Optymalizacja funkcji wielu zmiennych metodami gradientowymi

1. Cel ćwiczenia.

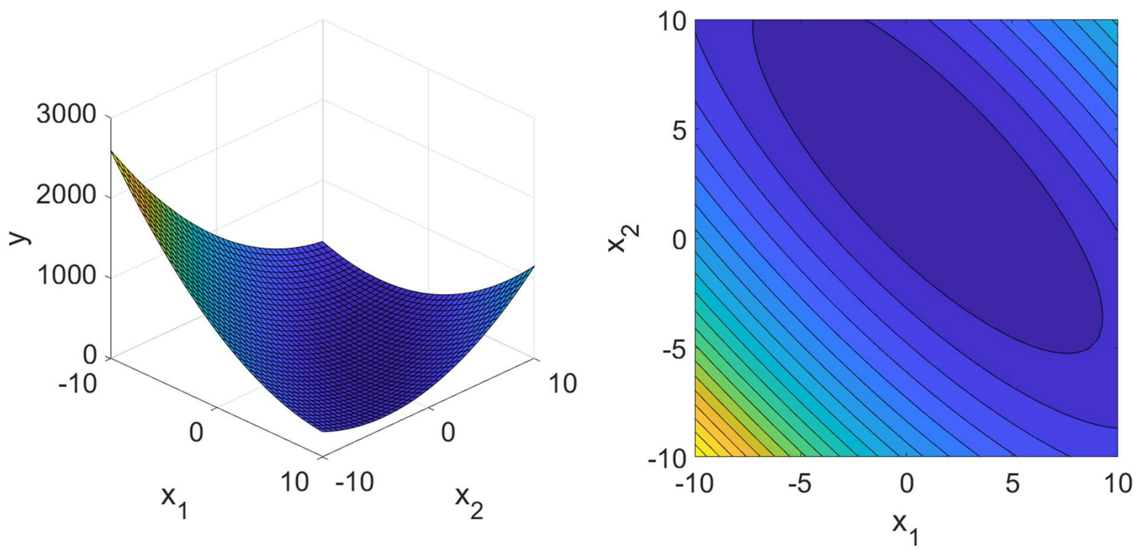
Celem ćwiczenia jest zapoznanie się z gradientowymi metodami optymalizacji poprzez ich implementację oraz wykorzystanie do wyznaczenia minimum podanej funkcji celu.

2. Testowa funkcja celu.

Funkcja celu dana jest wzorem:

$$f(x_1, x_2) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2 - 5)^2$$

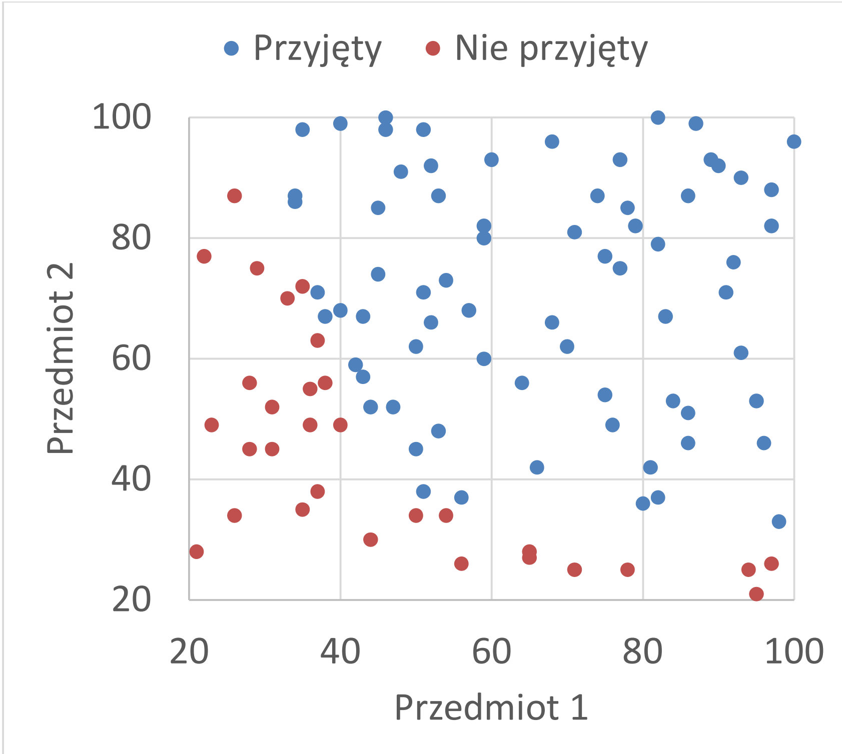
Jej wykres przedstawiony jest poniżej.



Punkt startowy powinien należeć do przedziału $x_1^{(0)} \in [-10, 10]$, $x_2^{(0)} \in [-10, 10]$.

3. Problem rzeczywisty.

O przyjęciu na pewną uczelnię decydują oceny uzyskane z dwóch przedmiotów. Na poniższym rysunku przedstawiono przykładowe oceny wraz z decyzją o przyjęciu lub odrzuceniu kandydata.



Celem optymalizacji jest znalezienie optymalnych parametrów klasyfikatora, który na podstawie otrzymanych ocen, będzie w stanie przewidzieć, czy dana osoba zostanie przyjęta na uczelnie. Klasyfikator będzie wykorzystywał hipotezę postaci:

$$h_\theta(\mathbf{x}) = \frac{1}{1 + e^{-\boldsymbol{\theta}^T \mathbf{x}}}$$

gdzie: wektor $\boldsymbol{\theta} = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \end{bmatrix}$ zawiera szukane parametry klasyfikatora, wektor $\mathbf{x} = \begin{bmatrix} 1 \\ x_1 \\ x_2 \end{bmatrix}$ zawiera oceny z przedmiotu 1 (x_1) oraz przedmiotu 2 (x_2).

Klasyfikator wskaże, że *i*-ty kandydat zostanie przyjęty na uczelnię, jeżeli wartość hipotezy będzie większa lub równa 0,5, tj.:

$$h_\theta(\mathbf{x}^i) \geq 0{,}5 \implies y^i = 1$$

Dane uczące (przedstawione na rysunku) są dostępne w plikach XData.txt oraz YData.txt w postaci macierzy X oraz Y:

$$X = [\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^m], \mathbf{x}^i = \begin{bmatrix} 1 \\ x_1^i \\ x_2^i \end{bmatrix}, i = 1, \dots, m$$

$$Y = [y^1, y^2, \dots, y^m],\ y^i = \{0{,}1\}, i = 1, \dots, m$$

gdzie: liczba danych $m = 100$, $y^i = 1$ oznacza, że *i*-ty kandydat został przyjęty, $y^i = 0$ oznacza, że *i*-ty kandydat nie został przyjęty .

Poszukiwanie optymalnych parametrów klasyfikatora odbywa się poprzez minimalizację funkcji kosztu danej wzorem:

$$J(\boldsymbol{\theta}) = -\frac{1}{m}\sum_{i=1}^{m}\left(y^i \cdot \ln\left(h_\theta(\mathbf{x}^i)\right) + (1 - y^i) \cdot \ln\left(1 - h_\theta(\mathbf{x}^i)\right)\right)$$

Pochodne cząstkowe funkcji kosztu $J(\boldsymbol{\theta})$ wynoszą:

$$\frac{\partial J(\boldsymbol{\theta})}{\partial \theta_j} = \frac{1}{m}\sum_{i=1}^{m}\left(h_\theta(\mathbf{x}^i) - y^i\right) \cdot x_j^i \text{ dla } j = 1, \dots, n$$

gdzie: n – liczba współrzędnych wektora gradientu.

W celu sprawdzenia poprawności implementacji funkcji kosztu oraz jej gradientu, można obliczyć ich wartości dla wektora $\boldsymbol{\theta} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$. Poprawne wartości wynoszą w przybliżeniu: $J(\boldsymbol{\theta}) = 0{,}6931$ oraz $\nabla J(\boldsymbol{\theta}) = \begin{bmatrix} -0{,}2 \\ -16{,}405 \\ -18{,}29 \end{bmatrix}$.

4. Algorytmy optymalizacji.

Do wyznaczenia minimum funkcji celu należy zastosować:

a. dla testowej funkcji celu metody najszybszego spadku, gradientów sprzężonych oraz Newtona, każdą w wersji stałokrokowej i zmiennokrokowej (długości kroku w wersji stałokrokowej należy przyjąć równe 0,05 oraz 0,12, w wersji zmiennokrokowej długość kroku należy wyznaczyć metodą złotego podziału),
b. dla problemu rzeczywistego metodę gradientów sprzężonych w wersji stałokrokowej (długości kroku należy przyjąć równe 0,01, 0,001 oraz 0,0001).

5. Zadanie do samodzielnego wykonania.

a. Testowa funkcja celu.

Zadanie polega na wykonaniu 100 optymalizacji dla każdej długości kroku startując z losowego punktu. Wyniki należy zestawić pliku xlsx w tabeli 1. Wartości średnie należy przedstawić w tabeli 2. Dodatkowo, dla jednego wybranego punktu startowego należy narysować 6 wykresów. Na każdym, na wykres poziomic należy nanieś rozwiązania optymalne uzyskane po każdej iteracji:

- wykres 1 – dla długości kroku równej 0,05 rozwiązania otrzymane każdą z metod,
- wykres 2 – dla długości kroku równej 0,12 rozwiązania otrzymane każdą z metod,
- wykres 3 – dla wersji zmiennokrokowej rozwiązania otrzymane każdą z metod,
- wykres 4 – dla metody najszybszego spadku rozwiązania otrzymane dla każdej długości kroku,
- wykres 5 – dla metody gradientów sprzężonych rozwiązania otrzymane dla każdej długości kroku,
- wykres 6 – dla metody Newtona rozwiązania otrzymane dla każdej długości kroku.

b. Problem rzeczywisty.

Zadanie polega na wykonaniu jednej optymalizacji dla każdej długości kroku startując z punktu $\boldsymbol{\theta}^{(0)} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$. Wyniki należy zestawić pliku xlsx w tabeli 3. W kolumnie $P(\boldsymbol{\theta}^*)$ należy podać jaki procent przypadków ze zbioru uczącego został poprawnie zaklasyfikowany, tj.:

$$P(\boldsymbol{\theta}^*) = \frac{1}{m}\sum_{i=1}^{m} \begin{cases} 1 & \text{jeżeli } \left[h_\theta(\mathbf{x}^i)\right] = y^i \\ 0 & \text{w przeciwnym przypadku} \end{cases}$$

gdzie: $[\cdot]$ oznacza zaokrąglenie liczby do najbliższej wartości całkowitej.

Dodatkowo, na wykres przedstawiający przyjęte/nie przyjęte osoby należy nanieść granicę klasyfikacji otrzymaną dla najlepszego przypadku (arkusz wykres). Granica klasyfikacji dana jest równaniem:

$$h_\theta(\mathbf{x}) = 0{,}5$$

W rozpatrywanym przypadku (przy założeniu, że $\theta_2 \neq 0$) granica klasyfikacji to funkcja o równaniu:

$$x_2 = -\frac{\theta_0 + \theta_1 x_1}{\theta_2}$$

6. Sprawozdanie

Sprawozdanie powinno zostać przygotowane w formacie docx (lub doc) albo pdf i powinno zawierać parametry poszczególnych algorytmów, dyskusję wyników oraz wnioski. Dodatkowo, w sprawozdaniu należy umieścić kod zaimplementowanych metod, funkcję lab4 oraz funkcje wykorzystane do obliczenia funkcji celu, gradientu oraz hesjanu. Wyniki optymalizacji oraz wykresy należy przygotować w formacie xlsx (lub xls).

Pseudokod metod gradientowych.

Dane wejściowe: punkt startowy $x^{(0)}$, dokładność $\varepsilon > 0$, maksymalna liczba wywołań funkcji celu N_{max}

```
i = 0
repeat
     wyznacz d(i)
     wyznacz h(i)
     x(i+1) = x(i) + h(i)·d(i)
     i = i + 1
     if fcalls > Nmax then
          return error
     end if
until ||x(i) – x(i-1)||2 < ε
return x* = x(i)
```

Kierunek $d^{(i)}$ wyznacza się ze wzoru:

- dla metody najszybszego spadku: $d^{(i)} = -\nabla f\left(x^{(i)}\right)$
- dla metody gradientów sprzężonych:

$$d^{(0)} = -\nabla f\left(x^{(0)}\right)$$

$$d^{(i)} = -\nabla f\left(x^{(i)}\right) + \beta \cdot d^{(i-1)}$$

$$\beta = \frac{\left(\left\|\nabla f\left(x^{(i)}\right)\right\|_2\right)^2}{\left(\left\|\nabla f\left(x^{(i-1)}\right)\right\|_2\right)^2}$$

- dla metody Newtona: $d^{(i)} = -H^{-1}\left(x^{(i)}\right)\nabla f\left(x^{(i)}\right)$

Pseudokod metody złotego podziału.

Dane wejściowe: przedział poszukiwań `[a, b]`, dokładność obliczeń `ε> 0`, maksymalna liczba wywołań funkcji celu N_{max}

```
i = 0
α = (5^0,5 – 1) / 2
a^(0) = 0, b^(0) = b
c^(0) = b^(0) – α(b^(0) – a^(0))
d^(0) = a^(0) + α(b^(0) – a^(0))
repeat
     if f(c^(i)) < f(d^(i)) then
          a^(i+1) = a^(i)
          b^(i+1) = d^(i)
          c^(i+1) = b^(i+1) - α(b^(i+1) – a^(i+1))
          d^(i+1) = c^(i)
     else
          a^(i+1) = c^(i)
          b^(i+1) = b^(i)
          c^(i+1) = d^(i)
          d^(i+1) = a^(i+1) + α(b^(i+1) – a^(i+1))
     end if
     i = i + 1
     if f_calls > N_max then
          return error
     end if
until b^(i) – a^(i) < ε
return x* = (a^(i) + b^(i)) / 2
```