Өнімділікті оңтайландыру. Ньютон әдісі және конъюгаттық градиенттер. Көрнекі мысалдар келтіру

Дәріс 8

- Біз өнімділікті оңтайландыруды талқылауды өткен дәрісте бастадық. Онда біз өнімділік бетін талдау құралы ретінде Тейлор сериясын кеңейтуді енгіздік, содан кейін оны оңтайлы нүктелермен қанағаттандырылуы тиіс шарттарды анықтау үшін пайдаландық. Бұл дәрісте біз тағы да Тейлор сериясын кеңейтуді қолданамыз, бұл жағдайда оңтайлы нүктелерді табу алгоритмдерін әзірлеу үшін. Біз оңтайландыру алгоритмінің үш түрлі категориясын талқылаймыз: ең тік түсу, Ньютон әдісі және конъюгаттық градиент. Келесі дәрістерде біз осы алгоритмдердің барлығын нейрондық жұмыстарды оқытуға қолданамыз.

- Осы дәрістің мақсаты өнімділік индексін оңтайландыру үшін алгоритмдерді әзірлеу болып табылады. Біздің мақсаттарымыз үшін «оңтайландыру» сөзі азайтудың мәнін табуды білдіреді. Біз талқылайтын барлық оңтайландыру алгоритмдері итеративті, бастапқы болжамнан бастаймыз, содан кейін келесі теңдеуге сәйкес кезең-кезеңімен болжамымызды жаңартамыз

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k ,$$

$$\Delta \mathbf{x}_k = (\mathbf{x}_{k+1} - \mathbf{x}_k) = \alpha_k \mathbf{p}_k ,$$

мұндағы pk вектор іздеу бағытын көрсетеді, ал оң скаляр α қадамның ұзындығын анықтайтын оқу жылдамдығы болып табылады. Осы дәрісте қарастыратын алгоритмдер іздеу бағытын таңдаумен ерекшеленеді, . Біз үш түрлі мүмкіндікті талқылаймыз. Сондай-ақ оқу жылдамдығын таңдаудың әртүрлі жолдары бар, және біз олардың бірнешеуін талқылаймыз.

Ең тік түсу

- Теңдеу арқылы оңтайлы (минималды) нүкте туралы болжамды жаңартқанда, біз әрбір итерацияда функцияның төмендеуін минимальді болғанын қалаймыз. Басқа сөздермен айтқанда

$$F(\mathbf{x}_{k+1}) < F(\mathbf{x}_k) .$$

$$F(\mathbf{x}_{k+1}) = F(\mathbf{x}_k + \Delta\mathbf{x}_k) \approx F(\mathbf{x}_k) + \mathbf{g}_k^T \Delta\mathbf{x}_k ,$$

$$\mathbf{g}_k \equiv \nabla F(\mathbf{x})\big|_{\mathbf{x} = \mathbf{x}_k} .$$

$$\mathbf{g}_k^T \Delta\mathbf{x}_k = \alpha_k \mathbf{g}_k^T \mathbf{p}_k < 0 . \qquad \mathbf{g}_k^T \mathbf{p}_k < 0 .$$

- Осы теңдеуді қанағаттандыратын кез келген вектор **төмендеу бағыты** деп аталады. Егер біз осы бағытта аз ғана қадам жасасақ, функция төмендеу керек. Ең тік түсу бағыты қандай? (Функция қай бағытта ең жылдам төмендейді?) Бұл қашан болады?

$$\mathbf{g}_k^T \mathbf{p}_k < 0 \,.$$

Бұл градиент пен бағыт векторы арасындағы ішкі көбейту. Бағыт векторы градиент теріс болған кезде ол теріс болады. Сондықтан ең тік түсу бағытын көрсететін вектор болып табылады.

$$\mathbf{p}_k = -\mathbf{g}_k \,.$$

Мұны теңдеудің итерациясында қолдану. (1) ең тік түсу әдісін шығарады:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \mathbf{g}_k \,.$$

Оқу жылдамдығы

- Ең тік түсу үшін оқу жылдамдығын анықтаудың екі жалпы әдісі бар, α. Тәсілдердің бірі - әрбір итерацияға қатысты өнімділік индексін азайту. Бұл жағдайда біз сызық бойымен азайтамыз

$$\mathbf{x}_k - \alpha_k \mathbf{g}_k \, .$$

Таңдаудың басқа әдісі тұрақты мәнді (мысалы, α =0,02) пайдалану немесе айнымалы, бірақ алдын ала анықталған мәндерді (мысалы, α=1/k) пайдалану болып табылады. Біз келесі мысалдарда таңдауды толығырақ қарастырамыз.

Ең тік түсу алгоритмін қолдану

$$F(\mathbf{x}) = x_1^2 + 25x_2^2, \qquad \mathbf{x}_0 = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}.$$

Бірінші қадам градиентті табу болып табылады:

$$\nabla F(\mathbf{x}) = \begin{bmatrix} \frac{\partial}{\partial x_1} F(\mathbf{x}) \\ \frac{\partial}{\partial x_2} F(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} 2x_1 \\ 50x_2 \end{bmatrix}. \qquad \mathbf{g}_0 = \nabla F(\mathbf{x})\big|_{\mathbf{x} = \mathbf{x}_0} = \begin{bmatrix} 1 \\ 25 \end{bmatrix}.$$

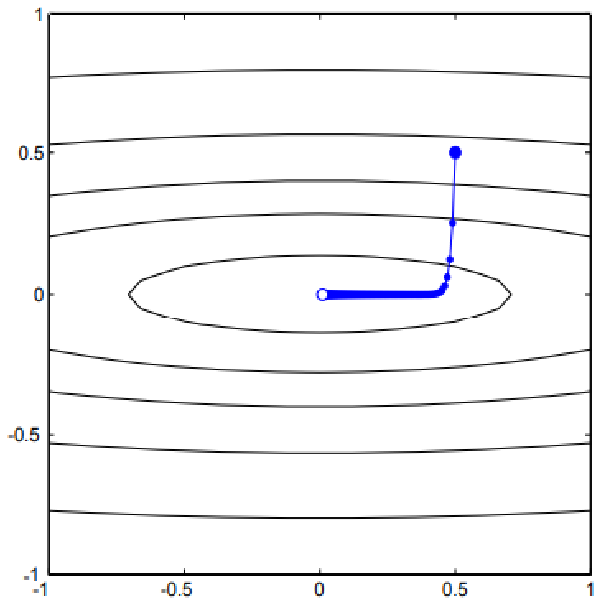
Біз тұрақты оқыту жылдамдығын қолданамыз делік α=0,01. Ең тік түсу алгоритмінің бірінші итерациясы болады

$$\mathbf{x}_1 = \mathbf{x}_0 - \alpha \mathbf{g}_0 = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} - 0.01 \begin{bmatrix} 1 \\ 25 \end{bmatrix} = \begin{bmatrix} 0.49 \\ 0.25 \end{bmatrix}.$$

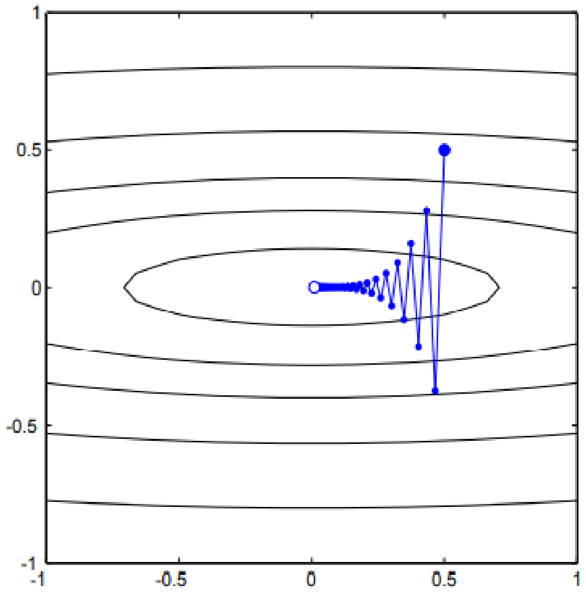
Ең тік түсудің екінші итерациясы

$$\mathbf{x}_2 = \mathbf{x}_1 - \alpha \mathbf{g}_1 = \begin{bmatrix} 0.49 \\ 0.25 \end{bmatrix} - 0.01 \begin{bmatrix} 0.98 \\ 12.5 \end{bmatrix} = \begin{bmatrix} 0.4802 \\ 0.125 \end{bmatrix}.$$

Егер біз итерацияларды жалғастырсақ, біз траекторияны аламыз



Кішігірім оқу жылдамдығы үшін ең тік түсу траекториясы контур сызықтарына әрқашан ортогональ болатын жолмен жүретінін ескеріңіз. Бұл градиент контур сызықтарына ортогональ болғандықтан. Оқу жылдамдығының өзгеруі алгоритм өнімділігін қалай өзгертеді? Оқу жылдамдығын – α=0,035 дейін арттырсақ, 2-суретте көрсетілген траекторияны аламыз. Енді траектория тербелетінін ескеріңіз. Егер оқу жылдамдығын тым үлкен етсек, алгоритм тұрақсыз болады; тербелістер ыдыраудың орнына артады.



- Біз оқу жылдамдығын жоғарылатқымыз келеді, содан бері біз үлкен қадамдар жасаймыз және тезірек жақындауын күтеміз. Алайда, осы мысалдан көріп отырғанымыздай, егер оқу жылдамдығын тым үлкен етсек, алгоритм тұрақсыз болады. Максималды рұқсат етілген оқу жылдамдығын болжаудың қандай да бір жолы бар ма? Бұл ерікті функциялар үшін мүмкін емес, бірақ квадраттық функциялар үшін жоғарғы шекті орнатуға болады.

Тұрақты оқу көрсеткіштері

- Өнімділік индексі квадраттық функция болсын делік:

$$F(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{A}\mathbf{x} + \mathbf{d}^T\mathbf{x} + c . \qquad \nabla F(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{d} .$$

Енді біз бұл өрнекті ең тік төмендеу алгоритмі үшін өрнекке апарып қойсақ (тұрақты оқу жылдамдығын ескере отырып), біз аламыз

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha\mathbf{g}_k = \mathbf{x}_k - \alpha(\mathbf{A}\mathbf{x}_k + \mathbf{d})$$

$$\mathbf{x}_{k+1} = [\mathbf{I} - \alpha\mathbf{A}]\mathbf{x}_k - \alpha\mathbf{d} .$$

Бұл сызықтық динамикалық жүйе, егер матрицаның меншікті мәндері бір шамадан аз тұрақты болса. Бұл матрицаның меншікті мәндерін Гессиан матрицасының меншікті мәндері арқылы өрнектей аламыз. Гессиан матрицасының λ меншікті мәндері мен z меншікті векторлары болсын және болсын.

$$[\mathbf{I}-\alpha\mathbf{A}]\mathbf{z}_i = \mathbf{z}_i-\alpha\mathbf{A}\mathbf{z}_i = \mathbf{z}_i-\alpha\lambda_i\mathbf{z}_i = (1-\alpha\lambda_i)\mathbf{z}_i .$$

Біздің ең тік түсу алгоритмінің тұрақтылығының шарты $|(1-\alpha\lambda_i)| < 1 .$

Егер квадраттық функцияның қатаң минимум нүктесі бар деп алсақ, онда оның меншікті мәндері оң сандар болуы керек. теңдеу. Жоғарыдағы теңсіздік келесідей болады:

$$\alpha < \frac{2}{\lambda_i} .$$

Өйткені бұл бізде бар Гессиан матрицасының барлық меншікті мәндері үшін дұрыс болуы керек

$$\alpha < \frac{2}{\lambda_{max}} .$$

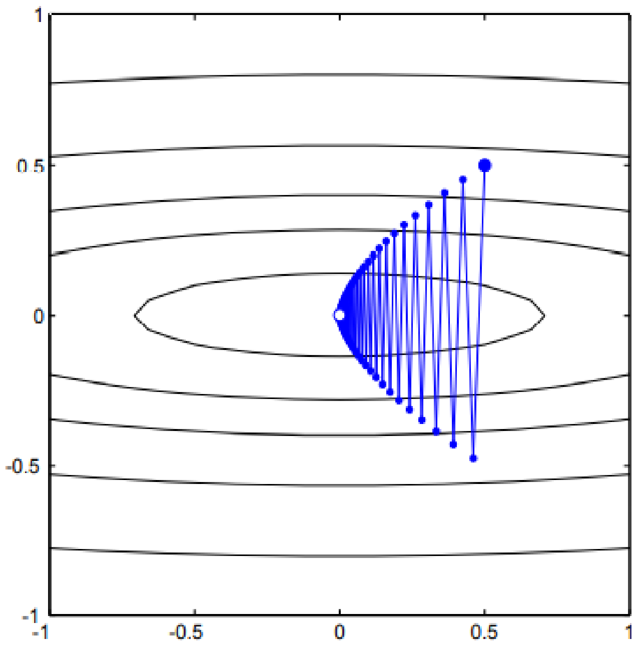
Оқытудың максималды тұрақты жылдамдығы квадраттық функцияның максималды қисықтығына кері пропорционал. Қисықтық градиенттің қаншалықты жылдам өзгеретінін көрсетеді. Егер градиент тым жылдам өзгерсе, біз ең төменгі нүктеден өтуіміз мүмкін, сондықтан жаңа орындағы градиент ескі орындағы градиенттен үлкенірек (бірақ қарама-қарсы бағытта) болады. Бұл қадамдардың әрбір итерацияда өлшемін үлкейтуіне әкеледі.

мысалы,

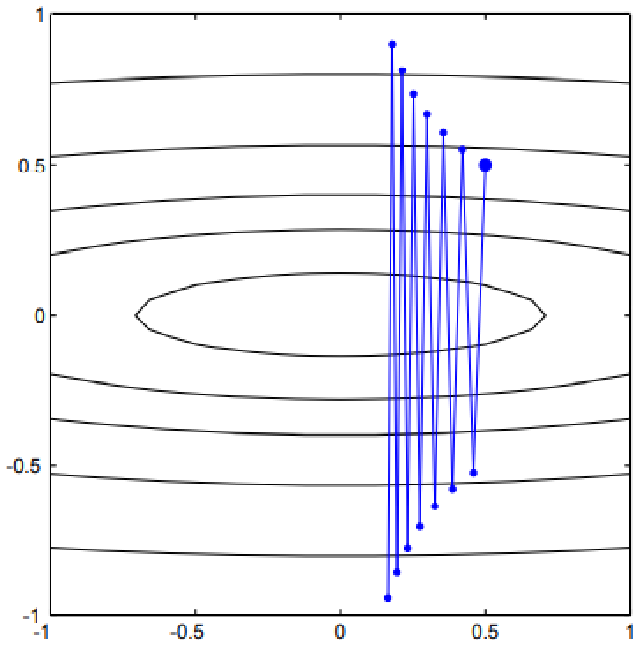
$$\mathbf{A} = \begin{bmatrix} 2 & 0 \\ 0 & 50 \end{bmatrix}.$$

$$\left\{(\lambda_1 = 2), \left(\mathbf{z}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}\right)\right\}, \left\{(\lambda_2 = 50), \left(\mathbf{z}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}\right)\right\}.$$

$$\alpha < \frac{2}{\lambda_{max}} = \frac{2}{50} = 0.04.$$



$\alpha = 0.039$



$\alpha = 0.041$

Сызық бойымен кішірейту

- Оқыту жылдамдығын таңдаудың тағы бір тәсілі әр итерацияға қатысты өнімділік индексін азайту болып табылады. Басқаша айтқанда, азайтуды таңдаңыз

$$F(\mathbf{x}_k + \alpha_k \mathbf{p}_k)\,.$$

$$\frac{d}{d\alpha_k} F(\mathbf{x}_k + \alpha_k \mathbf{p}_k) = \nabla F(\mathbf{x})^T \Big|_{\mathbf{x} = \mathbf{x}_k} \mathbf{p}_k + \alpha_k \mathbf{p}_k^T \nabla^2 F(\mathbf{x}) \Big|_{\mathbf{x} = \mathbf{x}_k} \mathbf{p}_k\,.$$

$$\alpha_k = -\frac{\nabla F(\mathbf{x})^T \Big|_{\mathbf{x} = \mathbf{x}_k} \mathbf{p}_k}{\mathbf{p}_k^T \nabla^2 F(\mathbf{x}) \Big|_{\mathbf{x} = \mathbf{x}_k} \mathbf{p}_k} = -\frac{\mathbf{g}_k^T \mathbf{p}_k}{\mathbf{p}_k^T \mathbf{A}_k \mathbf{p}_k}\,, \qquad \mathbf{A}_k \equiv \nabla^2 F(\mathbf{x}) \Big|_{\mathbf{x} = \mathbf{x}_k}\,.$$

$$F(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \mathbf{x}, \qquad \mathbf{x}_0 = \begin{bmatrix} 0.8 \\ -0.25 \end{bmatrix}.$$

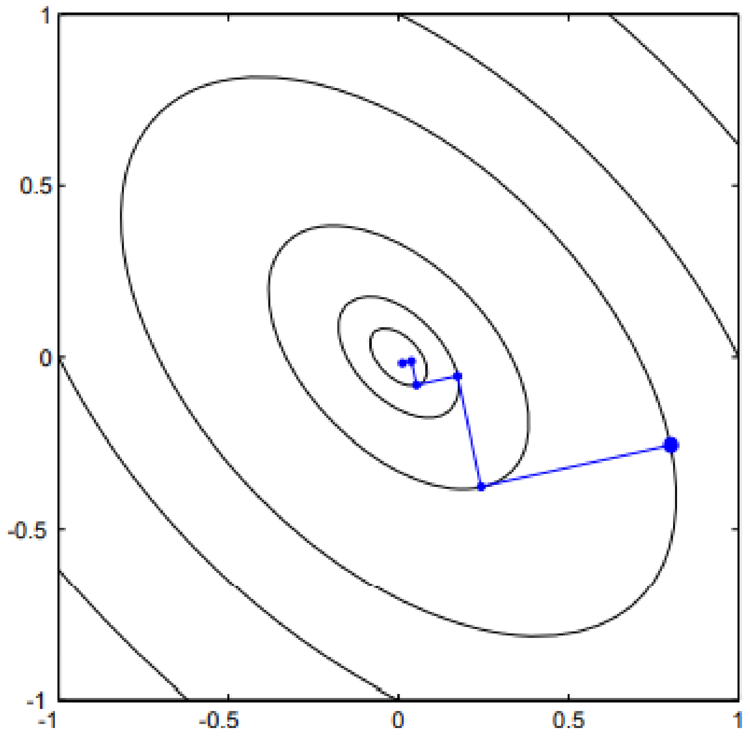
$$\nabla F(\mathbf{x}) = \begin{bmatrix} 2x_1 + x_2 \\ x_1 + 2x_2 \end{bmatrix}.$$

$$\mathbf{p}_0 = -\mathbf{g}_0 = -\nabla F(\mathbf{x})\Big|_{\mathbf{x} = \mathbf{x}_0} = \begin{bmatrix} -1.35 \\ -0.3 \end{bmatrix}. \qquad \alpha_0 = -\frac{\begin{bmatrix} 1.35 & 0.3 \end{bmatrix}\begin{bmatrix} -1.35 \\ -0.3 \end{bmatrix}}{\begin{bmatrix} -1.35 & -0.3 \end{bmatrix}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}\begin{bmatrix} -1.35 \\ -0.3 \end{bmatrix}} = 0.413.$$

$$\mathbf{x}_1 = \mathbf{x}_0 - \alpha_0 \mathbf{g}_0 = \begin{bmatrix} 0.8 \\ -0.25 \end{bmatrix} - 0.413 \begin{bmatrix} 1.35 \\ 0.3 \end{bmatrix} = \begin{bmatrix} 0.24 \\ -0.37 \end{bmatrix}.$$

$$\frac{d}{d\alpha_k}F(\mathbf{x}_k + \alpha_k\mathbf{p}_k) = \frac{d}{d\alpha_k}F(\mathbf{x}_{k+1}) = \nabla F(\mathbf{x})^T\Big|_{\mathbf{x} = \mathbf{x}_{k+1}}\frac{d}{d\alpha_k}[\mathbf{x}_k + \alpha_k\mathbf{p}_k]$$

$$= \nabla F(\mathbf{x})^T\Big|_{\mathbf{x} = \mathbf{x}_{k+1}}\mathbf{p}_k = \mathbf{g}_{k+1}^T\mathbf{p}_k \,.$$



Сондықтан бұл туынды нөлге тең болатын минималды нүктеде градиент алдыңғы іздеу бағытына ортогональ болады. Келесі іздеу бағыты осы градиенттің теріс бағыты болғандықтан, дәйекті іздеу бағыттары ортогональды болуы керек. (Бұл нәтиже кез келген бағытта кішірейту кезінде ең төменгі нүктедегі градиент іздеу бағытына ортогональ болатынын білдіреді, тіпті біз ең тік түсуді қолданбасақ та. Бұл нәтижені конъюгаттық бағыттар туралы талқылауда қолданамыз.)