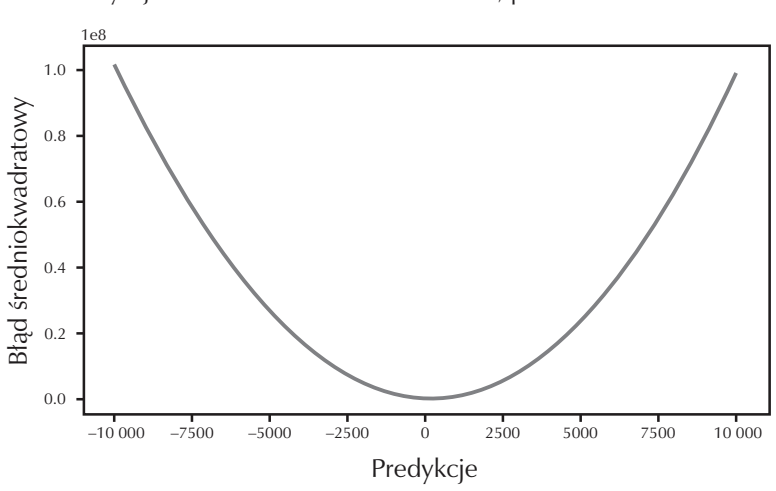
te różnice, tym dokładniejszy model. Do oceny modelu używa się funkcji kosztu θ. Uczenie modelu polega na znalezieniu takich wartości parametrów, dla których funkcja θ przyjmuje minimum. Zobaczmy, jak można znaleźć takie wartości parametrów.

7.8.1. Uczenie na błędach

Jedną z najczęściej stosowanych do oceny modeli regresji funkcji kosztu jest błąd średniokwadratowy *MSE*. Dla jednej zmiennej wejściowej *x* i *n* przykładów błąd średniokwadratowy wynosi

$$J(\beta_0, \beta_1) = \frac{1}{2n}\Sigma_{i=1}^{n}(\beta_0 + \beta_i(x_i) - y_i)^2$$

Zauważmy, że błąd rośnie z kwadratem różnicy między przewidzianą a prawdziwą wartością zmiennej wyjściowej. To dobrze, bo model będzie się szybciej uczył na dużych błędach. Jeszcze ważniejsze jest to, że **funkcja kosztu MSE ma tylko jedno, globalne minimum**. Brak lokalnych minimów znacznie upraszcza znalezienie globalnego minimum funkcji (rys. 7.8).



Rysunek 7.8. Jeden duży błąd kosztuje model znacznie więcej niż wiele drobnych pomyłek. Weźmy 1000 przykładów. Jeżeli błąd dla jednego z nich wyniósł 5000, a dla pozostałych 999 błąd wyniósł 0, to błąd średniokwadratowy MSE wyniesie 25 000. Jeżeli błąd dla wszystkich 1000 przykładów wyniósł 10, to błąd średniokwadratowy MSE modelu będzie równy 100

W tym miejscu powinniśmy zapytać, czy funkcja kosztu θ mierzy błąd treningowy, czy testowy? Skoro do uczenia używa się wyłącznie danych treningowych, to **funkcja θ ocenia model na podstawie danych treningowych**. Wiemy już, że dostatecznie skomplikowany model pozwala zmniejszyć błąd treningowy do zera. Wiemy też, że mały błąd treningowy nie zawsze przekłada się na mały błąd testowy.

Żeby zapobiec nadmiernemu dopasowaniu modelu do danych treningowych, do reguły uczącej dodaje się człon z regularyzacją. Jego zadaniem jest zwiększenie kosztu bardziej złożonych modeli. Wróćmy do uproszczonego przykładu z jedną zmienną i dwoma parametrami β_0 i β_1. Uzupełniona o człon z regularyzacją funkcja kosztu wygląda następująco:

$$J(\beta_0, \beta_1) = \frac{1}{2n}\Sigma_{i=1}^{n}(\beta_0 + \beta_i(x_i) - y_i)^2 + \lambda\Sigma_{j=1}^{k}\beta_j^2$$

Hiperparametr λ (lambda) określa wpływ członu z regularyzacją na funkcję kosztu. Im jest on większy, tym większe ryzyko niedopasowania modelu. Im jest on mniejszy, tym większe ryzyko nadmiernego dopasowania modelu.

Jeżeli w członie z regularyzacją dodaje się kwadraty wartości parametrów, tak jak w tym przypadku, mówimy o regularyzacji *L2*, nieformalnie nazywanej *Ridge*. Zmniejsza ona wartości parametrów. Siła redukcji zależy od wartości λ. **Działanie regularyzacji *L2* polega na eliminowaniu mniej przydatnych wartości zmiennych wejściowych** (wartości, dla których błąd predykcji był największy).

Jeżeli w członie z regularyzacją dodaje się wartości bezwzględne parametrów, mamy do czynienia z regularyzacją *L1*, nieformalnie nazywaną *Lasso*

$$J(\beta_0, \beta_1) = \frac{1}{2n}\Sigma_{i=1}^{n}(\beta_0 + \beta_i(x_i) - y_i)^2 + \lambda\Sigma_{j=1}^{k}|\beta^2|$$

Ściąga ona wartości parametrów do zera. Siła ściągania zależy od wartości λ. **Działanie regularyzacji *L1* polega na eliminowaniu mniej przydatnych zmiennych wejściowych** (zmiennych o najmniejszej sile predykcyjnej).

Łącząc oba powyższe człony, otrzymamy elastyczną regularyzację sieci

$$J(\beta_0, \beta_1) = \frac{1}{2n}\Sigma_{i=1}^{n}(\beta_0 + \beta_i(x_i) - y_i)^2 + \lambda\Sigma_{j=1}^{k}|\beta^2| + \lambda\Sigma_{j=1}^{k}\beta_j^2 \quad [27]$$

Zatrzymajmy się ponownie na chwilę przy przykładzie modelu z jedną zmienną wejściową. Uprośćmy go jeszcze bardziej, usuwając parametr β_0. Teraz możemy przedstawić funkcję kosztu na wykresie liniowym jako krzywą, której wartości zależą od wartości parametru β_1. Uczenie na błędach polega na szukaniu minimum funkcji kosztu. Odpowiednio zmieniając wartości parametru, możemy zmniejszyć błędy modelu i w ten sposób zminimalizować funkcję kosztu. Kierunek zmian wartości parametru wyznacza pochodna funkcji kosztu.

Pochodna funkcji to miara szybkości zmian wartości funkcji względem najmniejszych zmian jej argumentów. Leibniz zapisywał pochodną jako $\frac{dy}{dx}$, co czyta się *pochodna y względem x*. Formalnie pochodną definiuje się jako $\frac{\Delta f(a)}{a}$. Pochodna krzywej, takiej jak funkcja kosztu *MSE*, reprezentuje nachylenie tej krzywej w danym punkcie. To znaczy, że mniejszą od bieżącej wartość funkcji kosztu możemy znaleźć, poruszając się w kierunku przeciwnym do znaku pochodnej (rys. 7.9).