

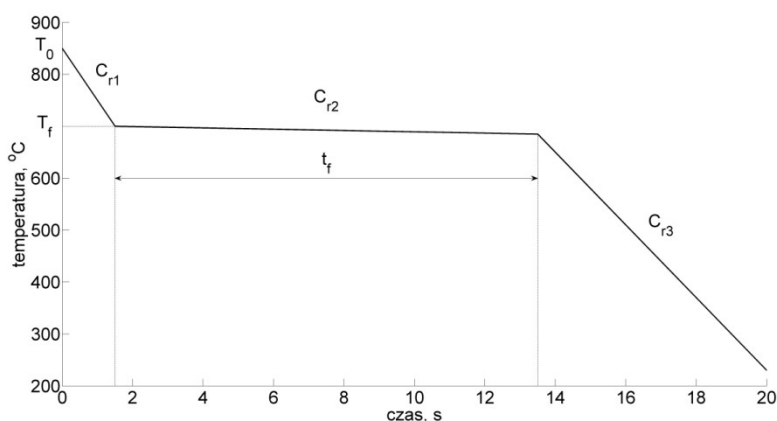
Regresja logistyczna cz. 1

1. Cel ćwiczenia.

Celem ćwiczenia jest zapoznanie się z regresją logistyczną poprzez implementację odpowiednich funkcji oraz wykorzystanie ich budowy modelu rzeczywistego procesu.

2. Proces rzeczywisty i dane uczące.

Blachy po walcowaniu na gorąco poddawane są procesowi chłodzenia laminarnego. W zależności od sposobu chłodzenia możliwe jest uzyskanie stali o różnych własnościach (w zależności od składu fazowego). Przykładowy schemat chłodzenia został przedstawiony na rysunku 1.



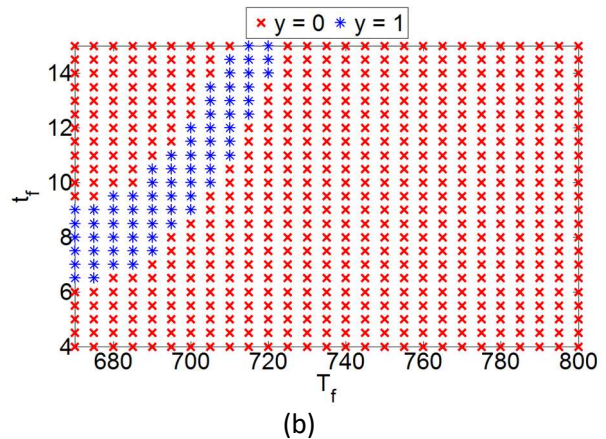
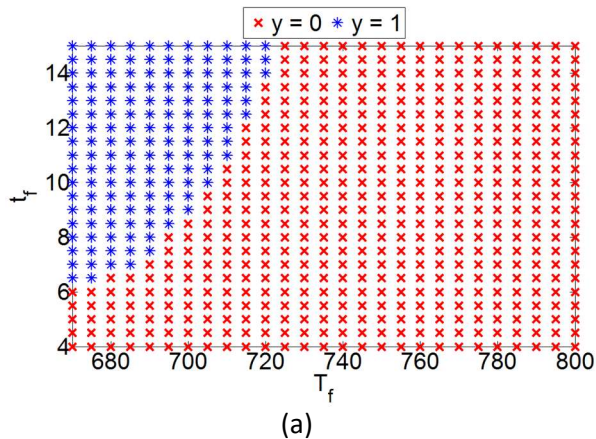
Rysunek 1. Przykładowy schemat słodzenia blach po walcowaniu na gorąco.

Jest on zdefiniowany przez pięć parametrów:

- prędkość chłodzenia w pierwszym etapie c_{r1} ,
- temperatura początku drugiego etapu T_f ,
- czas trwania drugiego etapu t_f ,
- prędkość chłodzenia w drugim etapie c_{r2} ,
- prędkość chłodzenia w trzecim etapie c_{r3} .

Prędkości chłodzenia w poszczególnych etapach są równe: $c_{r1} = 100\text{ }^{\circ}\text{C}/\text{s}$, $c_{r2} = 1,25\text{ }^{\circ}\text{C}/\text{s}$, $c_{r3} = 70\text{ }^{\circ}\text{C}/\text{s}$. Temperatura początku drugiego etapu może przyjmować wartości z zakresu $T_f \in [670, 800]^{\circ}\text{C}$, a czas jego trwania może wahać się w zakresie $t_f \in [4, 15]\text{s}$.

Na rysunku 2 pokazano czy udział ferrytu w stali po schłodzeniu należy do przedziału $F_f \in [0,7; 0,85]$ (a) oraz czy należy do przedziału $F_f \in [0,7; 0,8]$ (b).



3. Zadanie do samodzielnego wykonania.

Celem ćwiczenia jest stworzenie dwóch modeli (θ^1 oraz θ^2) potrafiących przewidzieć, czy udział fazowy ferrytu mieści się zakresie 1: $F_f \in [0,7; 0,8]$ oraz czy mieści się w zakresie 2: $F_f \in [0,7; 0,85]$. W tym celu należy napisać/dostosować następujące skrypty/funkcje:

1. **RunML2.m** - skrypt, który wypisuje imię i nazwisko Autora wykonuje wszystkie obliczenia oraz przedstawia wyniki: postać hipotezy, wartość funkcji kosztu, dokładność predykcji dla zbioru testowego, macierz błędów, wykres nauczanej zależności.
2. **ReadData.m** - funkcja odczytująca dane z pliku. Wywołanie $[X, Y] = \text{ReadData}$;
3. **MapFea.m** - funkcja mapująca cech (tworząca nowe cechy w oparciu o już istniejące). Wywołanie $X = \text{MapFea}(X)$;
4. **NormFea.m** - funkcja wykonująca standaryzację cech. Wywołanie $[X, \mu, \sigma] = \text{NormFea}(X, \mu, \sigma)$;
5. **SplitData.m** - funkcja dzieląca dane na zbiory: uczący, walidacyjny i testowy w podanych stosunku. Wywołanie $[X_{tr}, Y_{tr}, X_{val}, Y_{val}, X_{te}, Y_{te}] = \text{SplitData}(X, Y, \text{ratio})$;
6. **sigmoid.m** - funkcja obliczająca wartość funkcji sigmoidalnej. Wywołanie $Y = \text{sigmoid}(X)$;
7. **CostFun.m** - funkcja obliczająca funkcję kosztu oraz gradient funkcji kosztu. Wywołanie $[J, dJ] = \text{CostFun}(X, Y, \text{Theta})$;
8. **NumGrad.m** - funkcja obliczająca gradient numerycznie. Wywołanie: $\text{NumdJ} = \text{NumGrad}(X, Y, \text{Theta})$;
9. **FindTheta.m** - funkcja przeprowadzająca optymalizację wykorzystując procedurę fminunc. Wywołanie $[\text{ThetaOpt}, J_{\text{Opt}}] = \text{FindTheta}(\text{Theta0}, X, Y)$;
10. **PlotData.m** - funkcja rysująca punkty na płaszczyźnie z uwzględnieniem klasy, do której należy punkt. Wywołanie $\text{PlotData}(X, Y)$;
11. **PlotBoundry.m** - funkcja rysującą granicę klasyfikacji. Wywołanie $\text{PlotBoundry}(X, Y, \text{Theta}, \mu, \sigma)$;
12. **ConfMatrix.m** - funkcja zwracająca macierz błędów. Wywołanie: $\text{CM} = \text{ConfMatrix}(Y_{te}, Z)$;

Zakładając, że wektor \mathbf{x} po mapowaniu cech przyjmuje postać: $\mathbf{x} = [1 \quad T_f \quad t_f \quad T_f^2 \quad t_f^2]^T$ wektory $\boldsymbol{\mu}$ oraz $\boldsymbol{\sigma}$ wynoszą w przybliżeniu odpowiednio:

$$\boldsymbol{\mu} = [1 \quad 735 \quad 9,5 \quad 541742 \quad 101,25]^T$$

$$\boldsymbol{\sigma} = [0 \quad 38,98 \quad 3,32 \quad 57310 \quad 63,83]^T$$

Dla wektora $\boldsymbol{\theta} = [1 \quad 1 \quad 1 \quad 1 \quad 1]^T$ wartość funkcji kosztu oraz jej gradientu wynoszą:

- dla zakresu 1: $J(\boldsymbol{\theta}) = 1,9321$, $\nabla J(\boldsymbol{\theta}) = [0,3821 \quad 0,5089 \quad 0,0913 \quad 0,5054 \quad 0,0937]^T$
- dla zakresu 2: $J(\boldsymbol{\theta}) = 1,9812$, $\nabla J(\boldsymbol{\theta}) = [0,4997 \quad 0,3559 \quad 0,2075 \quad 0,3548 \quad 0,2128]^T$

4. Przesłanie kodu.

Napisane skrypty/funkcje wraz z plikiem zawierającym dane należy spakować do pliku *nazwisko.zip* (nazwisko bez polskich znaków) i przesłać poprzez system UPeL.