8 -Дәріс

ӨНІМДІЛІКТІ ОҢТАЙЛАНДЫРУ. НЬЮТОН ӘДІСІ ЖӘНЕ КОНЪЮГАТТЫҚ ГРАДИЕНТТЕР. КӨРНЕКІ МЫСАЛДАР КЕЛТІРУ

Дәрістің мақсаты

Студенттерде нейрондық желілердің өнімділік (қателік) беттерінің минимумын іздеуге арналған негізгі оңтайландыру алгоритмдері — ең жылдам түсу (градиенттік түсу), Ньютон әдісі және конъюгаттық (түйіндес) градиенттер әдісінің математикалық негіздері, жинақталу жылдамдығы және оларды нақты квадраттық функциялар мен желілерге қолдану дағдыларын қалыптастыру.

Оқу нәтижелері

Дәрісті меңгеру нәтижесінде студенттер:

- **Түсінеді:** Бірінші ретті (ең жылдам түсу) және екінші ретті (Ньютон әдісі) оңтайландыру алгоритмдерінің жұмыс істеу принциптерін, олардың артықшылықтары мен есептеу шектеулерін.
- **Салыстыра алады:** Алгоритмдердің жинақталу траекторияларын (зигзаг тәрізді қозғалыс, бір қадамдық жинақталу, \(n\)-қадамдық жинақталу) және үлкен өлшемді нейрондық желілер үшін әдістердің тиімділігін бағалай алады.

- 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.

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.

Қорытынды

Бұл дәрісте нейрондық желілердің өнімділік беттерін оңтайландырудың үш негізгі тәсілі салыстырмалы түрде талданды. Ең жылдам түсу әдісі есептеу жағынан қарапайым болғанымен, созылған қателік беттерінде баяу жинақталатыны көрсетілді. Ньютон әдісі екінші ретті туындыларды (Гессианды) пайдалану есебінен квадраттық функцияларда бір қадамда шешімге жеткенімен, көп параметрлі желілерде кері матрицаны есептеу қиындығына тап болады.

Конъюгаттық градиенттер әдісі осы екі тәсілдің арасындағы оңтайлы тепе-теңдікті қамтамасыз етеді: ол екінші ретті туындылар матрицасын тікелей есептемей-ақ, өзара түйіндес бағыттар бойынша қозғалу арқылы (n)-өлшемді квадраттық беттің минимумын ең көбі (n) қадамда табуға мүмкіндік береді.

Өзін-өзі бағалау сұрақтары

1. **Ең жылдам түсу әдісі:** Неліктен ең жылдам түсу әдісі Гессиан матрицасының меншікті мәндері қатты ерекшеленетін (созылған эллипс пішінді) өнімділік беттерінде баяу жұмыс істейді?

2. **Ньютон әдісі:** Ньютон әдісінің итерациялық формуласын жазыңыз. Бұл әдіс квадраттық функцияның минимумын неше қадамда табады және оның үлкен нейрондық желілерде қолданылуын шектейтін негізгі факторлар қандай?

3 . Конъюгаттық градиенттер: Конъюгаттық градиенттер әдісі \(n\) айнымалысы бар квадраттық өнімділік функциясының минимумын ең көбі неше итерацияда табады? Флетчер-Ривс формуласы не үшін қолданылады?

4. Алгоритмді таңдау: Егер нейрондық желіде мыңдаған салмақ коэффициенттері болса, Ньютон әдісі мен конъюгаттық градиенттер әдісінің қайсысын таңдаған тиімдірек және неліктен?

Әдебиеттер тізімі

1. **Хейган М. Т., Демут Х. Б., Бил М. Х., Де Хесус О.** Нейронные сети. Дизайн и применение (Neural Network Design). — М.: ДМК Пресс, 2021. — 800 с. *(Екі кірісті перцептрон \(R=2\), «алма мен апельсин» тану мысалы, Хэмминг және Хопфильд желілері бойынша негізгі базалық оқулық).*

2. **Хайкин С.** Нейронные сети: полный курс. — 2-е изд. — М.: Издательский дом «Вильямс», 2018. — 1104 с. *(Бірқабатты перцептрондар және Хопфильдтің рекурренттік ассоциативті жады теориясы).*

3. **Осовский С.** Нейронные сети для обработки информации. — М.: Финансы и статистика, 2017. — 448 с. *(Хэмминг желілері, бәсекелес қабаттар және рекурренттік модельдер).*

4. **Уоссермен Ф.** Нейрокомпьютерная техника: Теория и практика. — М.: Мир, 2016. — 240 с. *(Хопфильд желісінің энергия функциясы мен бейнелерді қалпына келтіру алгоритмдері).*

5. **Бейсембекова Р. Н., Әбдірахманова Ж. Б.** Жасанды интеллект және нейрондық желілер негіздері: Оқу құралы. — Алматы: Қазақ университеті, 2021. — 184 б. *(Қазақ тіліндегі үлгілерді тану және нейрондық желі модельдері).*