Semestrální práce

KIV-SU

Strojové učení

Jakub Zíka - A15N0087P
zikaj@students.kiv.zcu.cz

31. ledna 2017

Obsah

1 Zadání **3**

1.1 Vybrané zadání 3

2 Teoretický úvod **3**

2.1 Support Vector Machines 3

2.1.1 Lineární rozhodovací hranice 3

2.1.2 Cenová funkce lineární hranice 4

2.1.3 Nelineární rozhodovací hranice 5

2.1.4 Cenová funkce nelineární hranice 6

3 Závěr **8**

1 Zadání

Navrhněte téma zadání semestrální práce související s oblastí strojového učení. Cílem práce je prohloubit znalosti studenta v oblasti kognititvníchsystémů pomocí nabytých zkušeností ze semetrální práce.

1.1 Vybrané zadání

Sestrojte klasifikátor *Support Vector Machines*, dále už jen *SVM*, který bude skrze osobní údaje pasažérů lodi Titanic klasifikovat, zda daná osoba přežije či nepřežije potopení lodi. Dále se pokuste najít souvislosti mezi jednotlivými údaji o pasažérech a z nich zjistit či odvodit, které mají na přežití největší vliv.

2 Teoretický úvod

2.1 Support Vector Machines

Algoritmy strojového se skládají z *trénovací množiny* a *rozhodovací hranice*. Rozhodovací hranici můžeme též nazývat *hypotéza*. Trénovací množina může obsahovat i správné odpovědi. Algoritmy tedy dělíme na *učení s učitelem* (máme odpovědi) a *učení bez učitele* (nemáme odpovědi).

Každý vzorek $\mathbf{x}$ trénovací množiny je tvořen množinou příznaků

$$x_1, x_2, ...x_n,$$

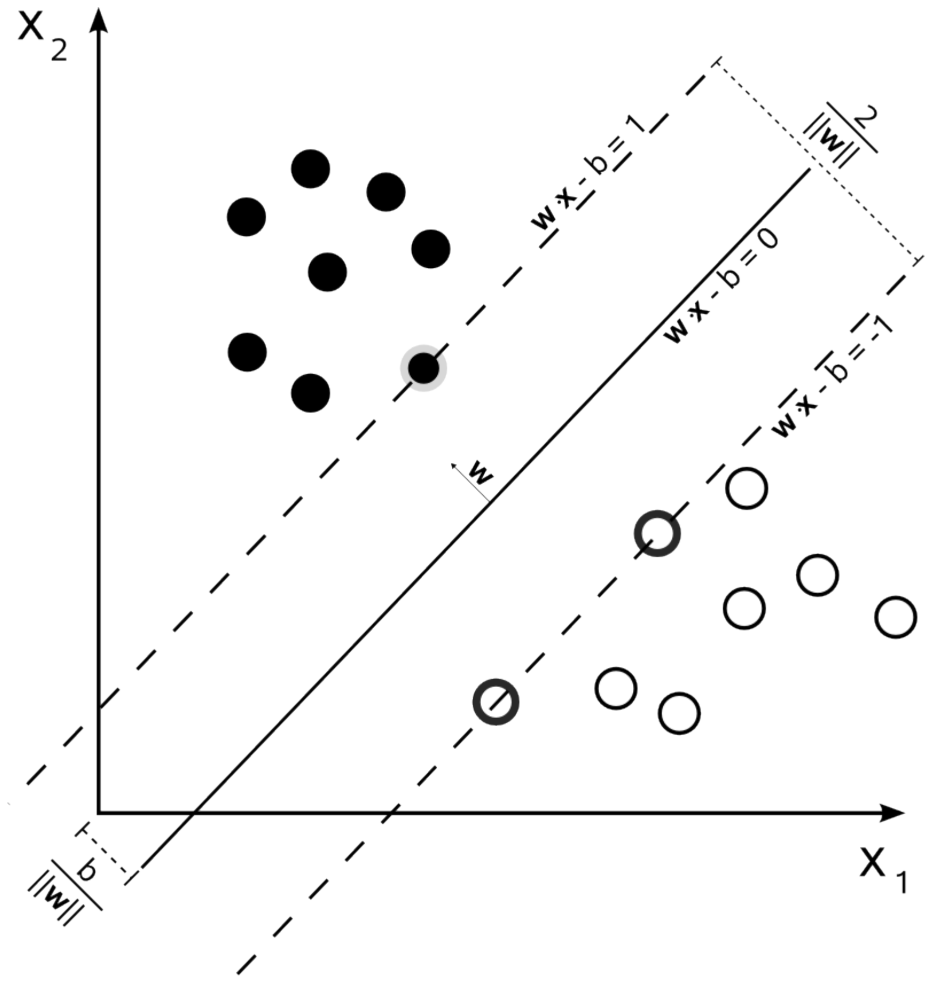
kde n je počet příznaků. Hypotéza má tvar *h:x-¿y* což znamená, že hypotéza je zobrazení x do y. Hypotézu tvoří modelovací parametry

$$\Theta_n,$$

které nám umožňují nastavit rozumnou rozhodovací hranici. Jejich hodnoty předem neznáme a získáme je trénováním.

2.1.1 Lineární rozhodovací hranice

Metoda strojového učení, která hledá v trénovací množině umístění optimální nadroviny. Tato nadrovina slouží k rozdělení bodů projekce na dvě třídy. V



Obrázek 1: Maximální pás bez bodů trénovací množiny [**?**]

tomto rozdělení je požadováno aby minimum vzdáleností bodů od této nadroviny bylo co největší. Chceme tedy, aby nadrovina měla po obou stranách co nejširší pás bez bodů. K popisu těchto pásů slouží pomocné vektory (*Support Vectors*) (viz Obr.:1). [1]

2.1.2 Cenová funkce lineární hranice

Metoda SVM je vylepšenou verzí *logistické regrese*. Ovšem, na rozdíl od cenové funkce logistické regrese nám SVM nevrací pravděpodobnost, ale rovnou příslušnost klasifikovaného vzorku k třídě 1 nebo 0. Zde vidíme *cenovou funkci* logistické regrese:

$$\min_{\theta} \frac{1}{m}[\sum_{i=1}^{m} y^{(i)}(-\log h_\theta(x^{(i)})) + (1-y^{(i)})(-\log(1-h_\theta(x^{(i)})))] + \frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2$$

Cenová funkce SVM pak vypadá následovně:

$$\min_{\theta} C\sum_{i=1}^{m}[y^{(i)}Cost_1(\theta^T x^{(i)}) + (1-y^{(i)})Cost_0(\theta^T x^{(i)})] + \frac{1}{2}\sum_{j=1}^{n}\theta_j^2,$$

kde *Cost* funkce pro y = 1 je

$$\log\frac{1}{1+e^{-\Theta^T x}}$$

a funkce *Cost* funkce pro y = 0 je

$$\log(1-\frac{1}{1+e^{-\Theta^T x}}).$$

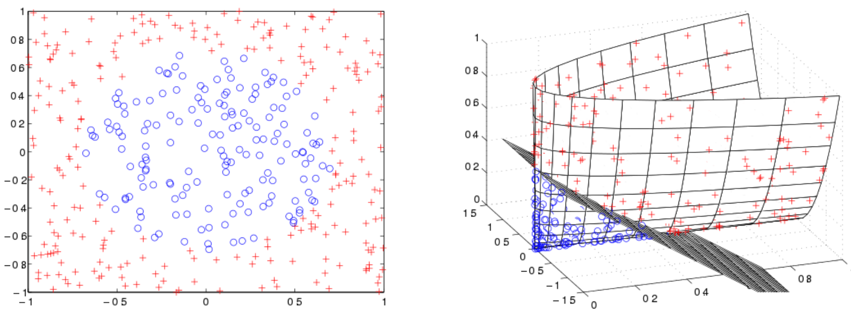
Optimalizací cenové funkce získáme hodnoty parametrů Θ, které určují tvar hypotézy.

Hodnota C je regularizační faktor, který ovlivňuje výběr hypotézy. Rozumně vybraná hodnota pak umožní dělat ve výběru hypotézy kompromisy v extrémních případech rodělení tříd v trénovací množině:

1. C - vysoké = malá odchylka, velký rozptyl (malá λ), hrozí *overfitting*
2. C - malé = velká odchylka, malý rozptyl (velká λ), hrozí *undefitting*

2.1.3 Nelineární rozhodovací hranice

Máme-li trénovací množinu, pro kterou je rozhodovací hranice nelineární (viz. Obr.:2), musíme použít metodu jader (*Kernels*). Zavedeme si i pomocných bodů tzv. *landmarky*. Každý landmark $l^{(i)}$ nese hodnotu třídy, do které patří. Ke



Obrázek 2: Ukázka lineárně neseparabilních a separabilních dat.[4]

každému prvku $x^{(n)}$ trénovací množiny spočteme podobnost f_i s každým landmarkem $l^{(i)}$. Pro každý prvek trénovací množiny tak dostaneme vektor podobnosti $f^{(n)}$. Podobnost počítáme následovně:

$$f_i(x^{(n)}, l^{(i)}) = exp(-\frac{||x^{(n)} - l^{(i)}||}{2\sigma^2})$$

vektor podobnosti pak bude vypadat:

$$f^{(n)} = [f_1, f_2, f_3, ..., f_i]$$

Jádrem se nazývá funkce počítání podobnosti. V tomto případě je jádro *Gaussové*. Parametr *sigma* nám určuje míru podobnosti. Máme i jiné funkce jádra jako například *Polynomiální* nebo *Lineární*. Lineární jádro je předchozí případ lineárně separabilních dat.[5],[4]

2.1.4 Cenová funkce nelineární hranice

Při použití gaussového jádra máme předpočítaný vektor podobnosti pro každý prvek trénovací množiny. To znamená, že vektor podobnosti může reprezentovat daný prvek trénovací množiny. Cenouvou funkci tedy můžeme pozměnit do tvaru:

$$\min_{\theta} C \sum_{i=1}^{m} [y^{(i)} Cost_1(\theta^T f^{(i)}) + (1 - y^{(i)}) Cost_0(\theta^T f^{(i)})] + \frac{1}{2} \sum_{j=1}^{n} \theta_j^2$$

Pozor, gaussové jádro je citlivé na velké rozdíly hodnot mezi jednotlivými příznaky. Je dobré příznakový vektor nejdříve naškálovat a až poté počítat cenovou funkci.

3 Závěr

Reference

[1] EKŠTEIN, Kamil. *Support Vector Machines* [online]. Plzeň, 2012 [cit. 2017-01-31]. Dostupné z: https://portal.zcu.cz/CoursewarePortlets2/DownloadDokumentu?id=123920. Přednášky k předmětu Strojové učení. Západočeská univerzita v Plzni.

[2] NG, Andrew. *CS229 Lecture notes - SVM* [online]. Standford, 2016 [cit. 2017-01-31]. Dostupné z: http://cs229.stanford.edu/notes/cs229-notes3.pdf. Přednášky k předmětu Machine Learning - CS229. Stanford University.

[3] NG, Andrew. *Support Vector Machines* [online]. Standford, 2016 [cit. 2017-01-31]. Dostupné z: https://d3c33hcgiwev3.cloudfront.net/_246c2a4e4c249f94c895f607ea1e6407_Lecture12.pdf?Expires=1485993600&Signature=MGhCSj6vnfSMVWpDERGUz8fc2312duxtjEpe2o4R0vhRA9KQQuPnZOPulQy5I0mCrICpxj3M0efelTXkmFFQsKB8&Key-Pair-Id=APKAJLTNE6QMUY6HBC5A. Přednášky k předmětu Machine Learning. Stanford University.

[4] Support vector machine. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2016 [cit. 2017-01-31]. Dostupné z: https://cs.wikipedia.org/wiki/Support_vector_machines.

[5] ZISSERMAN, Andrew. In: *SVM dual, kernels and regression* [online]. Oxford, 2015 [cit. 2017-01-31]. Dostupné z: http://www.robots.ox.ac.uk/~az/lectures/ml/lect3.pdf.