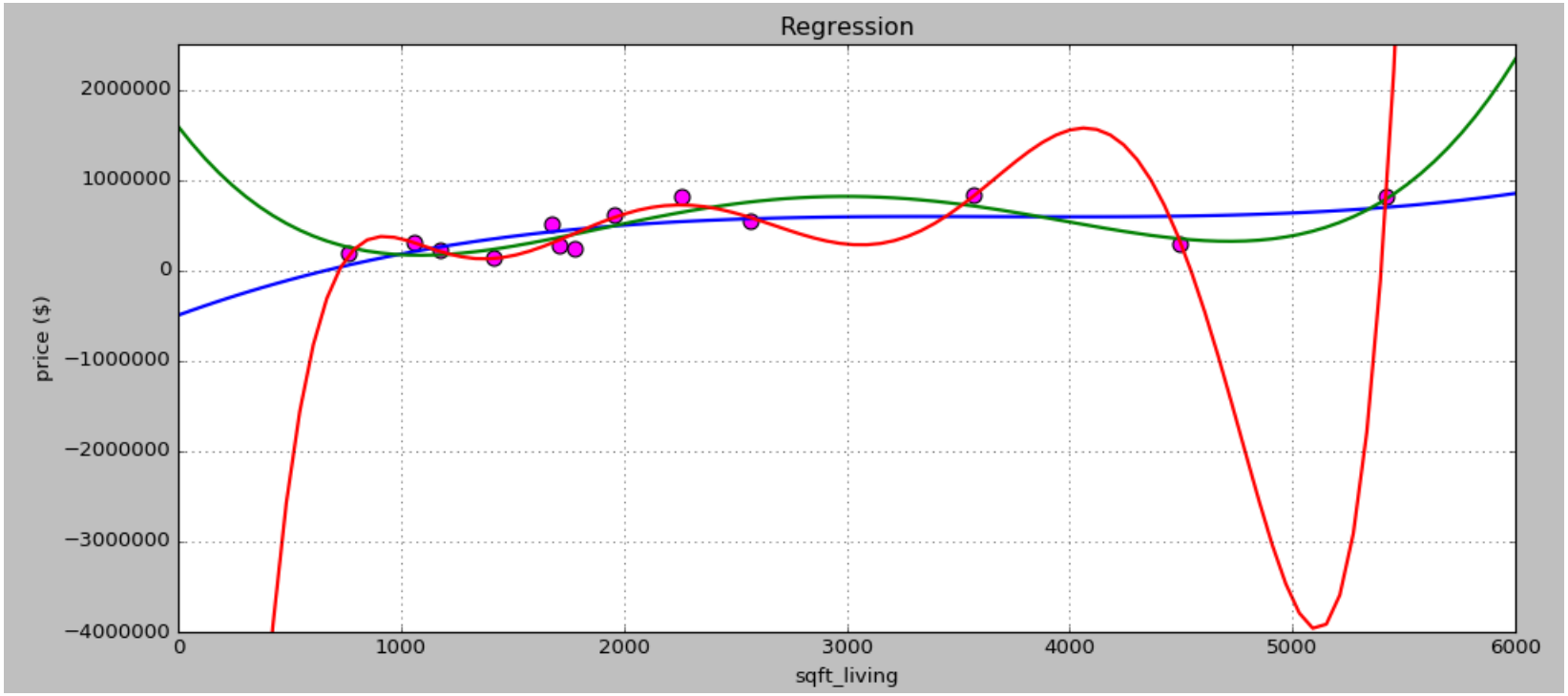
Klasa LinearRegression (oraz Ridge i Lasso) mają metodę score() wyznaczającą współczynnik determinacji r^2. Parametrami są $m \times n$ macierz X oraz wektor y.

```
from sklearn import linear_model
…
x, y = np.loadtxt(inp, delimiter='\t', usecols=(0, 1), unpack=True,skiprows=0)

features3 = np.stack((x,np.power(x,2),np.power(x,3)),axis=-1)
regr3 = linear_model.LinearRegression()
regr3.fit(features3, y)
print (regr3.score(features3,y))

fx=np.linspace(0,6000,100)
fy = regr3.predict(np.stack((fx,np.power(fx,2),np.power(fx,3)),axis=-1))
plt.plot(fx,fy,linewidth=2,color='b',label='3 st')
```

Przykład



Wielomian	Pearson: r	Współczynnik determinacji r^2
3 stopnia (niebieski)	0.6953	0.48349
4 stopnia (zielony)	0.8603	0.7402
7 stopnia (czerwony)	0.9410	0.8855

Metryki oceny modelu: MSE

Średni błąd kwadratowy (mean squared error)

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

Względny średni błąd kwadratowy

$$RelMSE = \frac{\sum_{i=1}^{m}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{m}(y_i - \bar{y})^2} = \frac{\sum_{i=1}^{m}(y_i - \hat{y}_i)^2}{\sigma^2}$$

Zauważmy, że $MSE = \frac{1}{m}RSS$. (RSS jest funkcją celu minimalizowaną podczas regresji).

Funkcje oceny modelu: MSE

Metryka MSE jest chętnie stosowana w każdej dziedzinie, gdzie ocenę można przeprowadzić przez porównanie dwóch ciągów wartości, np. ocena jakości modelu, ocena jakości i wierności sygnału.

- jest prosta do obliczenia
- jest miarą odległości:
  - nieujemna $MSE(X,Y) \geq 0$
  - zachowuje identyczność $MSE(X,Y) = 0$ jeśli $X = Y$
  - symetryczna $MSE(X,Y) = MSE(Y,X)$
  - nierówność trójkąta: $MSE(X,Z) \leq MSE(X,Y) + MSE(Y,Z)$
- może mieć interpretacją fizyczną (np. jako energia błędu sygnału)
- idealna dla optymalizacji
- w wielu dziedzinach jest używana z przyzwyczajenia

Funkcje oceny modelu: MSE

Zastosowanie MSE do oceny szeregów czasowych (sygnałów) bywa krytykowane.

[FIG2] **Comparison of image fidelity measures for "Einstein" image altered with different types of distortions. (a) Reference image. (b) Mean contrast stretch. (c) Luminance shift. (d) Gaussian noise contamination. (e) Impulsive noise contamination. (f) JPEG compression. (g) Blurring. (h) Spatial scaling (zooming out). (i) Spatial shift (to the right). (j) Spatial shift (to the left). (k) Rotation (counter-clockwise). (l) Rotation (clockwise).**

[Zhou Wang and Alan C. Bovik: Mean Squared Error: Love It or Leave It? IEEE SIGNAL PROCESSING MAGAZINE **[98]** JANUARY 2009]

Funkcje oceny modelu: RMSE

$RMSE$ (root mean square error) jest zdefinowana jako:

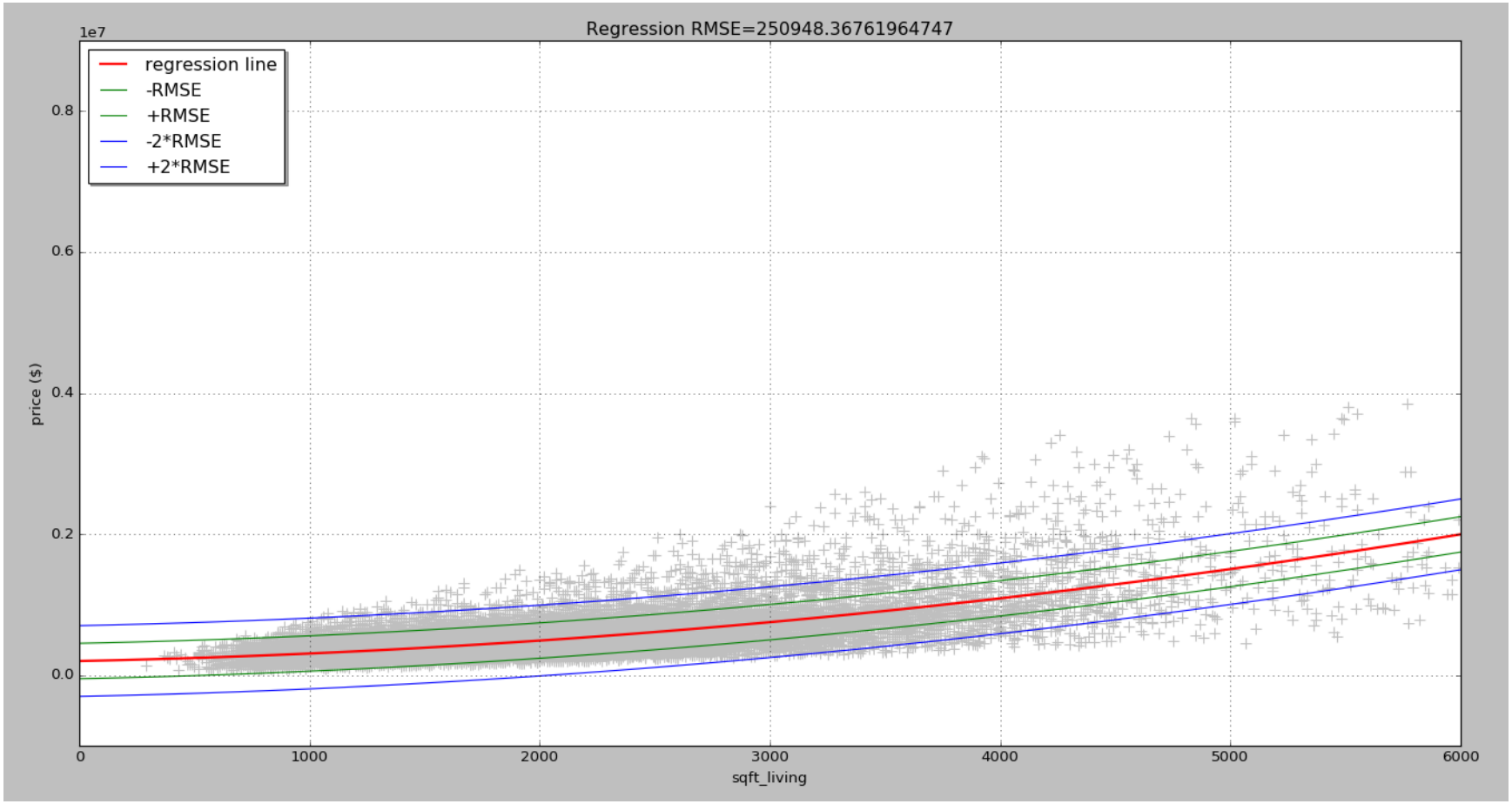
$$RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2}$$

Podobnie można zdefiniować względną wartość:

$$RelRMSE = \sqrt{\frac{\sum_{i=1}^{m}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{m}(y_i - \bar{y})^2}}$$

- RMSE jest wyrażana w jednostkach zmiennych wyjściowych
- Jest też nazywana **standardowym błędem regresji/estymaty**
- Pozwala określić przedziały ufności. Jeśli $\hat{y}$ jest estymowaną wartością dla x, to „prawdziwa" wartość $f_{true}(x)$:
  - Z prawdopodobieństwem 0.68 jest w przedziale $[\hat{y} - RMSE, \hat{y} + RMSE]$
  - Z prawdopodobieństwem 0.95 jest w przedziale $[\hat{y} - 2RMSE, \hat{y} + 2RMSE]$
- RMSE może być wrażliwa na przypadkowe duże wartości y

Przykład interpretacji RMSE



Zbiór kc_house_data

Funkcje oceny modelu wykorzystujące wartości bezwzględne

Średni błąd bezwględny MAE (mean absolute error)

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

Średni błąd względny (relative mean absolute error)

$$RelMAE = \frac{\sum_{i=1}^{m}|y_i - \hat{y}_i|}{\sum_{i=1}^{m}|y_i - \bar{y}|}$$

Średni bezwzględny błąd procentowy MAPE (mean absolute percentage error)

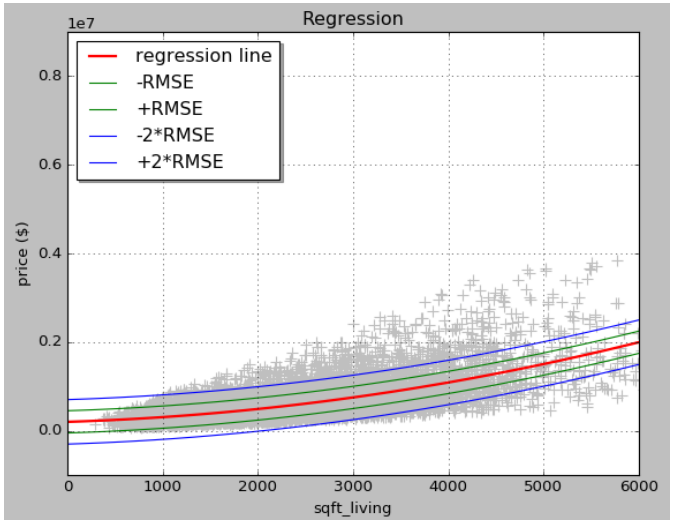
$$MAPE = \frac{100}{m}\sum_{i=1}^{m}\left|\frac{y_i - \hat{y}_i}{y_i}\right|$$

Funkcje oceny modelu wykorzystujące wartości bezwzględne

- Mediana błędu bezwzględnego (median absolute error)

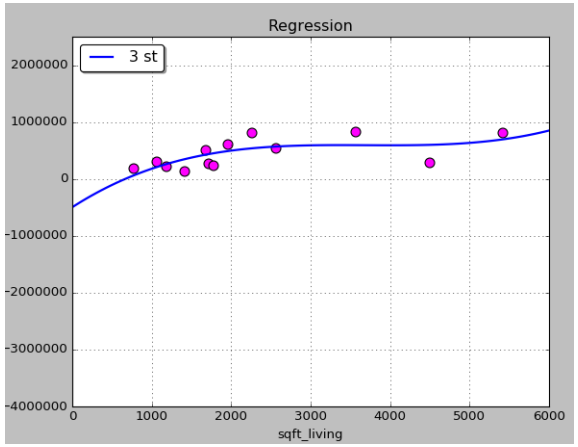
$$MedianAE = median(\{|y_i - \hat{y}_i| : i = 1, \dots, m\})$$

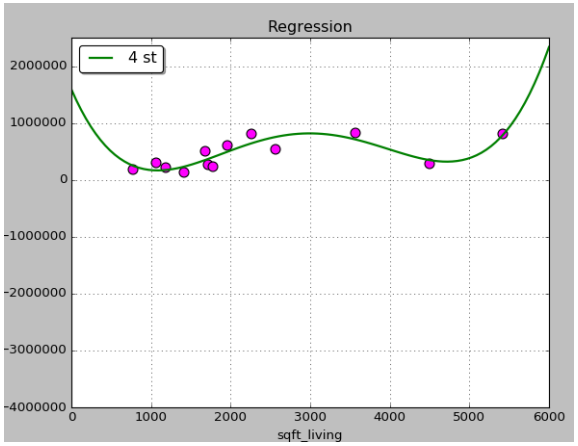
- Miary wykorzystujące wartości bezwzględne są mniej wrażliwe na obserwacje z przypadkowymi dużymi wartościami (nie są podnoszone do kwadratu).
- Miara medianowa jest szczególnie odporna
- Na ogół MAE < RMSE (błąd absolutny mniejszy od pierwiastka kwadratowego)

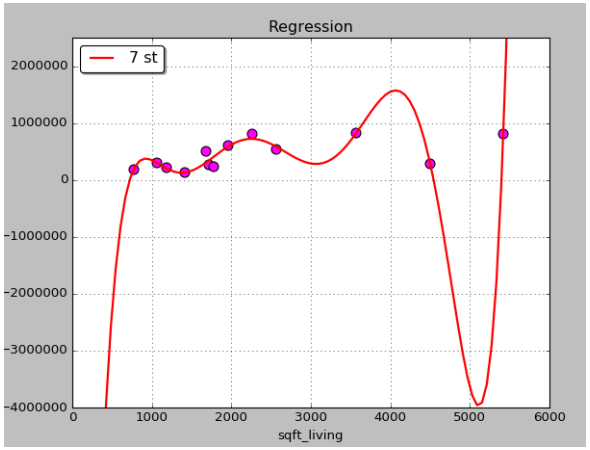


$MSE = 62975083210.97\ \$^2$
$RelMSE = 0.46726$
$RMSE = 250\,948.37\ \$$
$RelRMSE = 0.68356$
$MAE = 165\,737.57\ \$$
$RelMAE = 0.70846$
$MAPE = 34.73\%$
$MedianAE = 122\,900.81\ \$$

Porównanie – która krzywa jest najlepsza?





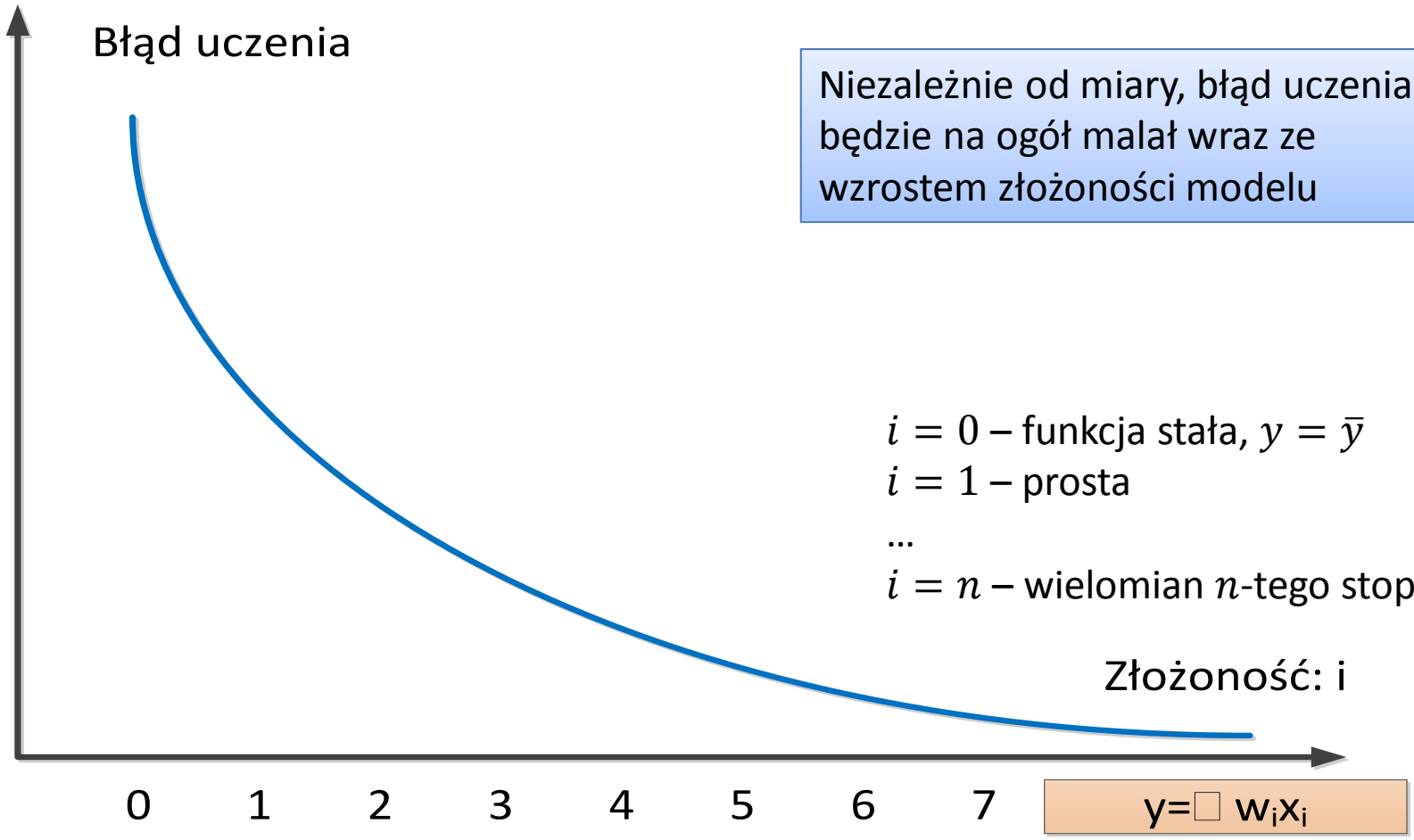


3 stopień
MSE= 31 440 814 308.86
RelMSE= 0.51650
RMSE= 177 315.58
RelRMSE= 0.71868
MAE= 154 739.49
RelMAE= 0.05285
MAPE= 52.89
MedAE= 119 875.08

4 stopień
MSE= 15 810 127 730.65
RelMSE= 0.25972
RootMSE= 125 738.33
RelRootMSE= 0.50963
MAE= 113 498.98
RelMAE= 0.03877
MAPE= 35.32789
MedAE= 111 812.74

7 stopień
MSE= 6 965 504 486.83
RelMSE= 0.11443
RootMSE= 83 459.60
RelRootMSE= 0.33827
MAE= 50 143.26
RelMAE= 0.01713
MAPE= 14.73982
MedAE= 16 473.52062

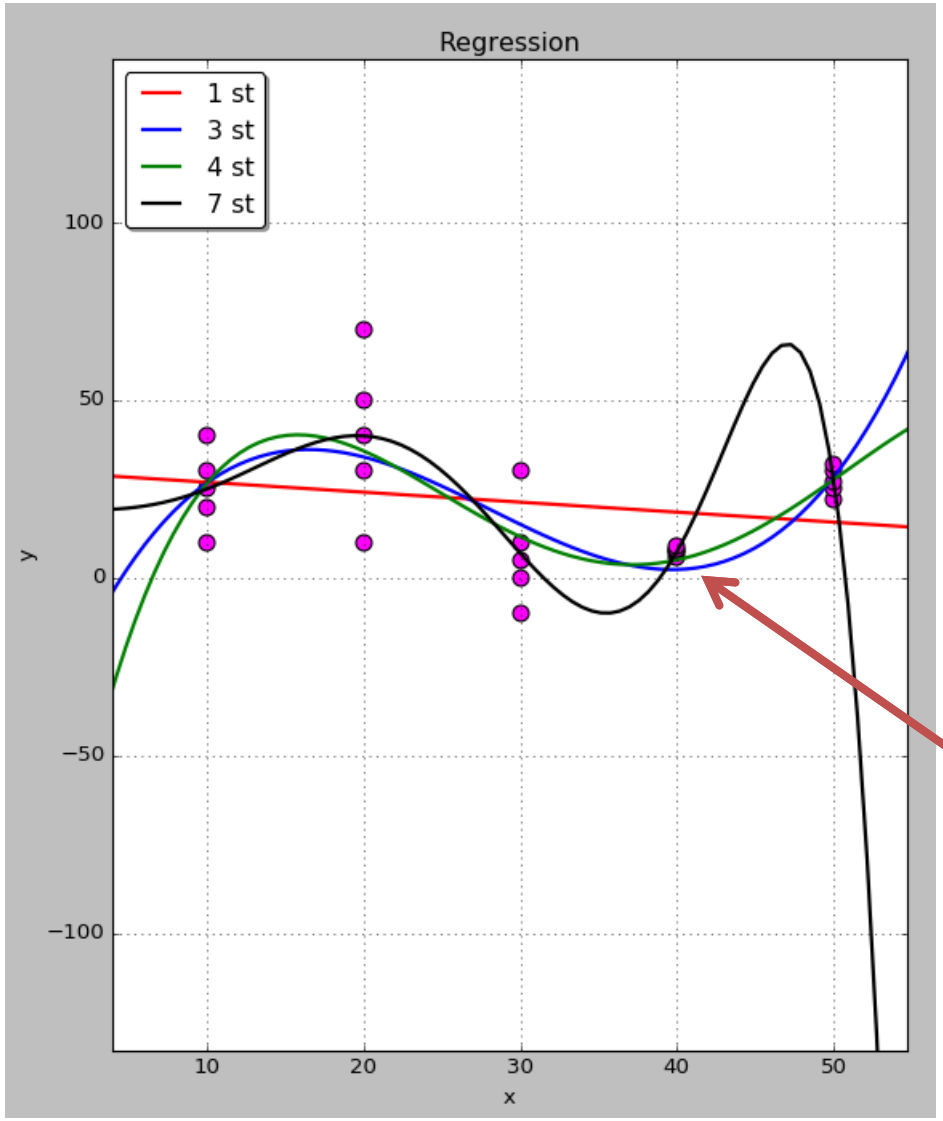
Błąd uczenia



Niezależnie od miary, błąd uczenia będzie na ogół malał wraz ze wzrostem złożoności modelu

$i = 0$ – funkcja stała, $y = \bar{y}$
$i = 1$ – prosta
…
$i = n$ – wielomian n-tego stopnia

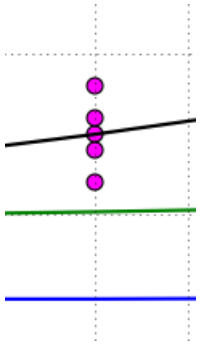
Czy błąd uczenia można zredukować do zera?



Model	MSE
1 stopnia	280.74
3 stopnia	164.23
4 stopnia	147.38
7 stopnia	137.91

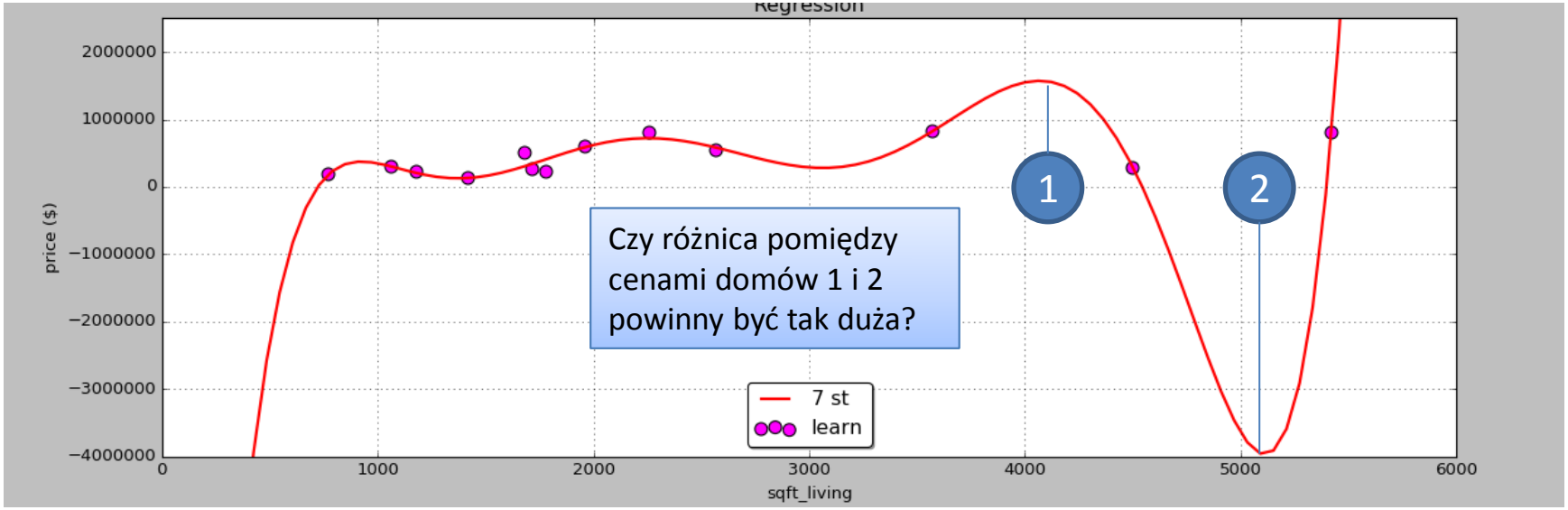
W ogólnym przypadku nie

Jeżeli zbiór uczący zawiera punkty przypisujące tym samym x różne wartości wyjściowe y, błąd uczenia zawsze pozostanie



Zastosowanie modelu do predykcji

- Podczas predykcji model jest stosowany, aby przewidzieć wartość wyjściową dla nieznanych danych.
- Czy błąd uczenia pozwala ocenić jakość modelu podczas predykcji?



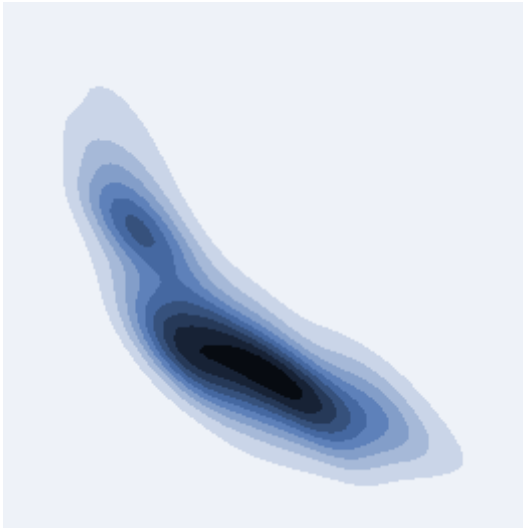
- Błąd uczenia jest bardzo optymistycznym oszacowaniem, ponieważ w wyniku optymalizacji krzywa została dopasowana do danych uczących.
- **Niska wartość błędu uczenia nie pociąga za sobą dobrych własności predykcyjnych**
  (chyba, że model był uczony na wszystkich kombinacjach danych wejściowych)

Błąd generalizacji

Niech f_n będzie modelem wyznaczonego na podstawie zbioru uczącego D.
Błąd generalizacji jest to wartość oczekiwana błędu dla **nieznanych** danych.

$$\mathcal{L}(f_n) = E_{xy}[L(f_n(x), y)] = \int_{X \times \mathrm{Y}} L(f_n(x), y)\, p(x, y) dx dy$$

- $\mathcal{L}(f_n)$ to wartość oczekiwana funkcji oceny $L(\cdot,\cdot)$ dla wszystkich kombinacji wartości wejściowych i wyjściowych ze zbioru $X \times \mathrm{Y}$.
- Wartość oczekiwana musi także uwzględniać rozkład prawdopodobieństwa $p(x, y)$ wystąpienia pary (x, y).
- Dla rzeczywistych danych, a nie np. danych wygenerowanych sztucznie, **rozkład ten jest nieznany**.



Błąd generalizacji

- Najczęściej błąd generalizacji jest definiowany jako „prawdziwy błąd”

$$G = \mathcal{L}(f_n) = true\ error$$

Wówczas pojęcia błędu generalizacji G i prawdziwego błędu $\mathcal{L}(f_n)$ używane są zamiennie

- Inne podejście
  - Dla modelu f_n i zbioru danych D może zostać wyznaczony błąd uczenia (empiryczny):

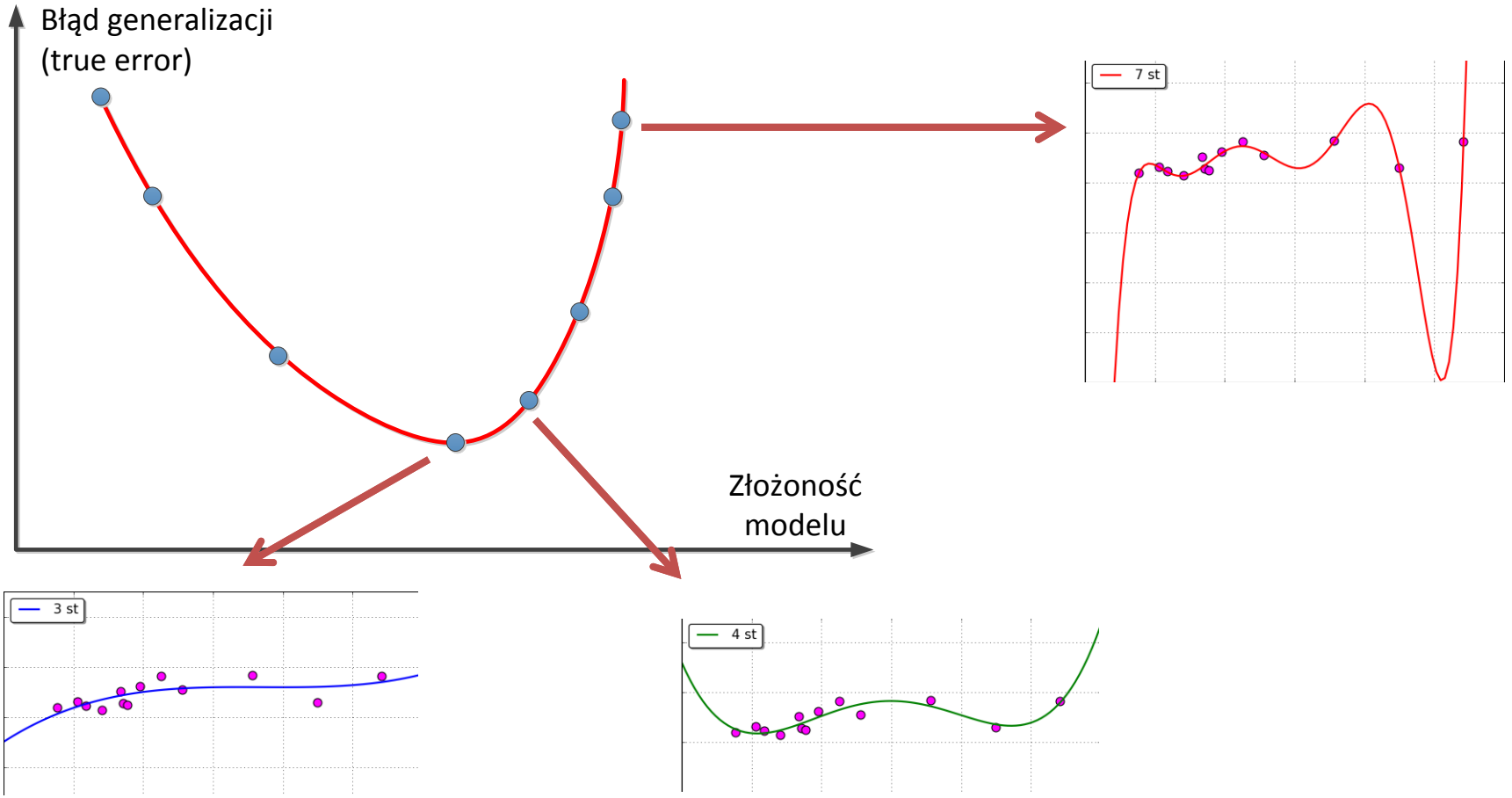
$$\mathcal{L}_D(f_n) = \frac{1}{m}\sum_{i=1}^{m} L(f_n(x_i), y_i)$$

  - Błąd generalizacji modelu f_n jest zdefiniowany jako:

$$G = \mathcal{L}(f_n) - \mathcal{L}_D(f_n) = true\ error - learning\ error$$

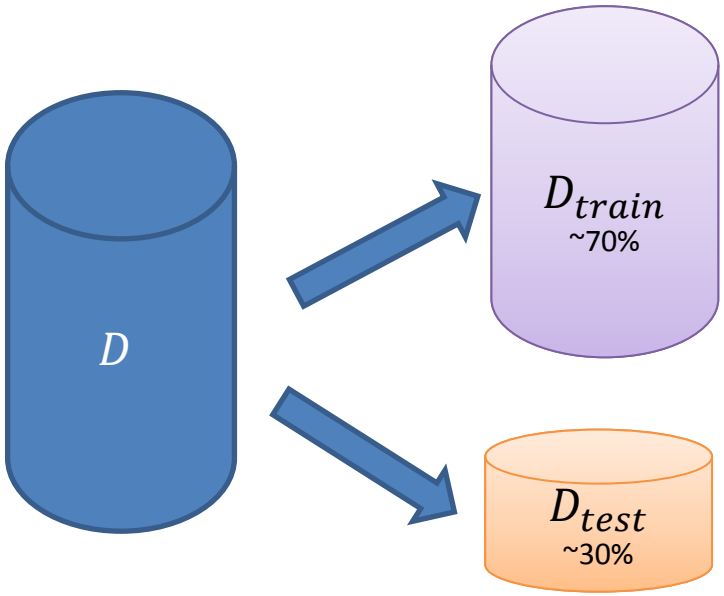
Błąd generalizacji

Na ogół wraz ze wzrostem złożoności „prawdziwy błąd” modelu początkowo maleje, a następnie rośnie.



Błąd testowy

- Prawdziwy błąd $\mathcal{L}(f_n)$ jest na ogół niemożliwy do wyznaczenia, ale można go przybliżyć testując model f_n na nieznanych danych.
- W tym celu dostępny zbiór danych dzielony jest na dwa **rozłączne** podzbiory:
  - D_{train} – dane użyte do uczenia
  - D_{test} – dane testowe

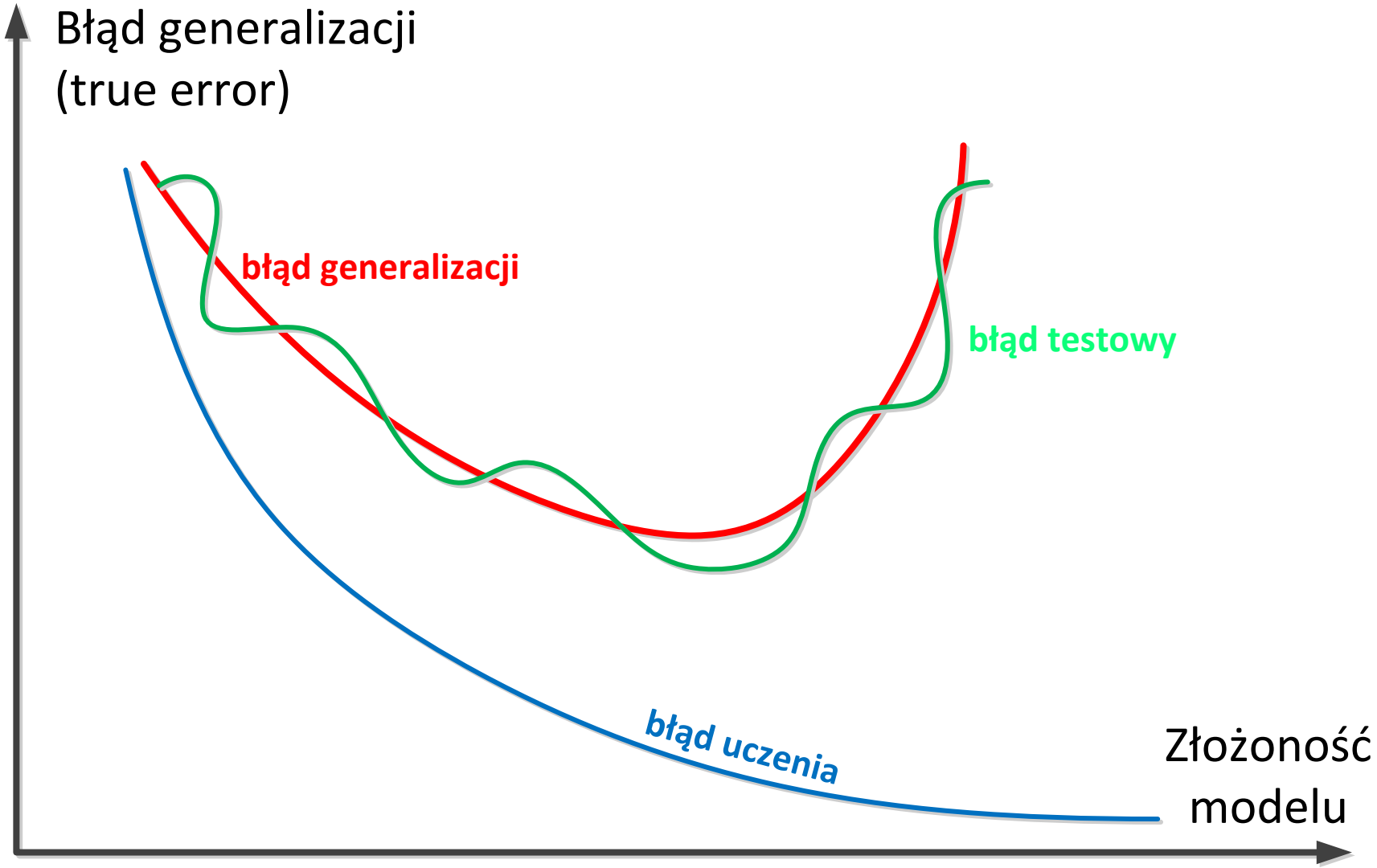


- Model jest uczony dla obserwacji ze zbioru D_{train}
- Obserwacje ze zbioru D_{train} mogą być użyte do oceny błędu uczenia
- Podczas oceny używa się wyłącznie obserwacji w zbiorze D_{test}

$$\mathcal{L}_{test}(f_n) = \frac{1}{k}\sum_{i=1}^{k} L(f_n(x_i), y_i)$$

- Oczekuje się, że $\mathcal{L}(f_n) \sim \mathcal{L}_{test}(f_n)$

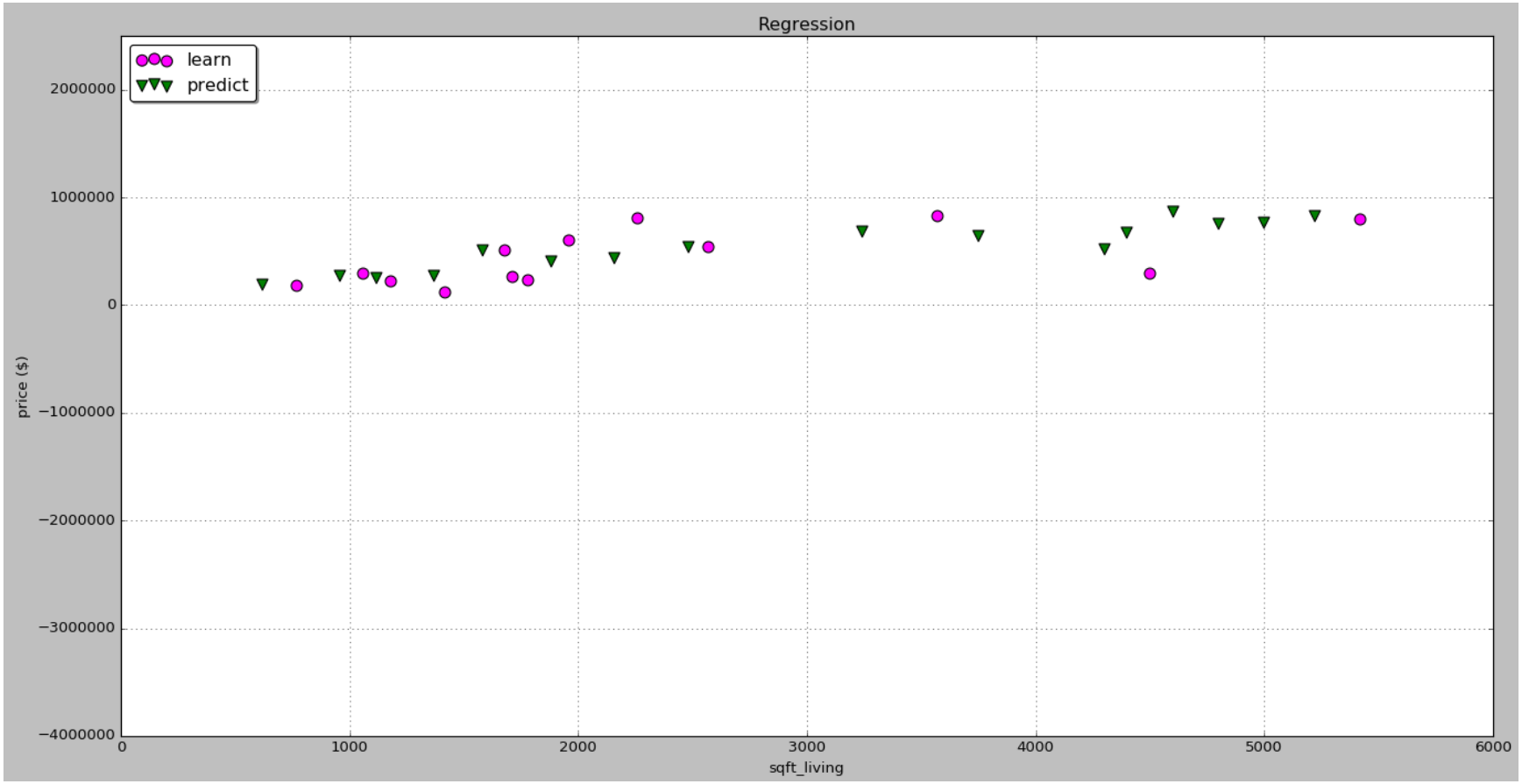
Błąd uczenia, testowy i generalizacji



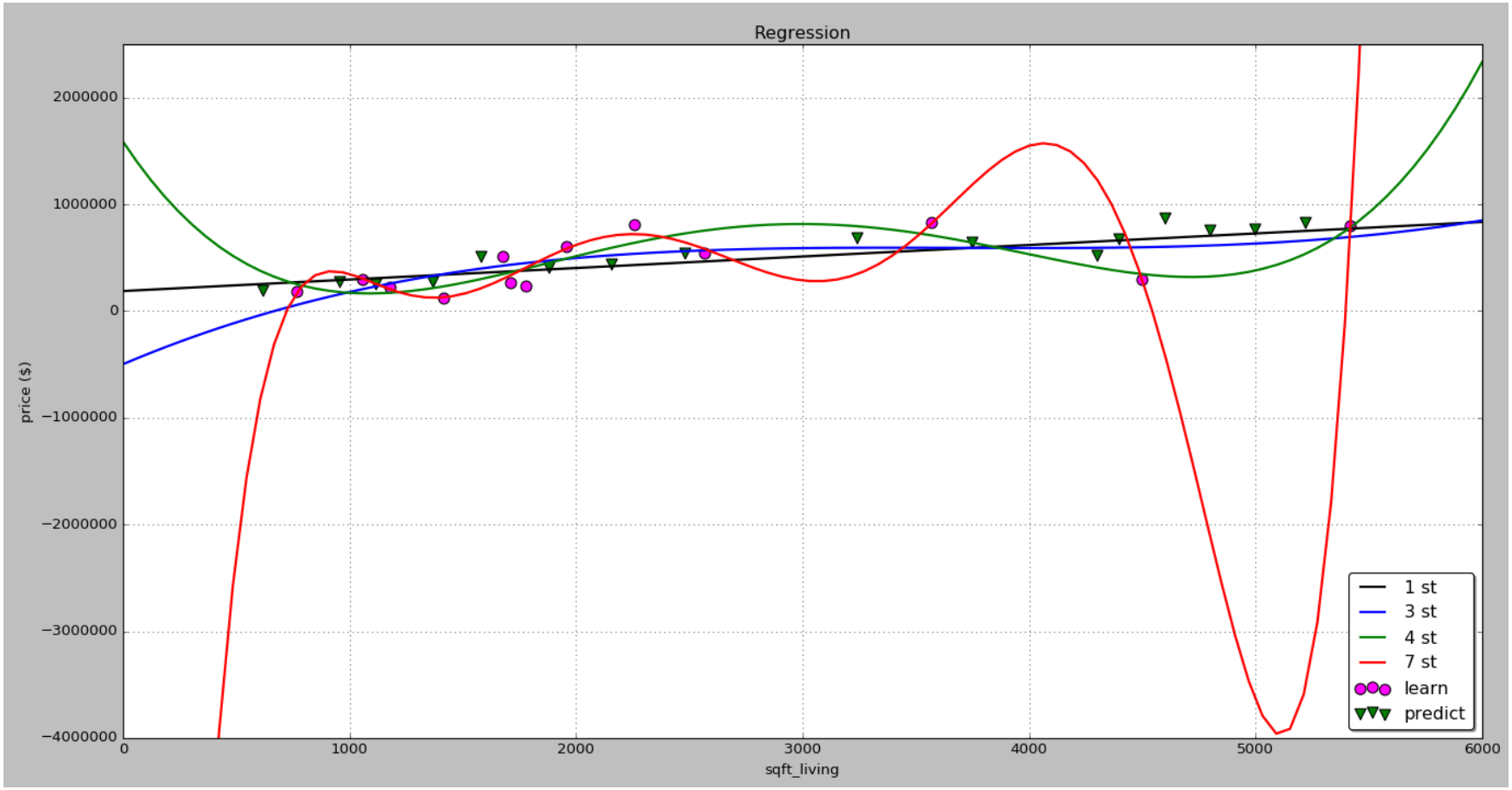
Wyznazanie błędu testowego

Przykładowy zbiór danych:

- okręgi – zbiór uczący
- trójkąty – zbiór testowy



Wyznaczanie błędu testowego



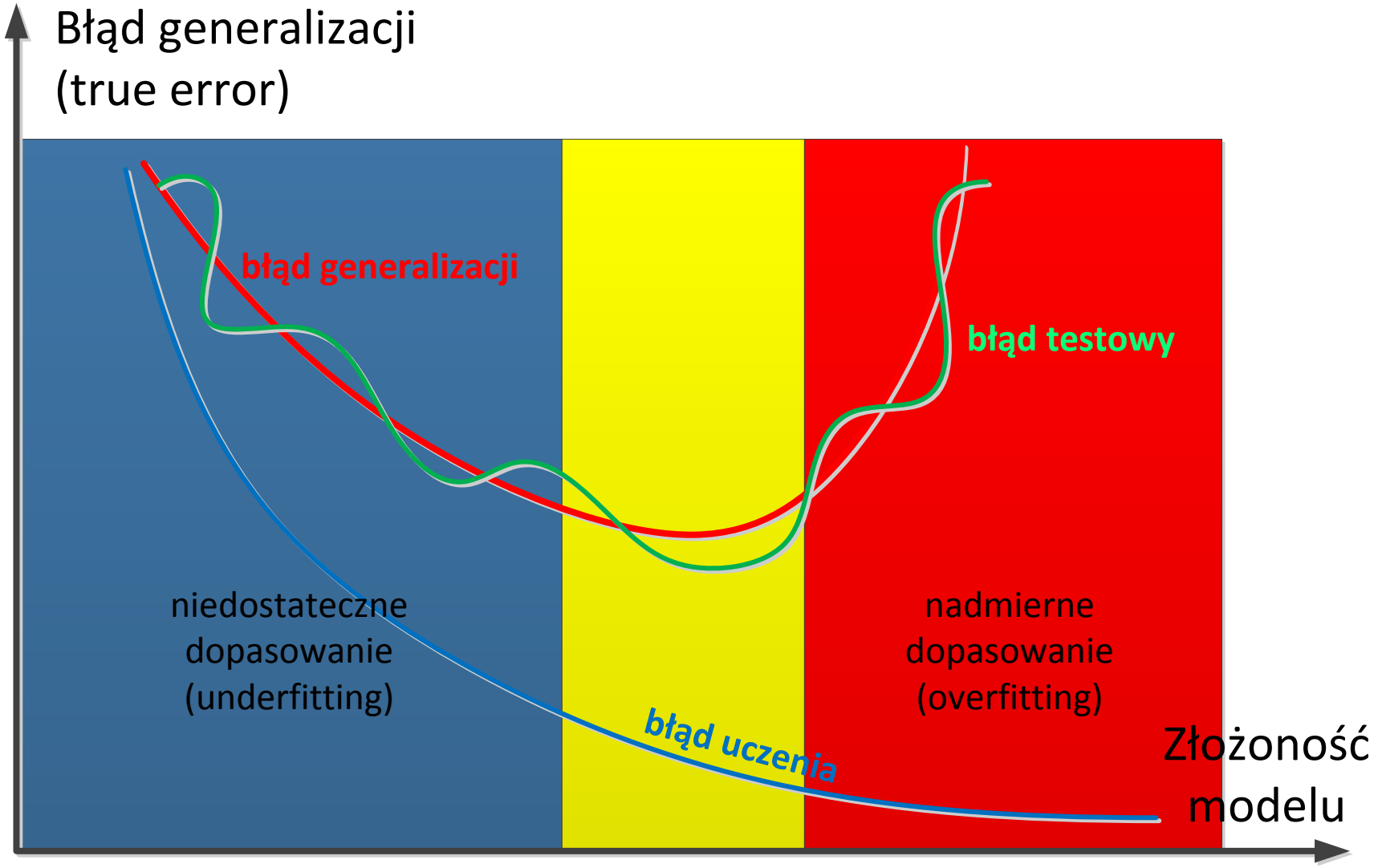
1 stopień	3 stopień	4 stopień	7 stopień
MSE= 7 909 189 801.61 RMSE= 88 933.63 MAE= 71 760.10 MedAE= 59 090.83	MSE= 15 454 501 213.96 RMSE= 124 316.13 MAE= 105 606.28 MedAE= 87 509.10	MSE= 60 899 960 613.24 RMSE= 246 779.17 MAE= 198 221.30 MedAE= 162 279.35	MSE= 3 140 346 233 438.67 RMSE= 1 772 102.21 MAE= 1 037 183.83 MedAE= 317 487.51

Wyznaczanie błędu testowego

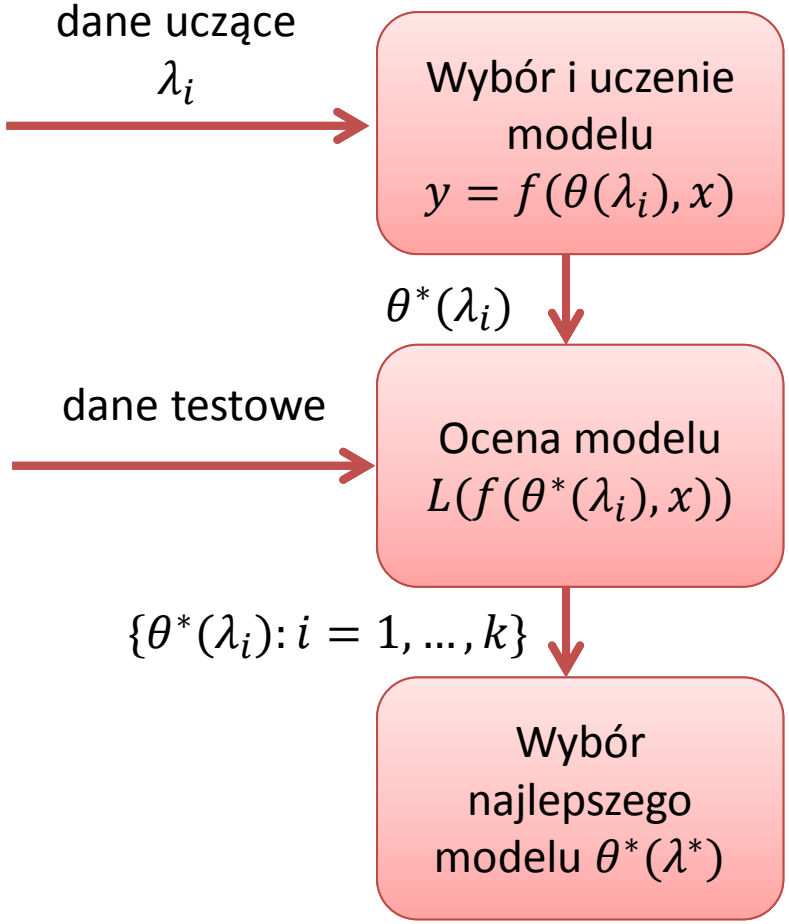
- Krzywe bardziej złożonych modeli coraz lepiej przybliżają dane użyte do uczenia
- Równocześnie stają się coraz bardziej skomplikowane (wiele ekstremów, na ogół pomiędzy punktami danych)
- **W tym przypadku:** jeżeli modele są testowane na danych nie użytych do uczenia, wszystkie miary rosną wraz ze złożonością modelu
- Nie jest to zależność ogólna (nie zawsze model prostszy jest lepszy), ale wybierając pomiędzy równie dobrymi modelami, powinniśmy wybrać prostszy (William Ockham XIII w)

Essentially, all models are wrong, but some are useful.
[Wszystkie modele są złe, ale niektóre są użyteczne.]
George Box

Niedostateczne i nadmierne dopasowanie



Przebieg uczenia

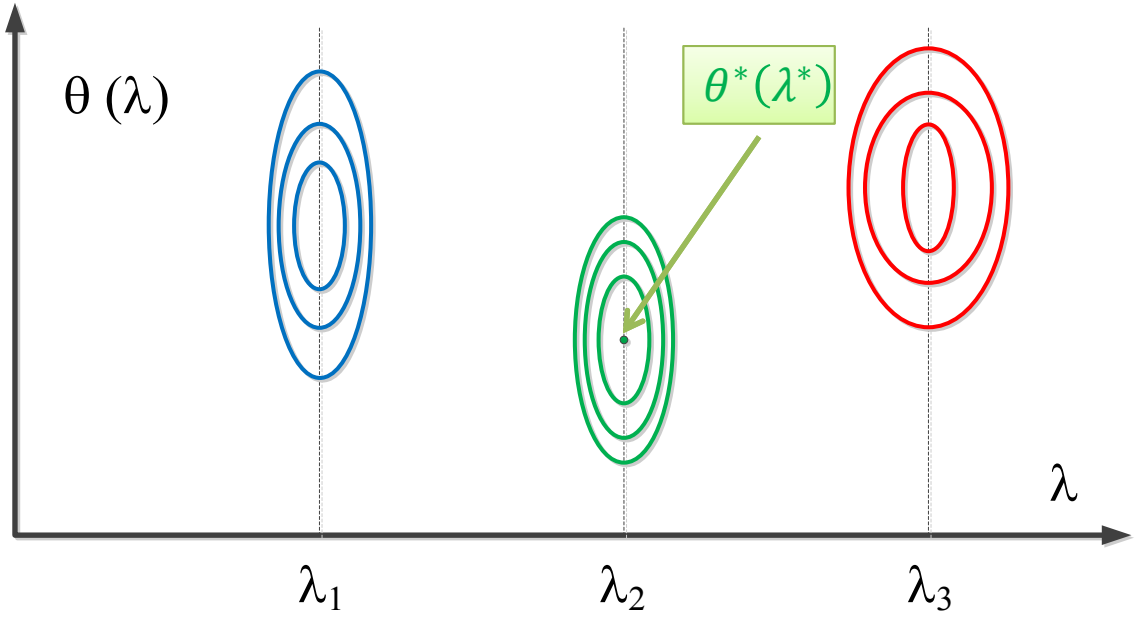


- Wybór i uczenie modelu
  - wybierana jest złożoność (postać) modelu λ_i (np. stopień wielomianu)
  - podczas uczenia (minimalizacja funkcji oceny, np. RSS) wyznaczany optymalny zestaw parametrów$\theta^*(\lambda_i)$
- Ocena modelu:
  - Błąd generalizacji $L(f(\theta^*(\lambda_i), x))$ określany na podstawie danych testowych
  - Wybierany jest najlepszy model $f(\theta^*(\lambda^*), x)$

Przebieg uczenia

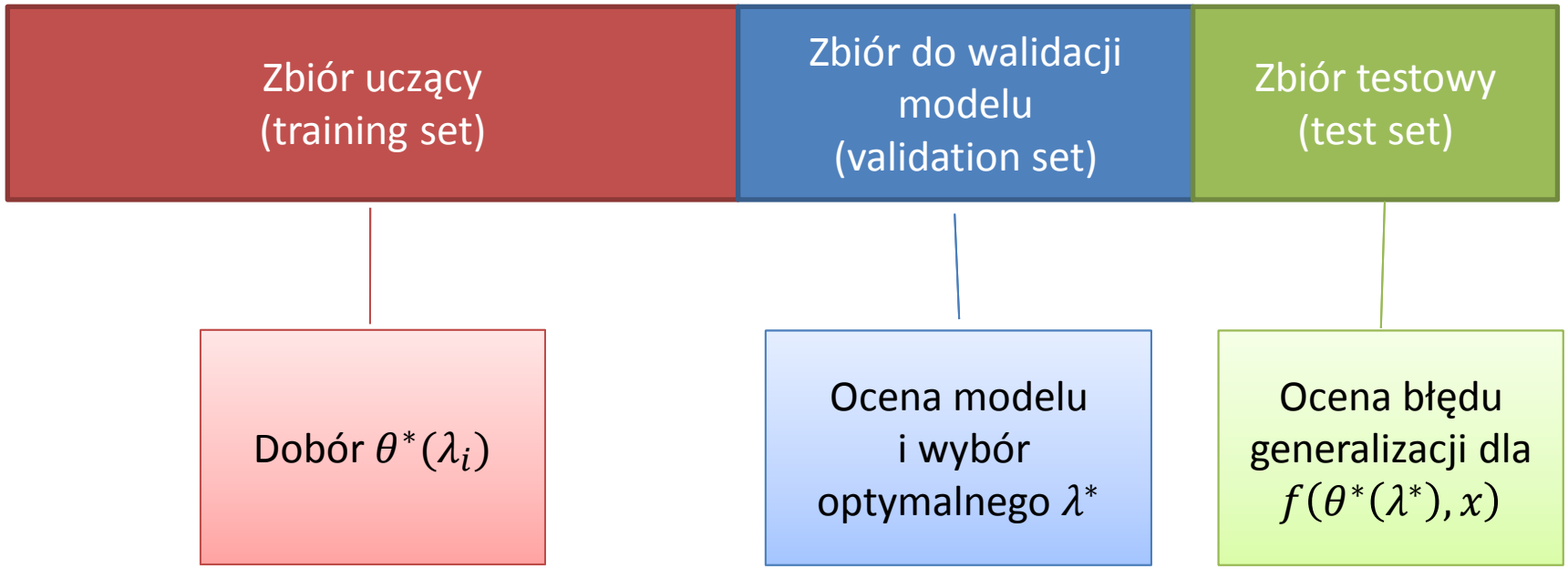
Uczenie jest procesem optymalizacji w dwóch kierunkach λ i θ :

- wybór postaci modelu i złożoności λ (dyskretny) i ocena z użyciem D_{test}
- optymalizacja parametrów modelu $\theta(\lambda_i)$ i ocena z użyciem D_{train}



- Czy $L\left(f\left(\theta^*(\lambda^*), x\right)\right)$ jest dobrym oszacowaniem błędu generalizacji?
- W ogólnym przypadku jest **zbyt optymistyczne**.
- Podczas optymalizacji w obu kierunkach możliwe jest nadmierne dopasowanie do zbioru uczącego i zbioru testowego

Rozwiązanie - dwa zbiory testowe



- Wybór λ^* minimalizującego błąd z użyciem **zbioru walidacyjnego**
- Aproksymacja błędu generalizacji modelu $f(\theta^*(\lambda^*), x)$ z użyciem **zbioru testowego**
- Typowe podziały:
  - uczący: 80%, walidacyjny: 10%, testowy: 10%
  - uczący: 50%, walidacyjny: 25%, testowy: 25%

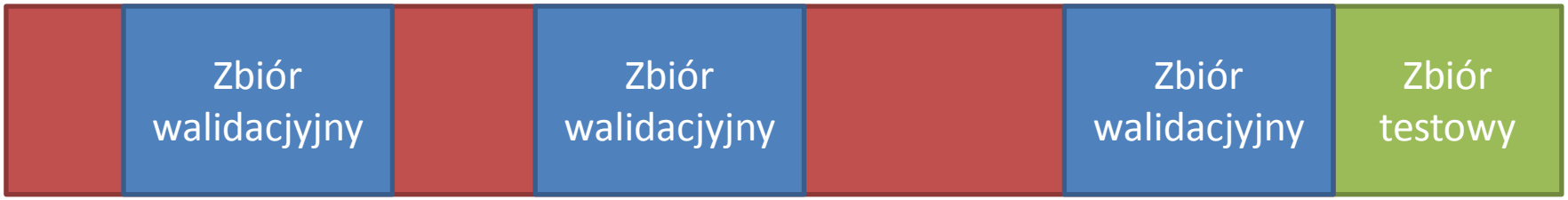
Sposób podziału zbioru danych

- Wydzielamy **zbiór testowy** z całego zbioru danych
- Powinien on w miarę możliwości reprezentować różnorodne dane (np. dobrany losowo)

Sposób podziału zbioru danych

- Pozostałe dane dzielimy na **zbiór uczący** i **zbiór walidacyjny**
- Zbiór walidacyjny służy do oceny jakości modelu $f(\theta^*(\lambda_i), x)$ w zależności od parametru sterującego złożonością λ_i.
- Jest z reguły niewielki.

Sposób podziału zbioru danych



- Do walidacji modelu może zostać wykorzystany dowolny podzbiór danych
- Ocena błędu będzie najbardziej wiarygodna, jeśli użyte zostaną **wszystkie dane** i ocena zostanie **uśredniona**
- Zauważmy, że wszystkie metryki, poza RMSE są addytywne:

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

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

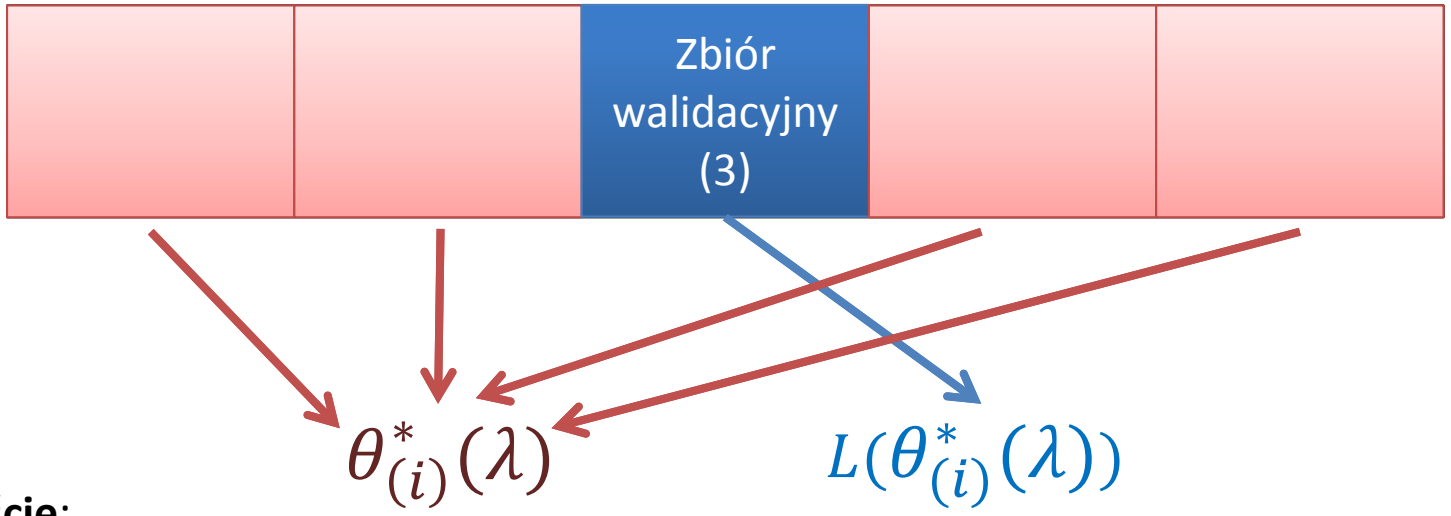
k-krotna walidacja krzyżowa

Angielski termin **k-fold cross validation** tłumaczony jest jako walidacja krzyżowa, sprawdzian krzyżowy lub kroswalidacja.

Przetwarzanie wstępne

- dane są dzielone losowo na k grup, każda liczy $\frac{m}{k}$ elementów
- podział jest zachowywany w kolejnych etapach

k-krotna walidacja krzyżowa



Wejście:

- zbiór danych podzielony na k rozłącznych bloków
- Parametr λ sterujący złożonością modelu

for $i = 1, \ldots, k$:

Wyznacz $\theta^*_{(i)}(\lambda)$ posługując się danymi z bloków $\{1, \ldots, k\} \backslash \{i\}$

Oszacuj i-ty wskaźnik błędu: $L(\theta^*_{(i)}(\lambda))$

Oblicz sumaryczny błąd:

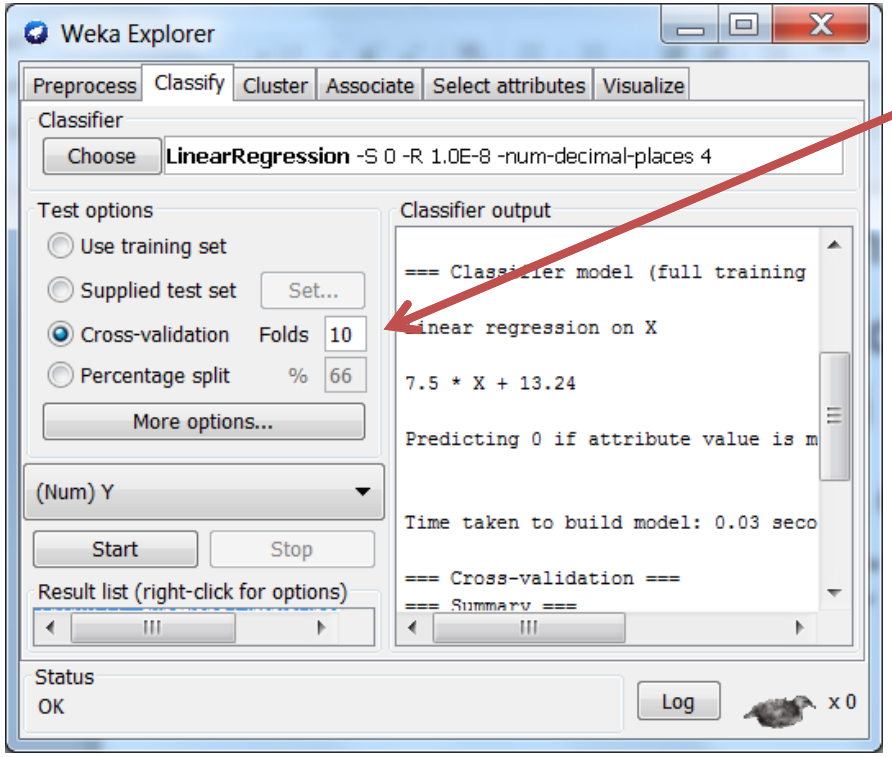
$$L(\lambda) = \frac{1}{k} \sum_{i}^{k} L\left(\theta^*_{(i)}(\lambda)\right)$$

k-krotna walidacja krzyżowa

- Jeżeli walidacja krzyżowa ma na celu wybór spośród różnych postaci modeli $\{\lambda_1, \dots, \lambda_r\}$, wybierany jest model, dla którego $L(\lambda_i)$ osiąga minimalną wartość.
  Wyznaczanych jest $k \cdot r$ modeli.
- Jeżeli walidacji poddany jest tylko jeden typ modelu (λ), wyznaczanych jest k modeli, dla każdego z nich $L(\theta^*_{(i)}(\lambda))$ i wartość jest uśredniana.
- Weka: ostatecznie model o złożoności λ jest wyznaczany dla całego dostępnego zbioru.

Wybór k

- Jakość oszacowania błędu rośnie wraz z k (liczbą bloków)
- Teoretycznie, najlepszą wartością jest $k = m$, czyli wydzielane są bloki jednoelementowe: **leave one out** cross validation
- Kosztowne obliczeniowo: wymaga wyznaczenia tylu modeli, ile jest obserwacji



Typowe wartości:
$k = 10$ lub $k = 5$
(ang: ten fold cross validation)

Walidacja krzyżowa jest częstą metodą oceny prototypowych algorytmów:

1. zaimplementuj nowy algorytm
2. wybierz zbiór danych z UCI Machine Learning Repository
3. zastosuj walidację krzyżową dla całego zbioru
4. porównaj wyniki z innymi

Kompromis pomiędzy bias i variance

- Wybór modelu w problemach uczenia nadzorowanego (nie tylko regresji) wiąże się z realizacją dwóch sprzecznych celów:
  - Model powinien być dobrze dopasowany do danych uczących, aby uchwycić zależności pomiędzy danymi
  - Model powinien też dobrze przybliżać nieznane dane (zapewniać mały błąd generalizacji)
- Modele złożone dobrze dopasowują się do danych wyjściowych, ale charakteryzują się dużą zmiennością (**variance**) wartości wyjściowych. Ryzykiem jest nadmierne dopasowanie (**overfitting**)
- Modele prostsze są obciążone dużym błędem systematycznym (**bias**) i ich zastosowanie niesie ryzyko niewystarczającego dopasowania (**underfitting**).
- Trzecim składnikiem błędów generalizacji jest nieredukowalny błąd związany ze **zmiennością danych**

Kompromis pomiędzy bias i variance

- Dla zagadnienia regresji i średniokwadratowej funkcji oceny MSE można przeprowadzić formalną dekompozycję na trzy składniki:
  - bias (odchylenie systematyczne)
  - variance (wariancja związana z modelem)
  - σ^2 - wariancja błędów danych
- Zagadnienie regresji
  - Dany jest zbiór uczący $\mathrm{D}_{\mathrm{train}} = \{(x_i, y_i)\}_{i=1,m}$
  - Poszukiwana jest „prawdziwa” funkcja $y = f(x)$, przy założeniu, że $y_i = f(x_i) + \varepsilon_i$ oraz ε=$N(0, \sigma^2)$ - wartość oczekiwana $\mathrm{E}[\varepsilon] = 0$, wariancja σ^2.

Kompromis pomiędzy bias i variance

- Załóżmy, że w wyniku uczenia wyznaczona została funkcja $y = \hat{f}(x)$.
- Jak funkcja zachowa się na **nieznanych** danych $D = \{(x_i, y_i): i = 1, k\}$?
- Trzy składniki:

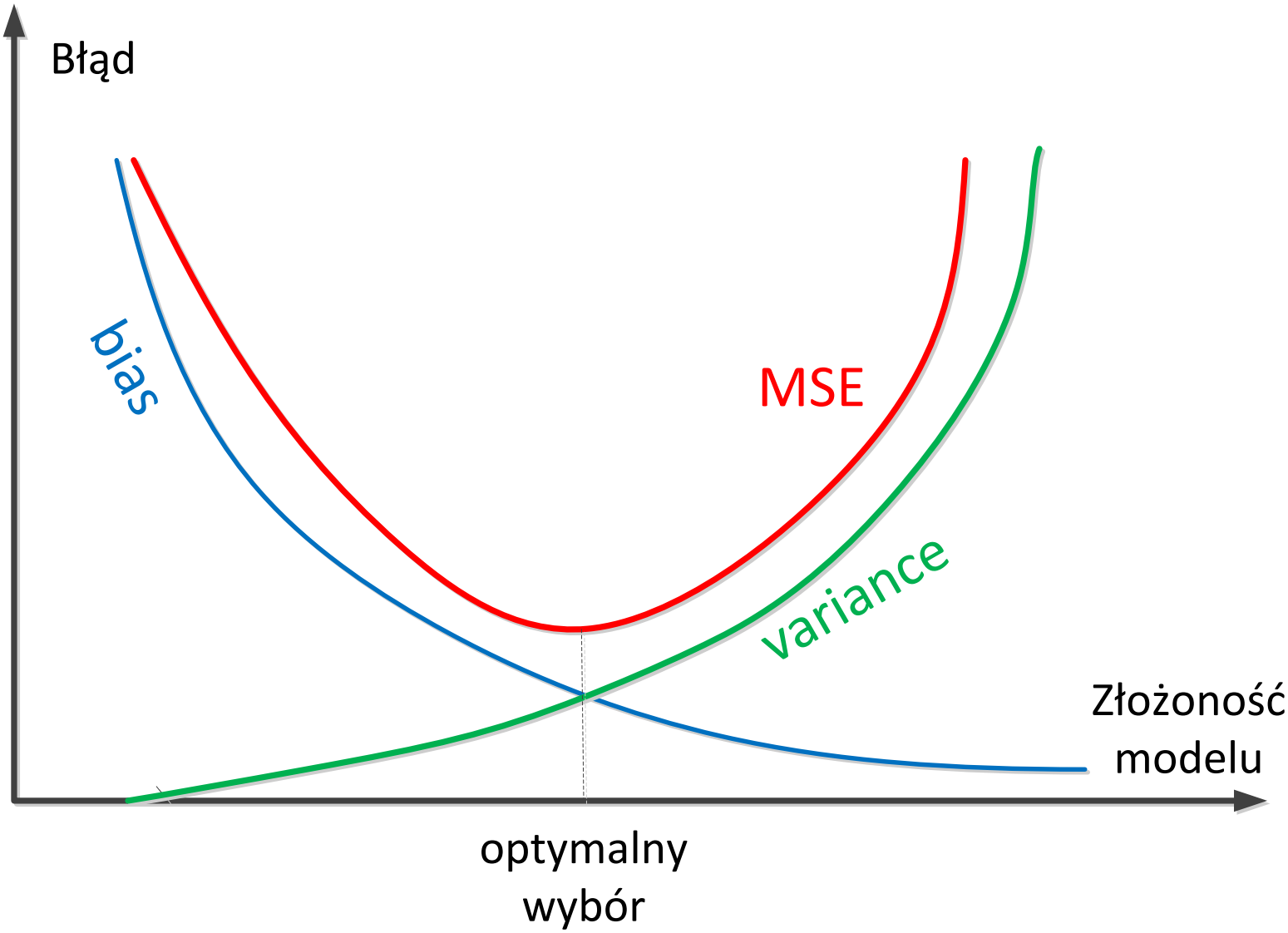
$$MSE = E_D\left[\left(y - \hat{f}(x)\right)^2\right] = Bias_D\left(\hat{f}(x)\right)^2 + Var_D\left(\hat{f}(x)\right) + \sigma_D^2$$

gdzie:

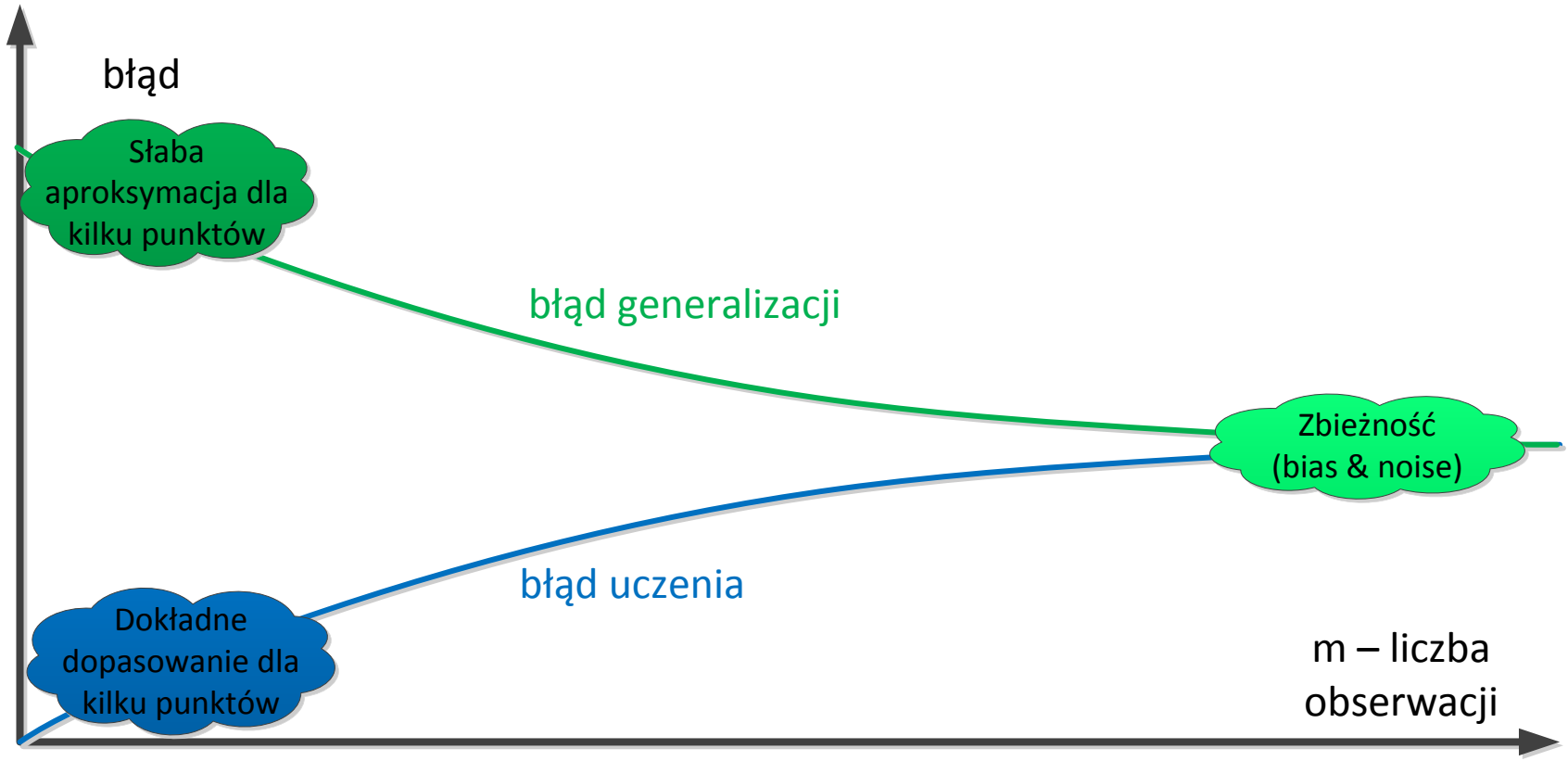
- $Bias_D\left(\hat{f}(x)\right) = \mathrm{E_D}[\hat{f}(x) - f(\mathrm{x})]$
- $Var_D\left(\hat{f}(x)\right) = \mathrm{E}_D[\hat{f}(x)^2] - \mathrm{E}\left[\hat{f}(x)\right]^2$
- $\sigma_D^2 = E(y^2) - E[y]^2$ (wariancja danych w D)

- Komponent **bias** (odchylenie systematyczne) spada ze wzrostem złożoności modelu. Skrajnym przypadkiem jest funkcja stała.
- Komponent **variance** jest zerowy dla funkcji stałej i wraz ze wzrostem złożoności modelu rośnie (przebiegi $\hat{f}(x)$ stają się coraz bardziej skomplikowane i dla nieznanych danych wartości są bardzo zmienne)
- Komponent σ_D^2 opisuje **szum** (zmienność danych): na wejście estymatora można dostarczyć dowolnie zaszumione dane, które wpłyną na wynik końcowy.

Kompromis pomiędzy bias i variance



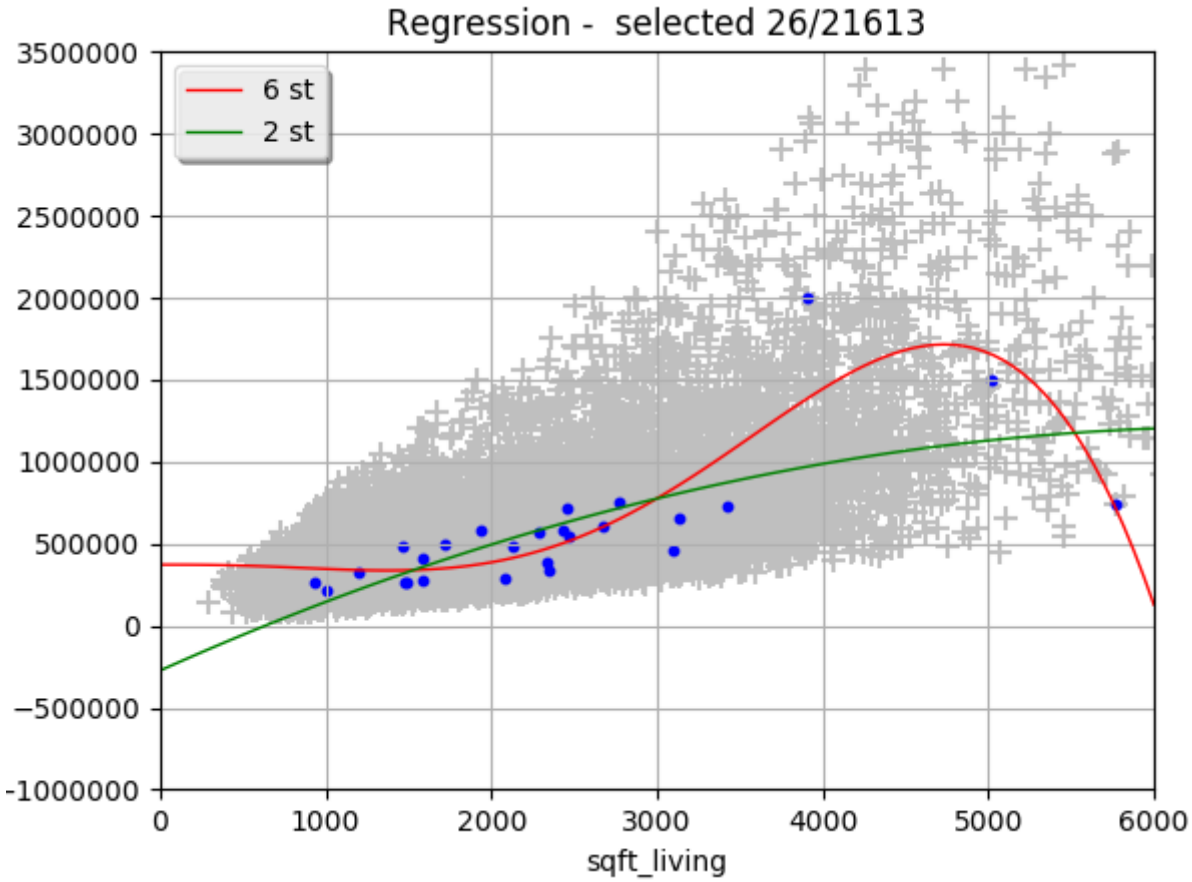
Zależność błędu od liczby obserwacji w zbiorze uczącym



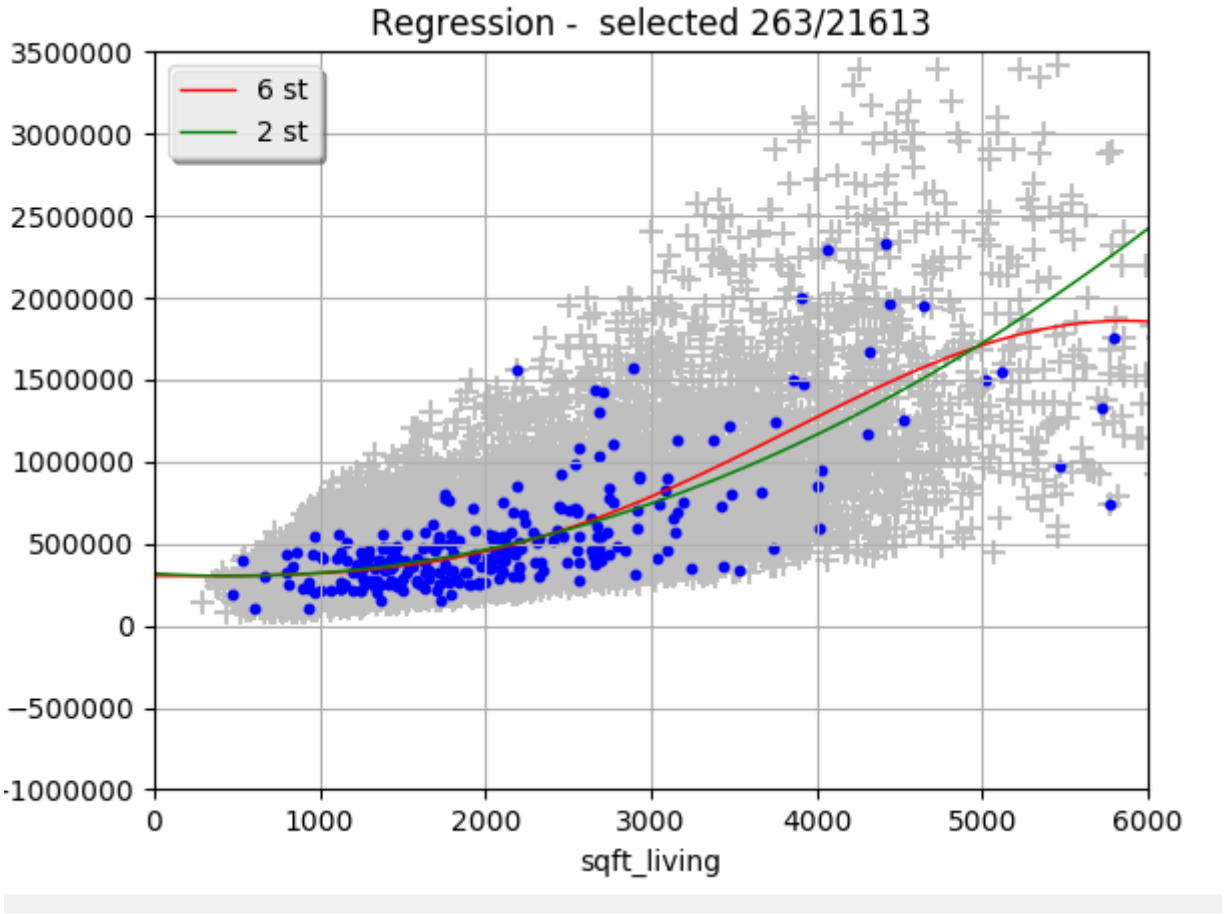
Porównanie modeli 1

Szare krzyżyki – rzeczywiste dane (w tym te nieznane)
Niebieskie kropki – dane użyte do budowy modelu (uczenia)

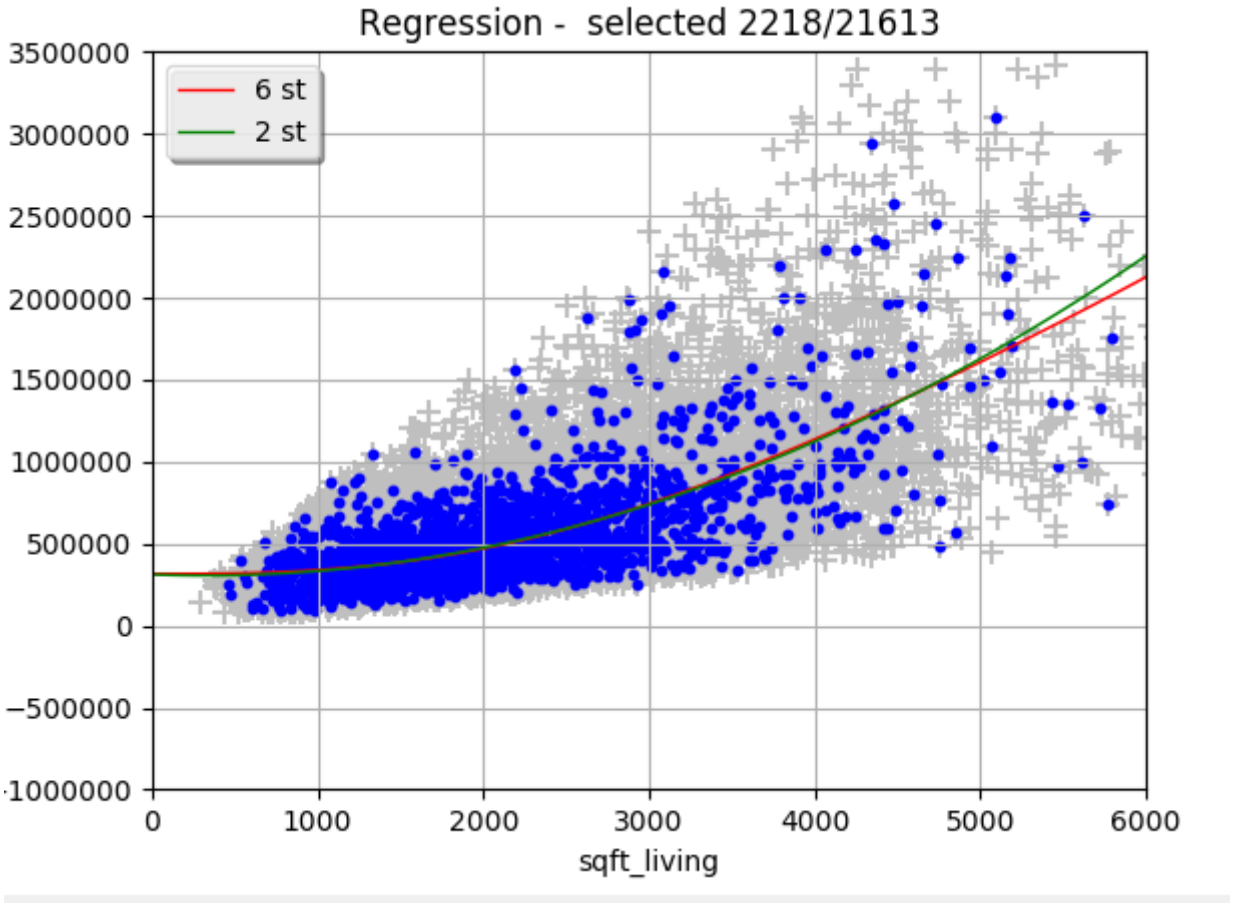
Porównanie modeli 2



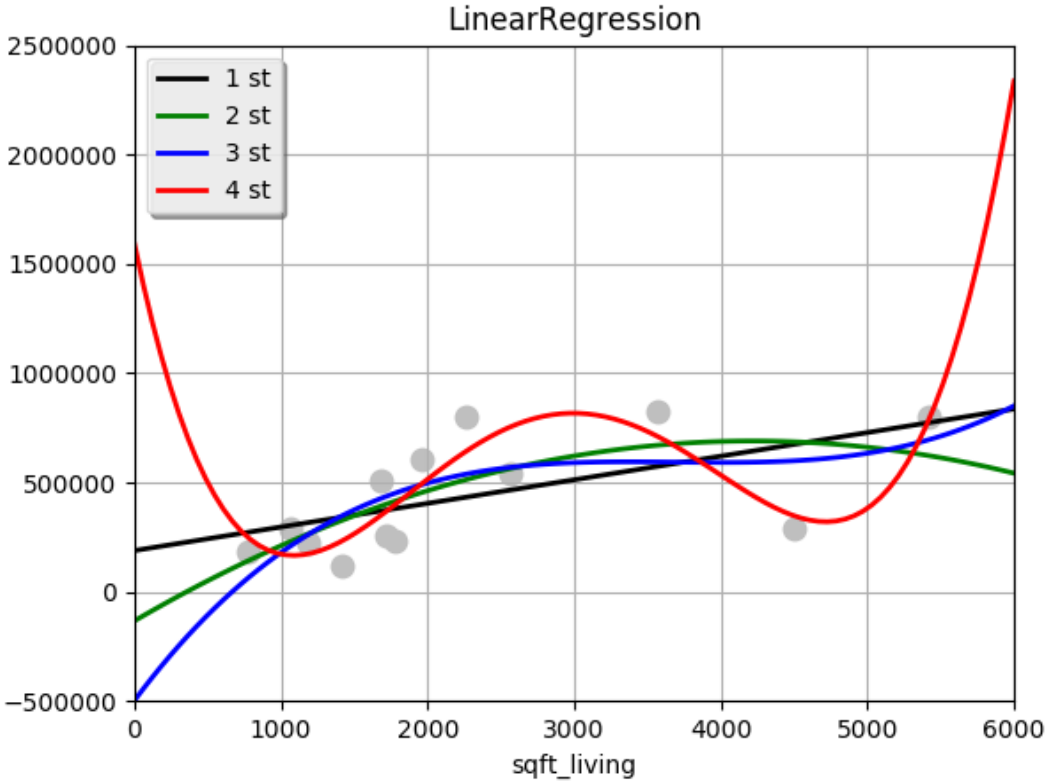
Porównanie modeli 3



Porównanie modeli 4

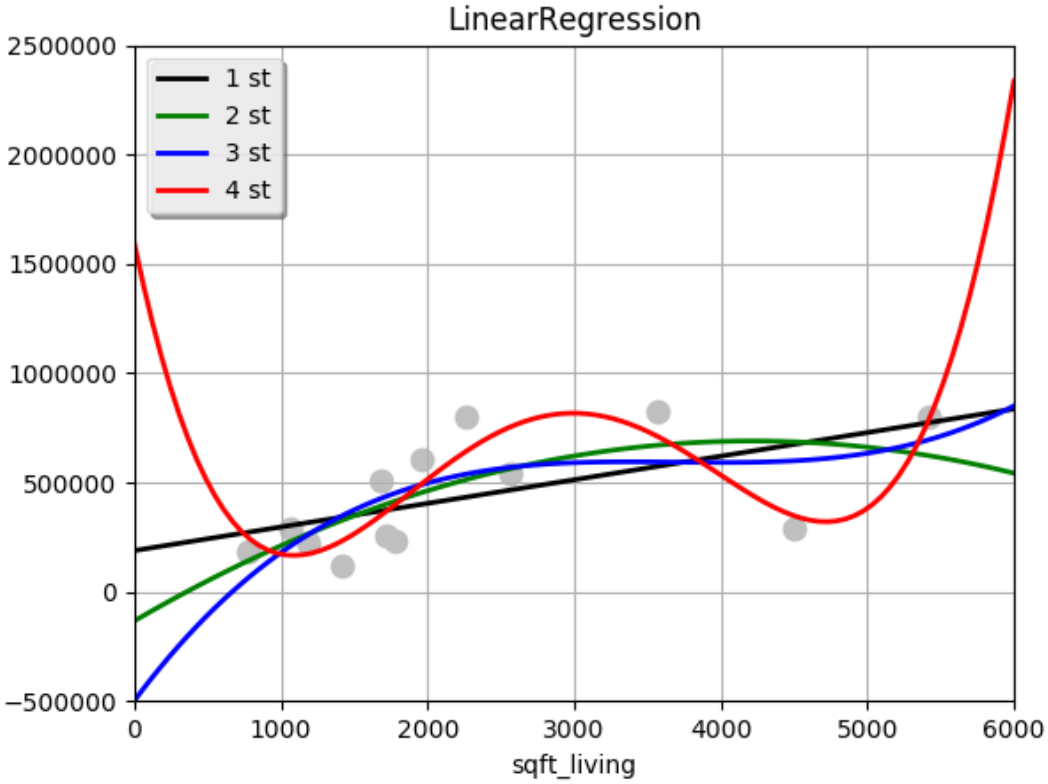


Regularyzacja



- Dla regresji wielomianowej algorytm będzie starał się dobrze dopasować krzywą do obserwacji.
- Im większy stopień wielomianu tym więcej jest możliwych zagięć krzywej
- Równocześnie krzywizny stają się coraz „chudsze”, co z reguły odpowiada wysokim wartościom wag

Regularyzacja 2



- $f_1(x) = \mathbf{107.97}\, x^1 + 188160.37$
- $f_2(x) = \mathbf{390.28}\, x^2 - 0.05\, x^1 - 133371.02$
- $f_3(x) = \mathbf{900.23}\, x^3 - 0.25\, x^2 + 0.00\, x^1 - 497434.19$
- $f_4(x) = \mathbf{-3214.81}\, x^4 + 2.36\, x^3 - 0.00\, x^2 + 0.00\, x^1 + 1589582.21$

Dlaczego duże wagi są złe?

- Dla regresji wielomianowej jednej zmiennej nie jest to oczywiste – cechy są dość mocno skorelowane, więc wpływ dużej wagi w_i może skorygowany przez wpływ innej dużej wagi w_j
- Dla niezależnych cech – duża wartość wagi w_i oznacza dużą wrażliwość funkcji regresji na drobne fluktuacje i-tej cechy $h_i(x)$
  - Dla blisko położonych obserwacji x^k oraz x^l różnice wartości funkcji $f(x^k)$ i $f(x^l)$ mogą być bardzo duże.
  - Model bardzo dobrze dopasowany do danych uczących może nie sprawdzić się dla nieznanych danych
- Lepszym rozwiązaniem jest gorsze dopasowanie do danych uczących przy równoczesnym ograniczeniu parametrów świadczących o potencjalnie dużym błędzie generalizacji – czyli zmniejszeniu wartości wag.

Regularyzacja L2 (Ridge Regression)

- W przypadku zwykłej regresji liniowej szukane są wagi minimalizujące funkcję celu postaci

$$J(w) = \sum_{i=1}^{m} (y_i - w^T x_i)^2$$

- Dla regresji grzbietowej dodany jest składnik (funkcja kary) ograniczający wartości wag

$$J(w) = \sum_{i=1}^{m} (y_i - w^T x_i)^2 + \lambda \|w\|^2$$

- Czynnik $\|w\|^2$ to tzw. norma L^2, czyli po prostu:

$$w^T w = \sum_{i=1}^{n} w_i^2$$

- Nieujemna stała λ określa udział składnika $\|w\|^2$ w funkcji celu.

Regularyzacja L2 (Ridge Regression)

Na poprzednim wykładzie pokazane było rozwiązanie analityczne polegające na

- Obliczeniu gradientu funkcji celu RSS i przyrównanie do 0:
$$-2X^T y + 2X^T Xw = 0$$
- Znalezieniu rozwiązaniu równania postaci:
$$w = (X^T X)^{-1} X^T y$$

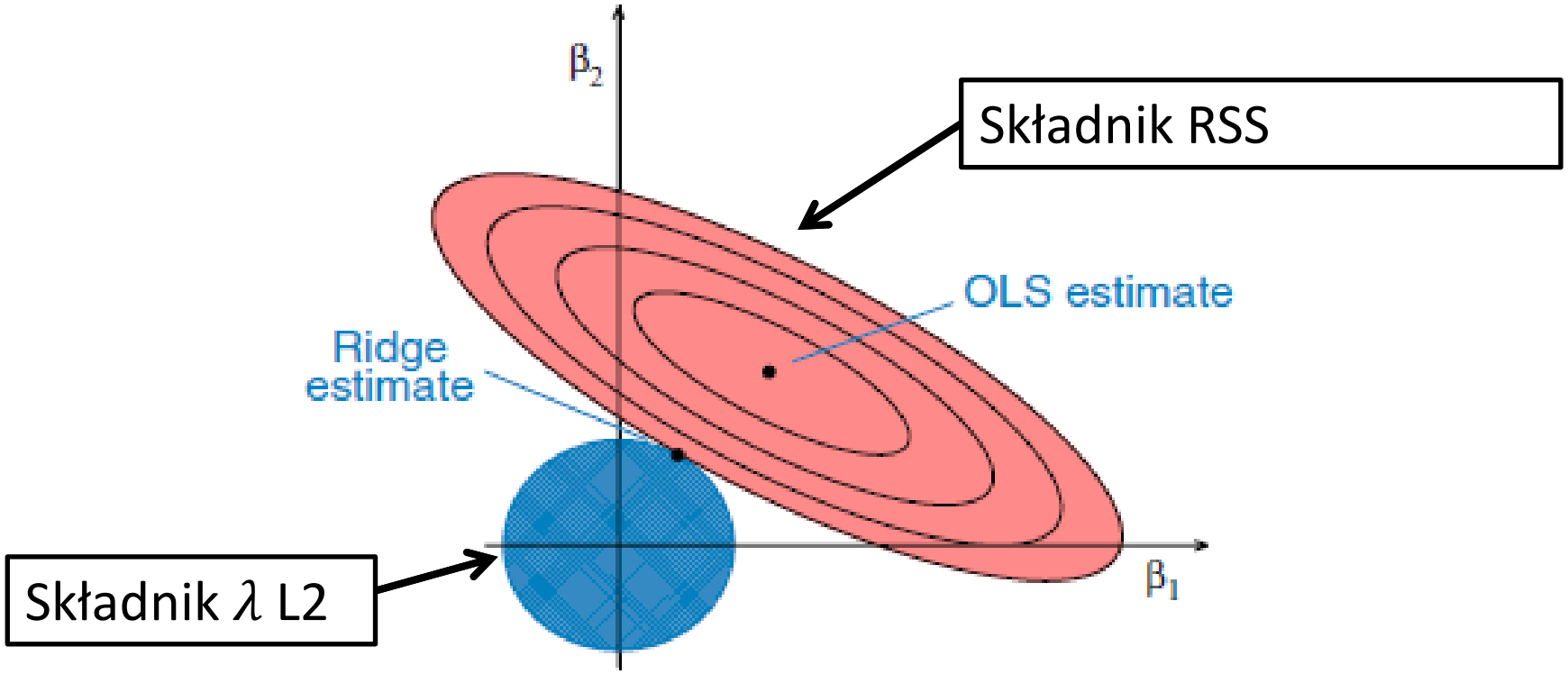
W tym przypadku analogiczne operacje to:

- Obliczenie gradientu: $\nabla J(w) = -2X^T y + 2X^T Xw + 2\lambda w = 0$
- Wyznaczenie rozwiązania: $w = (X^T X + \lambda I)^{-1} X^T y$

Jeżeli w X jest wiele mocno skorelowanych atrybutów macierz $(X^T X)^{-1}$ może nie istnieć (liniowa zależność kolumn). Dodanie wystarczająco dużej wartości λ na przekątnej $X^T X + \lambda I$ usuwa osobliwość macierzy.

Regularyzacja L2 (Ridge Regression) -ilustracja

$$J(w) = \sum_{i=1}^{m}(y_i - w^T x_i)^2 + \lambda\|w\|^2$$



Źródło: https://onlinecourses.science.psu.edu/stat857/node/155
Tutaj: β_1 i β_2 to wagi w

Regularyzacja L2 – analiza lambda

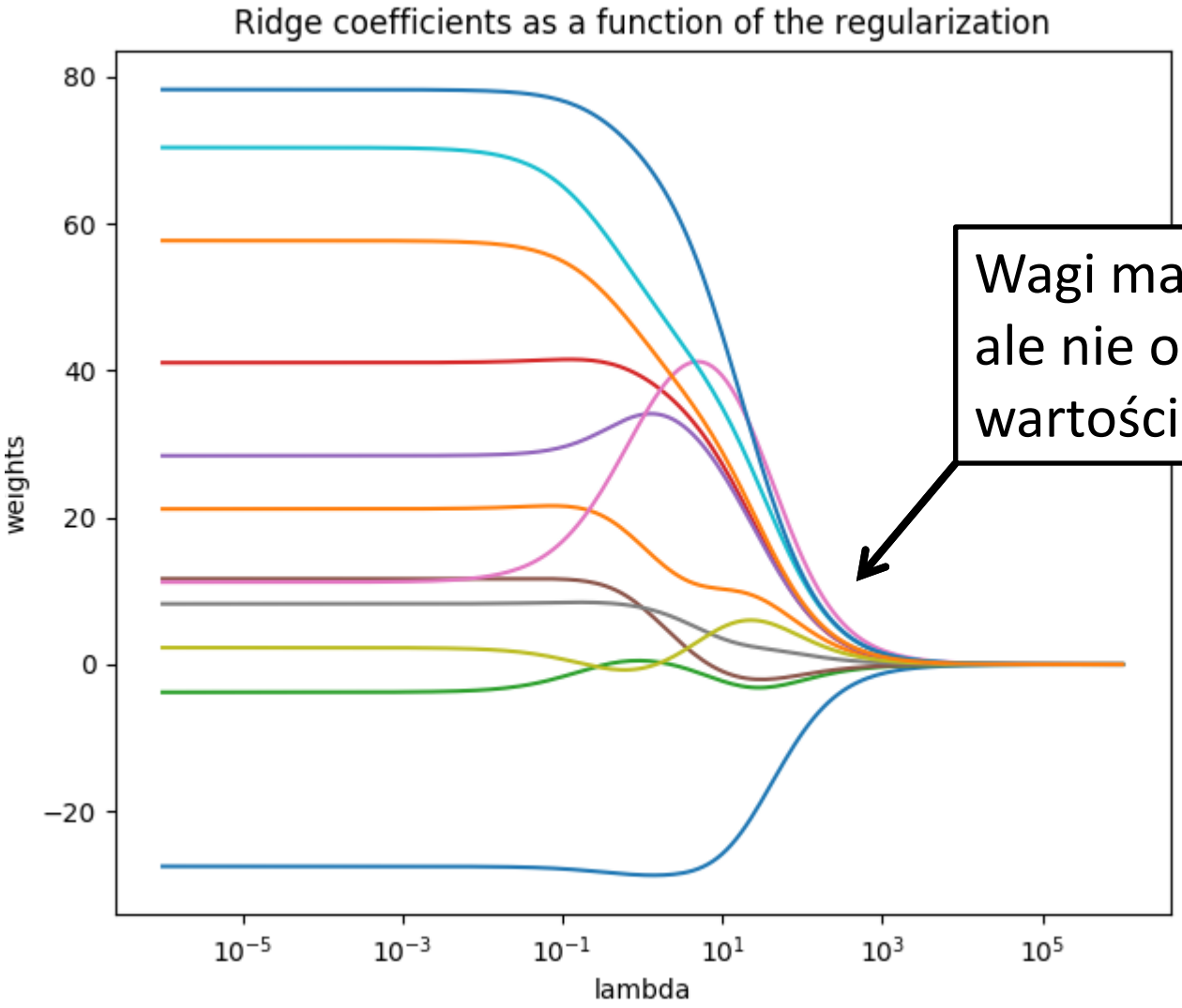
$$J(w) = \sum_{i=1}^{m} (y_i - w^T x_i)^2 + \lambda \|w\|^2$$

- Jeżeli $\lambda = 0$, funkcja celu jest taka sama, jak dla zwykłej regresji
- Dla małych wartości λ wpływ czynnika regularyzującego będzie mniejszy – wagi będą się powiększać
- Dla dużych wartości λ wagi będą bliskie zeru (i większości przypadków błąd RSS będzie duży)

Czasami współczynnik oznaczany jest jako α

class `sklearn.linear_model.` **Ridge** (*alpha=1.0, fit_intercept=True, normalize=False, copy_X=True, max_iter=None, tol=0.001, solver='auto', random_state=None*) [source]

Regularyzacja L2 – analiza lambda



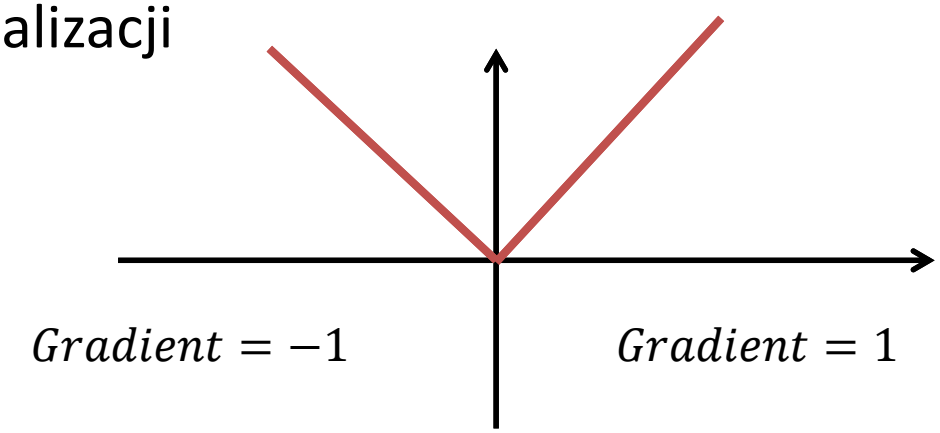
Wagi maleją do zera, ale nie osiągają wartości 0

Regularyzacja L1 (Lasso)

- W przypadku regularyzacji L1 składnikiem regularyzującym jest norma L1 (czyli suma wartości bezwzględnych wag)

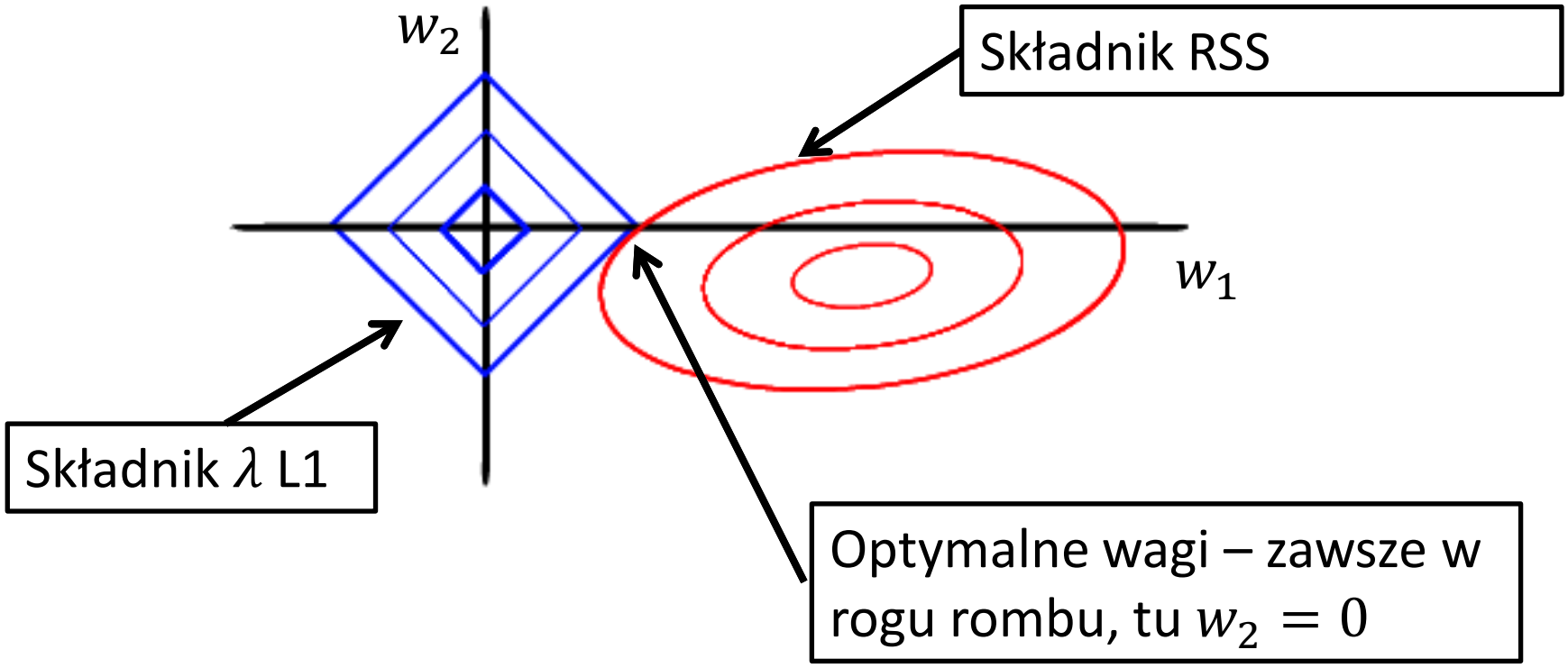
$$J(w) = \sum_{i=1}^{m} (y_i - w^T x_i)^2 + \lambda \sum_{i=1}^{n} |w_i|$$

- Funkcja ta nie jest różniczkowalna w zerze, więc nie istnieje rozwiązanie analityczne
- Mimo tego można zastosować gradientowe metody optymalizacji



Regularyzacja L1 (Lasso) -ilustracja

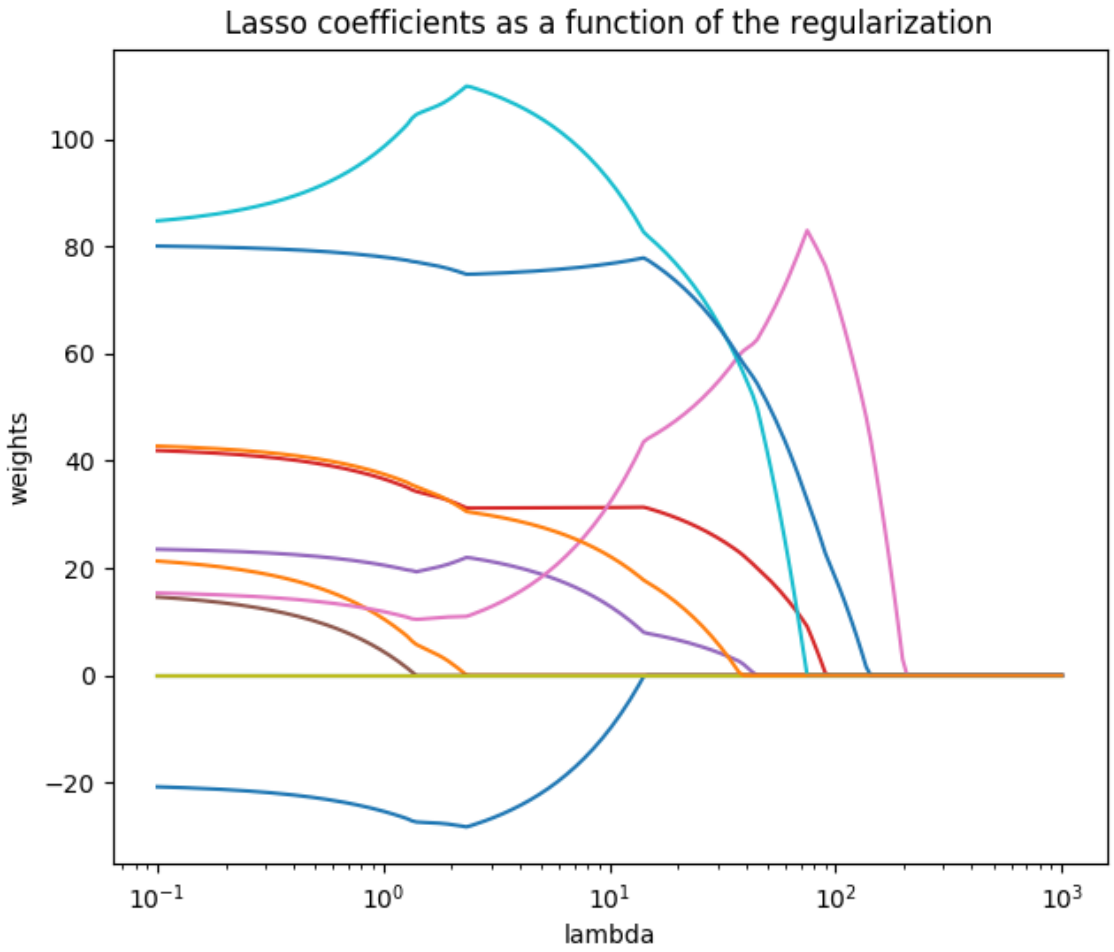
$$J(w) = \sum_{i=1}^{m}(y_i - w^T x_i)^2 + \lambda \sum_{i=1}^{n}|w_i|$$



Źródło: https://stats.stackexchange.com/questions/30456/geometric-interpretation-of-penalized-linear-regression

Regularyzacja L1 – analiza lambda

- Dla L1 wraz ze wzrostem lambda kolejne wagi będą znikać (przyjmować wartość 0)
- Dla L2 wagi będą stawały się dowolnie małe, ale nie zanikały zupełnie



Lasso działa więc jak mechanizm wyboru cech (ang. feature selection).

Stopniowo odrzuca współliniowe atrybuty, pozostawia zbiór najbardziej istotnych (tych, które najlepiej „objaśniają” zmienność wartości wyjściowych).