Logistická regresia a klasifikácia dát

Dátová veda: tím Moja mama optimalizuje lepšie ako tvoja

Adrián Pauer - 1/5
Dávid Števaňák - 1/5
Ján Kamas - 1/5
Lukáš Jankola - 1/5
Peter Franček - 1/5

Fakulta matematiky, fyziky a informatiky
Univerzita Komenského
Máj 2023

Obsah

Kapitola A

Odvodenie predpisu účelovej funkcie

$$J(x) = -\sum_{i=1}^{m} v_i \cdot \ln(g(x^T u_i)) + (1 - v_i) \cdot \ln(1 - g(x^T u_i))$$

$$g(x^T u) = \frac{1}{1 + e^{-x^T u_i}}$$

Pomocou pravidiel logaritmovania dostávame nasledovné odvodenie:

1.
$$J(x) = -\sum_{i=1}^{m} v_i \cdot \ln(\frac{1}{1 + e^{-x^T u_i}}) + (1 - v_i) \cdot ln(1 - \frac{1}{1 + e^{-x^T u_i}})$$

2.
$$J(x) = -\sum_{i=1}^{m} v_i \cdot (\ln(\frac{1}{1 + e^{-x^T u_i}}) - \ln(\frac{e^{-x^T u_i}}{1 + e^{-x^T u_i}})) + ln(\frac{e^{-x^T u_i}}{1 + e^{-x^T u_i}})$$

3.
$$J(x) = -\sum_{i=1}^{m} v_i \cdot (\ln(\frac{1}{e^{-x^T u_i}})) + ln(e^{-x^T u_i}) - \ln(1 + e^{-x^T u_i})$$

4.
$$J(x) = -\sum_{i=1}^{m} v_i \cdot (x^T u_i) - x^T u_i - \ln(1 + e^{-x^T u_i})$$

5.
$$J(x) = -\sum_{i=1}^{m} x^T u_i \cdot (v_i - 1) - \ln(1 + e^{-x^T u_i})$$

6.
$$J(x) = \sum_{i=1}^{m} x^T u_i \cdot (1 - v_i) + \ln(1 + e^{-x^T u_i})$$

QED

Kapitola B

Odvodenie gradientu účelovej funkcie

Derivácia účelovej funkcie pomocou vektorového derivovania

$$\nabla J(x) = \sum_{i=1}^{m} u_i \cdot (1 - v_i) - u_i \cdot (\ln(\frac{e^{-x^T u_i}}{1 + e^{-x^T u_i}}))$$

Kapitola C

Riešenie úlohy (3) pomocou DFP a BFGS + približne optimálny krok

C.1 BFGS + približne optimálna dĺžka kroku

Úlohu sme riešili pomocou metódy BFGS s iteračnou schémou $x_{k+1} = x_k + smer \cdot krok$. Veľkosť kroku sme hľadali pomocou metódy backtracking_linesearch().
Vyšledok:

optimálne riešenie	**metóda pre krok**	**počet iterácií**	**čas (s)**
-0.04536027 0.32766468 0.34624817	backtracking	8	3.40

C.2 DFP + optimálna dĺžka kroku

Úlohu sme riešili pomocou metódy DFP s iteračnou schémou $x_{k+1} = x_k + smer \cdot krok$. Veľkosť kroku sme hľadali pomocou metódy backtracking_linesearch().
Vyšledok:

optimálne riešenie	**metóda pre krok**	**počet iterácií**	**čas (s)**
-0.04536013 0.32766552 0.34624645	backtracking	9	4.24

Obidve metódy našli približne identické riešenie, metóda BFGS potrebovala o 1 iteraćiu menej. Čas potrebný pre obidve metódy je približne rovnaký.

Kapitola D

Riešenie úlohy (3) pomocou DFP a BFGS + optimálny krok

D.1 BFGS + približne optimálna dĺžka kroku

Úlohu sme riešili pomocou metódy BFGS s iteračnou schémou $x_{k+1} = x_k + smer \cdot krok$. Veľkosť kroku sme hľadali pomocou metódy bisection() s hranicami intervalu, na ktorom sa hľadá krok : $(0, 0.5)$.
Vyśledok:

optimálne riešenie	metóda pre krok	počet iterácií	čas (s)
-0.04536012 0.32766542 0.34624637	bisection	22	42.16

D.2 DFP + optimálna dĺžka kroku

Úlohu sme riešili pomocou metódy DFP s iteračnou schémou $x_{k+1} = x_k + smer \cdot krok$. Veľkosť kroku sme hľadali pomocou pomocou metódy bisection() s hranicami intervalu, na ktorom sa hľadá krok : $(0, 0.5)$.
Vyśledok:

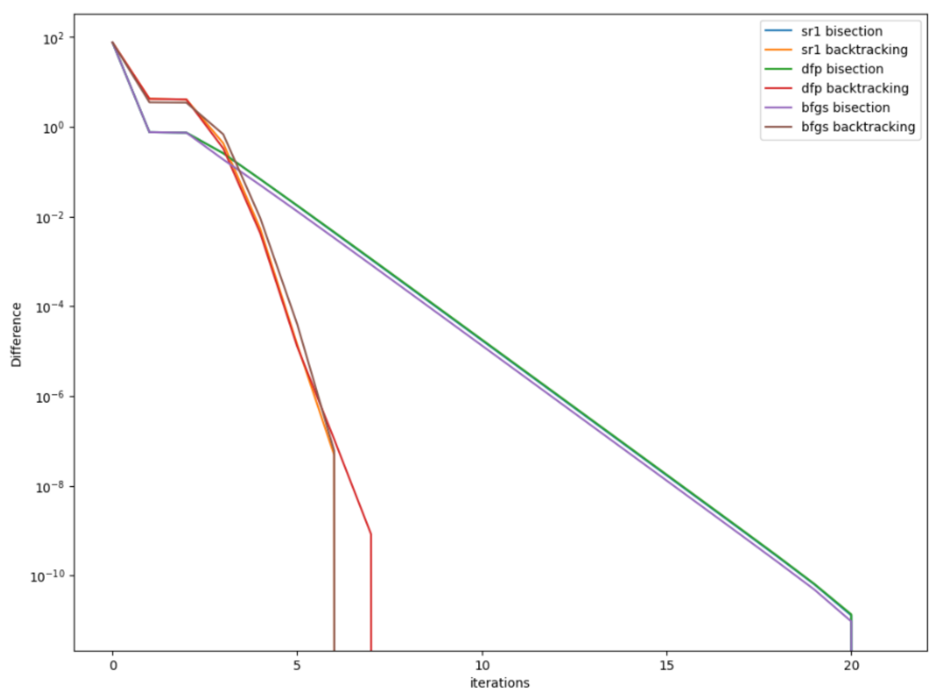
optimálne riešenie	metóda pre krok	počet iterácií	čas (s)
-0.04536012 0.32766541 0.34624637	bisection	22	42.98

Obidve metódy našli približne identické riešenie, v oboch prípadoch potrebovali viac iterácii ako metódy v predošlej úlohe. Metódy bežali približne rovnako dlho.

Kapitola E

Porovnanie iterácii pre kvázinewtonovské metódy

Vykreslili sme do jedného grafu ako postupne rôzne metódy konvergovali k nájdenému riešeniu. Kvôli veľkým rozdielom sme použili logaritmickú škálu na y - ovej osi. Z grafu vidíme, že všetky metódy, ktoré používajú na nájdenie optimálnej dĺžky kroku tú istú metódu konvergujú takmer rovnako. Vidíme, že ak sme použili bisekciu, tak sme potrebovali približne 20 iterácií a pri backtrackingu medzi 5 až 10. Toto môže byť spôsobené aj tým, že pri backtrackingu sme dovolili aj väčšie hodnoty dĺžky kroku (pri bisekcii by nám to pretieklo).

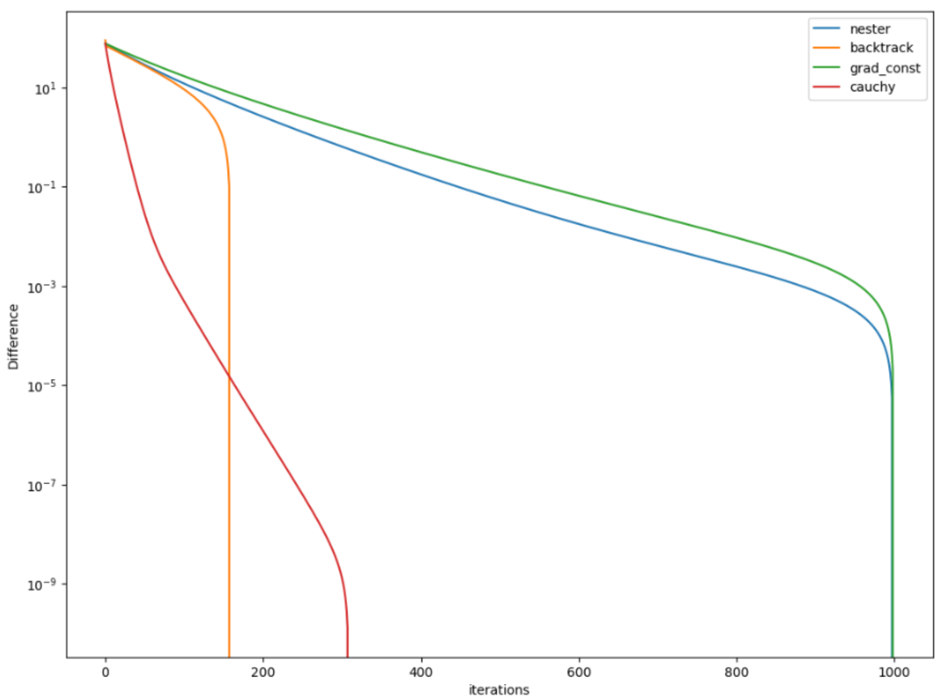


Obr. E.1: Porovnanie kvázinewtonovských metód

Kapitola F

Porovnanie iterácii pre gradientné metódy

Podobný graf ako v minulej úlohe sme vykreslili pre gradientné metódy. Gradientná metóda s konštantným krokom ani Nesterova metóda nám neskonvergovali ani po 1000 iteráciách a konvergovali dosť podobne. Najrýchlejšie skonvergovala gradientná metóda, kde hľadáme optimálnu dĺžku kroku backtrackingom (už po približne 190 iteráciách). Treba si ale uvedomiť, že časovo mohla kľudne zabrať viac, keďže v každej iterácií sa spúšťal backtracking, ktorý má tiež niekoľko iterácií. Cauchyho metóda zo začiatku konvergovala ešte rýchlejšie, lenže skonvergovala až po približne 300 iteráciách, ale platí to isté, čo predtým, až na to, že tu sme použili metódu bisekcie na nájdenie dĺžky kroku.



Obr. F.1: Porovnanie gradientných metód

Kapitola G

Klasifikácia dát

V tejto časti sme porovnali nájdené parametre $x \in \mathbb{R}^3$ pomocou vyššie uvedených metód a určili úspešnosť klasifikácie solventného a nesolventného klienta pomocou funkcie

$$g(x^T u) = \frac{1}{1 + e^{-x^T u}}$$

teda či pravdepodobnosť je vyššia ako 50%.
Porovnanie funkcií a ich úspešnosť je zhrnuté v nasledovnej tabuľke:

Metóda	**Nájdené parametre** (x_1, x_2, x_3)	**Úspešnosť klasifikácie**
BFGS(backtracking pre optimálny krok)	[-0.04536027, 0.32766468, 0.34624817]	71,4286%
BFGS(bisekcia pre optimálny krok)	[-0.04536012, 0.32766542, 0.34624637]	71,4286%
DFP(backtracking pre optimálny krok)	[-0.04536013, 0.32766552, 0.34624645]	71,4286%
DFP(bisekcia pre optimálny krok)	[-0.04536012, 0.32766541, 0.34624637]	71,4286%
Nester	[-0.0453397, 0.32499705, 0.34757128]	71,4286%
Backtracking	[-0.02416365, 0.20511938, 0.23394588]	74,4186%
Grad const	[-0.04519898, 0.32256955, 0.34771353]	71,4286%
Cauchy	[-0.04536018, 0.3276618, 0.34624906]	71,4286%

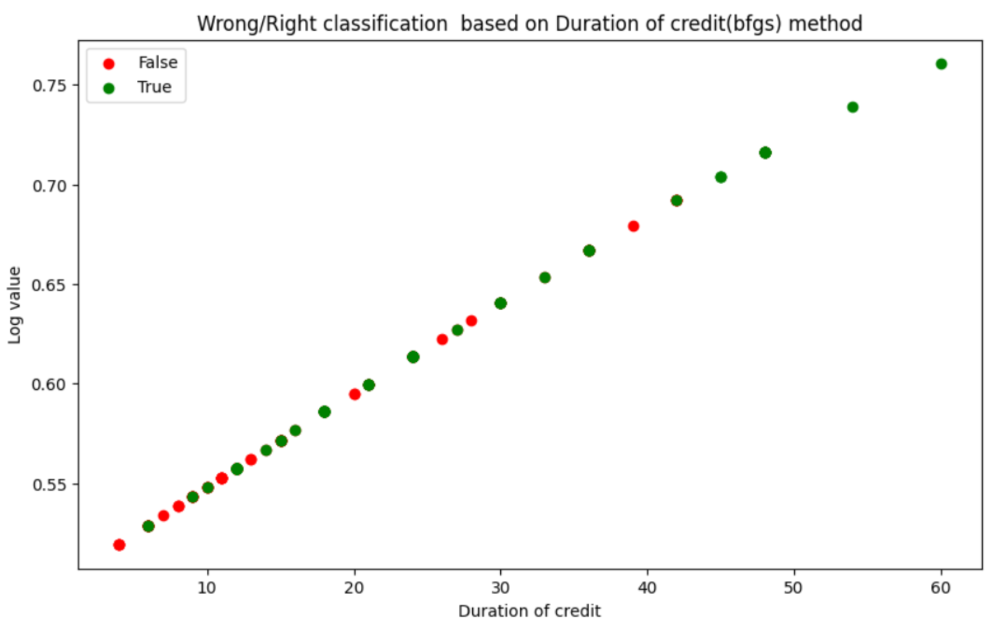
Tabuľka G.1: Úspešnosť klasifikácie pre nájdené $x \in \mathbb{R}^3$

Z tabuľky je vidieť, že najvyššia úspešnoť nastáva pri metóde *Backtracking*, avšak nie príliš výrazne. Taktieňájdené optimá nie sú príliš odlišné.

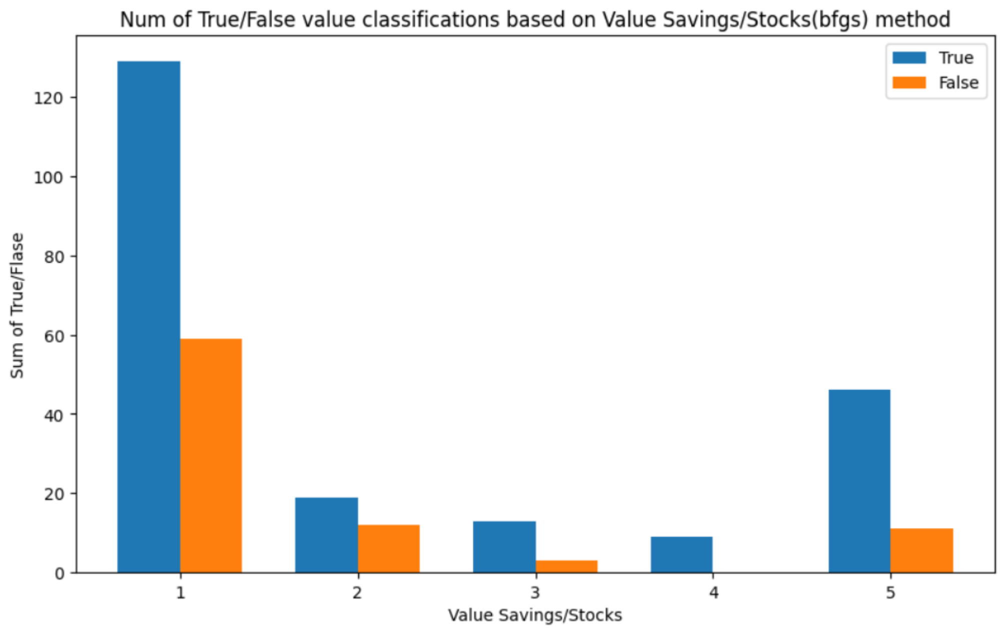
Kapitola H

Nadstavba

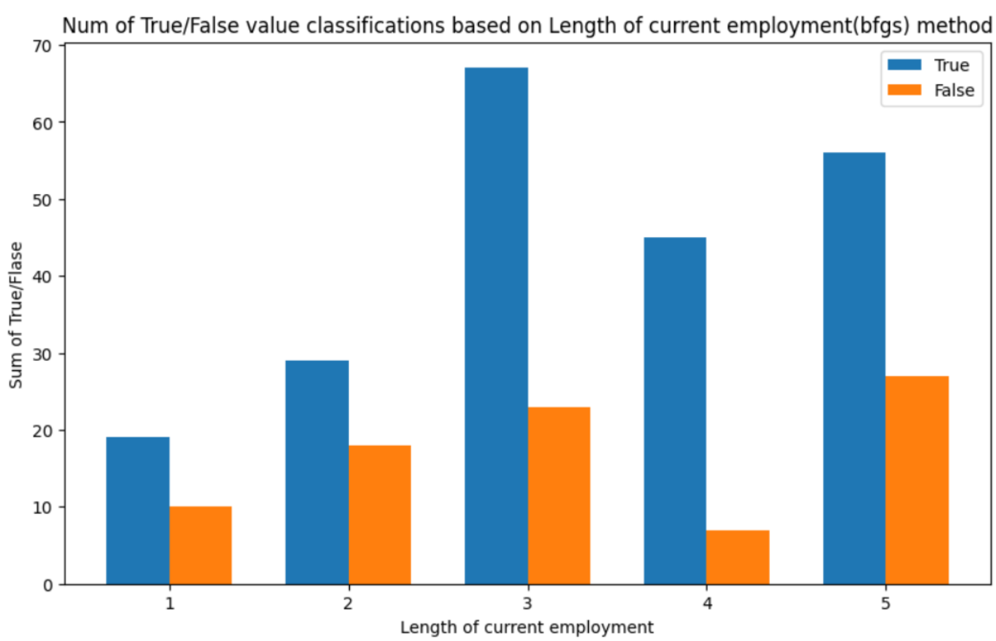
Modifikovali sme projekt tak, že sme binárnu premennú v modelovali len pomocou jedného ukazovateľa. Výsledky tejto klasifikácie sme vykreslili v grafe. Zaujímavé je, že úspešnosť pri modelovaní pomocou len jedného ukazovateľa dopadla porovnateľne ako pri modelovaní pomocou troch. Vo všetkých sme to porovnávali pomocou bfgs metódy.



Obr. H.1: Ne/správna klasifikácia na základe ukazovateľa Duration of credits



Obr. H.2: Počet správnych/nesprávnych klasifikácii na základe ukazovateľa Value Savings/Stocks



Obr. H.3: Počet správnych/nesprávnych klasifikácii na základe ukazovateľa Length of current employment