Fakulta matematiky, fyziky a informatiky Univerzity Komenského, Bratislava

Projekt z metód voľnej optimalizácie

Logistická regresia pomocou kvázinewtonovských metód – predikcia solventnosti klientov

Piati za optimalizáciu
Tomáš Antal, 2DAV, 0.23
Erik Božík, 2DAV, 0.23
Róbert Kendereš, 2DAV, 0.08
Teo Pazera, 2DAV, 0.23
Andrej Špitalský, 2DAV, 0.23

Obsah

0 Predstavenie témy **2**
0.1 Zavedenie značenia 2

1 Odvodenie účelovej funkcie a jej gradientu **3**
1.1 Kompaktnejší tvar účelovej funkcie 3
1.2 Gradient účelovej funkcie 4

2 Riešenie optimalizačnej úlohy **5**
2.1 Kvázinewtonovské metódy 5
2.2 Gradientné metódy 5

3 Vizualizácia konvergencie **7**

4 Binárna klasifikácia solventnosti klientov **8**

5 Nadstavba - všeobecný model pre logistickú regresiu pomocou kvázinewtonovských alebo gradientných metód **9**

6 Záver a diskusia **10**

7 Prehľad kódu **11**

0 Predstavenie témy

0.1 Zavedenie značenia

- $m = 699$ značí počet klientov, o ktorých máme dáta
- $v \in \mathbb{R}^m$, i-ta zložka má hodnotu 1, ak je klient i solventný, inak 0
- $u_j \in \mathbb{R}^m$, $j = 1, 2, 3$, vektory údajov o klientoch
  - u_1 – počet mesiacov od otvorenia účtu
  - u_2 – pomer úspor a investícií
  - u_3 – počet rokov v súčasnom zamestnaní
- v^i, u_j^i označujú i-te položky jednotlivých vektorov pre $i = 1, \ldots, m$, $j = 1, 2, 3$

1 Odvodenie účelovej funkcie a jej gradientu

V tejto časti sa budeme venovať odvodzovaniu účelovej funkcie a jej gradientu, ktorú v neskorších častiach budeme minimalizovať, pomocou čoho vytvoríme model na binárnu klasifikáciu.

Do logistickej funkcie $g(z) = \frac{1}{1+e^{-z}}$, ktorá bude odhadovať pravdepodobnosť solventnosti klienta, budeme dosádzať hodnoty $z = x^T u^i$ pre vektor parametrov $x = (x_0, \ldots, x_3)$ a vektor údajov o klientovi $u^i = (1, u_1^i, u_2^i, u_3^i)$, pre $i = 1, \ldots, m$.

Chceme odhadnúť zložky vektora x tak, aby čo najvierohodnejšie predpovedal solventnosť vzhľadom na naše dáta. To vedie k optimalizačnej úlohe:

$$\min J(x) \tag{1}$$

$$x \in \mathbb{R}^4 \tag{2}$$

kde

$$J(x) = -\sum_{i=1}^{m} v^i \ln\left(g\left(x^T u^i\right)\right) + (1 - v^i)\ln\left(1 - g\left(x^T u^i\right)\right)$$

Z predpisu funkcie si môžeme všimnúť, že suma nadobúda záporné hodnoty, čiže $J(x)$ nadobúda kladné hodnoty. Taktiež si môžeme všimnúť, že ak je klient i solventný, čiže $v^i = 1$ a pre nejaký vektor parametrov x je hodnota $g\left(x^T u^i\right)$ blízka nule, má to za následok „výrazné“ zvyšovanie hodnoty účelovej funkcie. Podobnou logikou vidíme zvyšovanie hodnoty účelovej funkcie pre nesolventných klientov, ak pomocou vektora x mu prisúdime veľkú pravdepodobnosť solventnosti hodnotou $g\left(x^T u^i\right)$. Chceme teda nájsť taký vektor x, že $g\left(x^T u^i\right)$ bude blízke 1 pre solvetného klienta a blízke 0 pre nesolventného.

1.1 Kompaktnejší tvar účelovej funkcie

Pre lepšiu manipuláciu a neskoršiu implementáciu si zjednodušíme tvar účelovej funkcie nasledovne:

$$\begin{aligned}
J(x) &= -\sum_{i=1}^{m} v^i \ln\left(g\left(x^T u^i\right)\right) + (1 - v^i)\ln\left(1 - g\left(x^T u^i\right)\right) \\
&= -\sum_{i=1}^{m} v^i \ln\left(\left(1 + e^{-x^T u^i}\right)^{-1}\right) + (1 - v^i)\ln\left(\frac{e^{-x^T u^i}}{1 + e^{-x^T u^i}}\right) \\
&= -\sum_{i=1}^{m} -v^i \ln\left(1 + e^{-x^T u^i}\right) + (1 - v^i)\left(\ln\left(e^{-x^T u^i}\right) - \ln\left(1 + e^{-x^T u^i}\right)\right) \\
&= -\sum_{i=1}^{m} -v^i \ln\left(1 + e^{-x^T u^i}\right) - (1 - v^i)x^T u^i - (1 - v^i)\ln\left(1 + e^{-x^T u^i}\right) \\
&= \sum_{i=1}^{m} (1 - v^i)x^T u^i + \ln\left(1 + e^{-x^T u^i}\right)
\end{aligned}$$

S takýmto vyjadrením funkcie $J(x)$ budeme pracovať v nasledujúcich častiach.

1.2 Gradient účelovej funkcie

Vyjadríme si najprv parciálnu deriváciu podľa x_0, potom podľa $x_j, j = 1, 2, 3$, keďže tie sa správajú symetricky.

$$\begin{aligned}
\frac{\partial}{\partial x_0} J(x) &= \frac{\partial}{\partial x_0} \sum_{i=1}^{m} (1 - v^i) x^T u^i + \ln\left(1 + e^{-x^T u^i}\right) \\
&= \sum_{i=1}^{m} \frac{\partial}{\partial x_0} \left((1 - v^i)(x_0 + x_1 u_1^i + x_2 u_2^i + x_3 u_3^i) + \ln\left(1 + e^{-x^T u^i}\right)\right) \\
&= \sum_{i=1}^{m} (1 - v^i) - \frac{e^{-x^T u^i}}{1 + e^{-x^T u^i}} \\
&= \sum_{i=1}^{m} 1 - v^i - \frac{1}{1 + e^{x^T u^i}}
\end{aligned}$$

$$\begin{aligned}
\frac{\partial}{\partial x_j} J(x) &= \frac{\partial}{\partial x_j} \sum_{i=1}^{m} (1 - v^i) x^T u^i + \ln\left(1 + e^{-x^T u^i}\right) \\
&= \sum_{i=1}^{m} \frac{\partial}{\partial x_j} \left((1 - v^i)(x_0 + x_1 u_1^i + x_2 u_2^i + x_3 u_3^i) + \ln\left(1 + e^{-x^T u^i}\right)\right) \\
&= \sum_{i=1}^{m} (1 - v^i) u_j^i - u_j^i \frac{e^{-x^T u^i}}{1 + e^{-x^T u^i}} \\
&= \sum_{i=1}^{m} \left(1 - v^i - \frac{1}{1 + e^{x^T u^i}}\right) u_j^i \qquad j = 1, 2, 3
\end{aligned}$$

Toto vieme kompaktne zapísať nasledovne:

$$\nabla J(x) = \sum_{i=1}^{m} \begin{pmatrix} 1 \\ u_1^i \\ u_2^i \\ u_3^i \end{pmatrix} \left(1 - v^i - \frac{1}{1 + e^{x^T u^i}}\right)$$

2 Riešenie optimalizačnej úlohy

V tejto časti sa venujeme riešeniu optimalizačnej úlohy 1 rôznymi metódami. Tie boli implementované v Pythone. Konkrétne sme implementovali gradientné metódy (s optimálnou a konštnantou dĺžkou kroku) a kvázinewtonovské metódy BFGS a DFP (s približne optimálnou dĺžkou kroku nájdenou backtracking-om alebo s optimálnou dĺžkou kroku, nájdenou bisekciou).

Ako štartovací bod sme pri každej metóde volili $x_0 = (0,0,0,0)^T$ a ako kritérium optimality bolo použité $||\nabla J(x^k)|| \leq 10^{-3}$. Optimálnym bodom bude teda vektor parametrov x, ktorý budeme používať v logistickej funkcii na odhadovanie solventnosti klienta podľa jeho dát.

2.1 Kvázinewtonovské metódy

Minimalizujeme 1 pomocout metód BFGS a DFP s optimálnym krokom (nájdeným bisekciou) a približne optimálnym krokom (nájdeným backtrackingom).

	x_0	x_1	x_2	x_3
BFGS + backtracking	0.20751015	-0.04712048	0.31535175	0.30654686
BFGS + bisekcia	0.20751337	-0.04712051	0.31535088	0.30654664
DFP + backtracking	0.20750999	-0.04712047	0.31535176	0.30654688
DFP + bisekcia	0.20751338	-0.04712052	0.31535087	0.30654663

Všetky štyri minimalizácie skonvergovali k minimu (vzhľadom na kritérium optimality) za menej ako 13 iterácií. Vidíme, že optimálne hodnoty všetkých štyroch minimalizácií sa odlišujú najskôr v ráde 10^{-5}, čiže môžeme predpokladať, že konvergujú k rovnakému bodu minima.

Môžeme si všimnúť, že pozitívny vplyv na pravdepodobnosť solventnosti klienta má druhý sledovaný parameter, čiže pomer úspor a investícií, a tretí sledovaný porameter, čiže počet rokov v súčasnom zamestnaní. Takisto si môžeme všimnúť, že počet mesiacov od otvorenia účtu (prvý sledovaný parameter), má na odhad pravdepodobnosti solventnosti klienta negatívny vplyv, čo je prekvapivý výsledok.

Nájdený koeficient x_0 má za následok to, že pre klienta, ktorého parametre sú $(0,0,0)^T$, po dosadení do logistickej funkcie dostaneme pravdepodobnosť solventnosti približne 0.5517.

Môžeme si takisto porovnať čas (v sekundách) potrebných na nájdenie minima pre jednotlivé metódy.

	čas[s]
BFGS + backtracking	0.0067
BFGS + bisekcia	0.0074
DFP + backtracking	0.0031
DFP + bisekcia	0.0069

Vidíme, že pre obidve implementované kvázinewtonovské metódy je implementácia s približne optimálnou dĺžkou rýchlejšia.

2.2 Gradientné metódy

Podobne ako vyššie, minimalizujeme 1 pomocou gradientnej metódy s optimálnym a s konštantným krokom. Na nájdenie optimálneho kroku používame bisekciu, ako konštantný krok používame $2 \cdot 10^{-5}$.

	x_0	x_1	x_2	x_3
optimálny krok	0.20742273	-0.04711977	0.31535679	0.30656397
konštantný krok	0.19322267	-0.0470058	0.31617533	0.30934507

Gradientná metóda s optimálnym krokom skonvergovala (vzhľadom na kritérium optimality) po rádovo 5000 iteráciách. Gradientná metóda s konštantným krokom nedokonvergovala (vzhľadom na kritérium optimality) ani po 10000 iteráciách (neboli sme experimentovaním nájsť vhodnú dĺžku kroku).

Signifikancia jednotlivých parametrov klientov je zhodná s tou, ktorá je popísaná vyššie. Takisto, pre klienta $(0, 0, 0)^T$ je odhad pravdepodobnosti solventnosti približne 0.5517 (pre optimálny krok), resp. 0.5482 (pre konštantný krok).

Rovnako ako vyššie, môžeme porovnať časy potrebné na minimalizáciu.

	čas[s]
optimálny krok	6.1772
konštantný krok	0.8142

Vidíme, že hľadanie optimálneho kroku v každej iterácií pridá približne 5.3 sekundy k času výpočtu, aj keď iterácií bolo rádovo polovica oproti konštantnému kroku. Skúsili sme preto nastaviť maximálny počet iterácií pre gradientnú metódu s konštatným krokom na 10^5. Už po rádovo 30000 iteráciách bolo dosiahnuté kritérium optimality a jeho nájdenie trvalo približne 0.9181 sekundy. Vidíme teda, že pri vysokom počte iterácií môže mať zmysel použiť skôr konštatný krok, keďže vieme výrazne ušetriť čas potrebný na hľadanie optimálneho kroku.

$$J^*_{GM\ const} = (0.20742016,\ -0.04711976,\ 0.31535694,\ 0.30656448)^T$$

3 Vizualizácia konvergencie

Čo sem spísať

1. opísať graf, log-škála
2. 2*2 grid pre KNM, 2*1 pre gradientné
3. popísať teoretický typ konvergencie
4. porovnať počet iterácii

4 Binárna klasifikácia solventnosti klientov

Čo sem spísať

1. vypísať výsledky úspešnosti jednotlivých metód
2. porovnať najlepšiu klasifikáciu s najhoršou

5 Nadstavba - všeobecný model pre logistickú regresiu pomocou kvázinewtonovských alebo gradientných metód

Čo sem spísať

1. popísať štruktúru modulu
2. mierne popísať funkčnosť a spúšťanie
3. spomenúť testy

6 Záver a diskusia

7 Prehľad kódu