Jméno:

Konvexní optimalizace

Sada 10 domácích úkolů

Termín odevzdání: 20. prosince 2018 ve 12:21

Všechna svá řešení zdůvodněte.

Problém	Bodů max	Bodů
1	2	
2	2	
3	3	
4	3	
Σ	10	

Problém 1. Uvažme hru kámen-nůžky-(papír), která se od známé hry liší tím, že první hráč může hrát kámen, nůžky i papír, ale druhý hráč může hrát jenom kámen a nůžky. Navíc aby druhý hráč nebyl smutný, tak v případě remízy (kámen-kámen nebo nůžky-nůžky) mu první hráč dá 1 korunu. Pokud první hráč vyhraje, dostane 2 koruny od druhého hráče. Kolik korun by měl první hráč dávat druhému, pokud druhý hráč vyhraje, aby hodnota hry byla 0?

Problém můžete řešit buď teoreticky, nebo numericky na počítači. Ve druhém případě popište hlavní myšlenky použité metody a napište mi sem numerické řešení (program mi nemusíte posílat). Dejte si pozor na to, abyste zdůvodnili všechny použité nerovnosti, které (třeba i skrytě) používáte (třeba když násobíte nerovnost nějakým reálným číslem a, tak potřebujete $a \geq 0$).

Problém 2. Zformulujte konvexní problém, který odpovídá odhadu metodou maximální věrohodnosti pro následující problém:

Chceme spočíst vektor $\mathbf{x}$. Z měření známe $\mathbf{y} = A\mathbf{x} + \mathbf{v}$, kde A je známá matice a $\mathbf{v}$ je vektor šumu, který má nezávislé stejně rozdělené složky v_i, s hustotou pravděpodobnosti $p(z) = 1/(2|a|)$ pro $|z| \leq a$ a $p(z) = 0$ jinak.

Na rozdíl od příkladu z hodiny zde ale předem *neznáme* a, tj. $p_{\mathbf{x},a}(\mathbf{y})$ závisí na $\mathbf{x}$ i na a.

Problém 3. Mějme minci, která má neznámou pravděpodobnost x, že na ní při hodu padá panna. Naše apriorní pravděpodobnostní rozdělení pro x má hustotu $p(x) = C \exp(-4(x - 1/2)^2)$ pro $x \in (0, 1)$ a 0 jinak, kde $C > 0$ je (nezajímavá) normalizační konstanta. Řekněme, že nám z 10 hodů padla 7x panna a 3x orel.

Zformulujte problém nalezení nejlepšího aposteriorního odhadu x jako problém konvexní optimalizace a vyřešte ho na počítači. Vysvětlete, jaký optimalizační problém řešíte a jaké x Vám vyšlo. Svůj program mi nemusíte posílat.

Problém 4. V této úloze si vyzkoušíte základy trénování lineárních klasifikátorů (SVM).

1. Stáhněte si sadu dat z

   `http://archive.ics.uci.edu/ml/machine-learning-databases/wine/wine.data`

   obsahující údaje o chemickém složení tří druhů vína. Řádky jsou jednotlivá měření; první číslo na řádku je číslo druhu vína a dalších 13 čísel jsou různé chemické vlastnosti vína (reálná čísla) $\mathbf{x}_i \in \mathbb{R}^{13}$.

2. Protože jsme probírali jenom rozdělení prostoru na dvě části, budeme trénovat klasifikator, který rozpozná víno číslo 1 nebo 3 od vína číslo 2. Data o vínech číslo 1 a 3 tak sloučíme a přidělíme jim značku $y_i = 1$, víno číslo 2 bude mít $y_i = -1$.

3. Budeme potřebovat trénovací a testovací data. Rozdělte soubor wine.data na wine.train a wine.test, kde wine.test bude obsahovat každý pátý řádek z wine.data a wine.train bude obsahovat zbytek (tj. wine.train má 143 řádků a wine.test má 35 řádků).

4. Pomocí CVXOPT/CVXPY najděte $\mathbf{a} \in \mathbb{R}^{13}, b \in \mathbb{R}$ takové, že minimalizují

   $$1/143 \sum_{i=1}^{143} \max(0, 1 - y_i(\mathbf{a}^T\mathbf{x} - b)) + \epsilon/2\|\mathbf{a}\|_2^2$$

   kde data $y_i, \mathbf{x}_i$ jsou řadky ve wine.train (pozor na přepočet y_i!) a parametr ϵ budeme volit 0,1, 1, 2 a 5.

   Všechna čtyři výsledná $\mathbf{a}, b$ a svůj program, který počítá $\mathbf{a}, \mathbf{b}$ mi pošlete na kazda@karlin.mff.cuni.cz.

5. Výsledné klasifikátory nyní vyzkoušejte na datech ze souboru wine.test: Pro dané $\mathbf{a}, b$ a víno se složením $\mathbf{x}_i$ je předpověď vašeho klasifikátoru $y_i = 1$ pokud $\mathbf{a}^T\mathbf{x} \geq b$ a $y_i = -1$ pokud $\mathbf{a}^T\mathbf{x} < b$. Spočtěte a nahlašte mi kolikrát se čtyři klasifikátory z předchozí části zmýlily na testovacích datech. Který klasifikátor měl největší úspěšnost?

Při rešení úloh je možné se poradit s dalšími lidmi (nejlépe dalšími studenty a studentkami Konvexní optimalizace), ale svá řešení (včetně programů!) *pište samostatně* a před termínem odevzdání úloh sepsaná řešení (a programy) nikomu *neukazujte*.