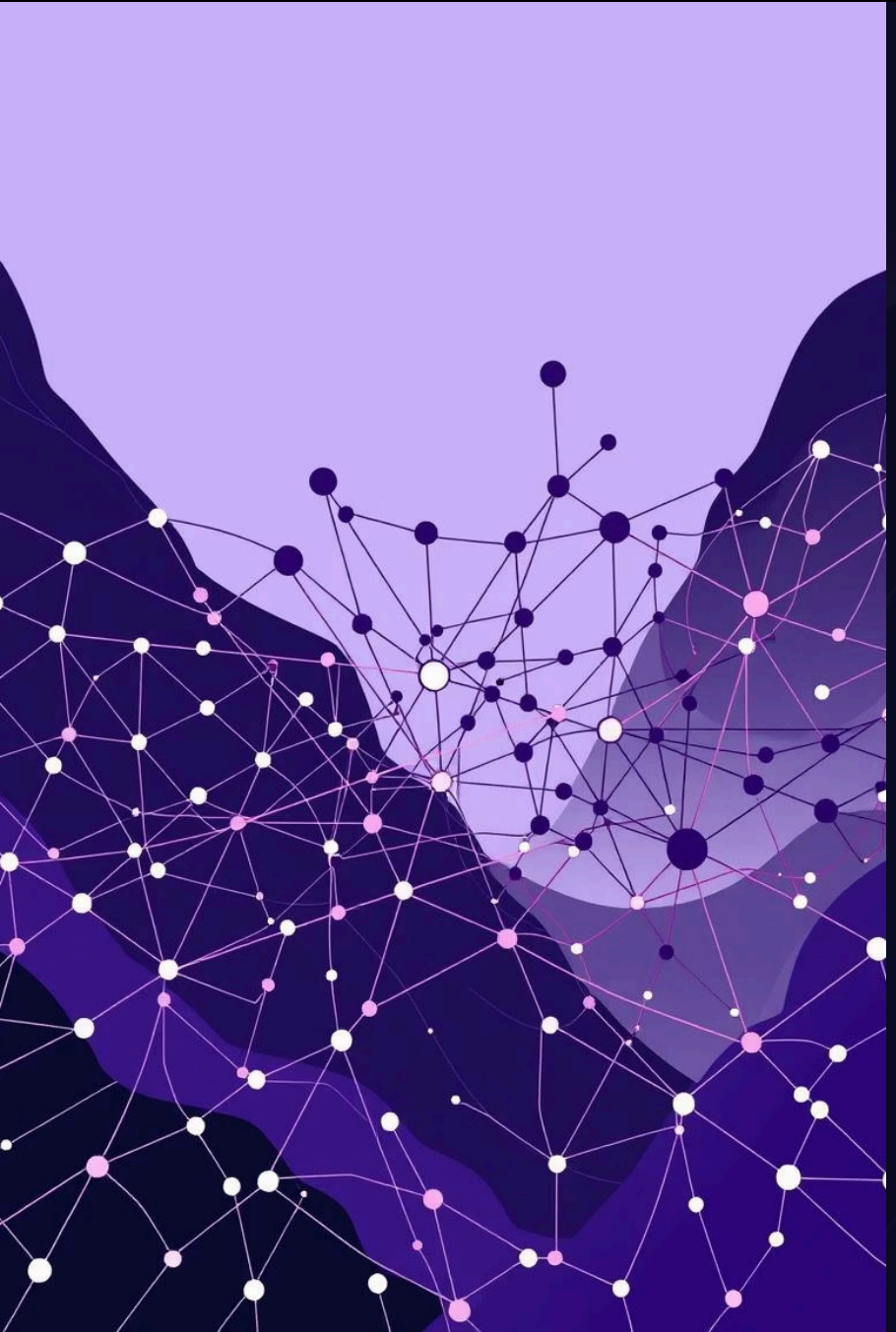


Дәріс 10. Машиналық оқыту алгоритмдері (2 бөлім).

Оқытушы: Сарсембаева Т.С.

Дәрістің мақсаты.

Машиналық оқытудың алдыңғы дәрісте басталған негізгі алгоритмдерін талдауды жалғастыру. Бұл мақсат аясында, біріншіден, бірнеше сызықтық регрессияның қарапайым сызықтық регрессиядан айырмашылығын және оның бірнеше тәуелсіз айнымалылармен қалай жұмыс істейтінін түсіндіру қарастырылады. Екіншіден, логистикалық регрессияның классификация мәселелерін шешу үшін ықтималдықтарды қалай модельдейтінін меңгерту көзделеді. Сонымен қатар, дәріс шешім ағаштарының (Decision Trees) интуитивті, ережелерге негізделген құрылымын және оның шешім қабылдау процесін зерттеуге бағытталған. Соңында, кездейсоқ ормандардың (Random Forest) бірнеше шешім ағаштарының ансамблі ретінде қалай жұмыс істейтінін және оның дәлдікті арттыру әдістерін түсіндіру мақсат етіледі.

Негізгі сұрақтар:

- Бірнеше сызықтық регрессияның қарапайым сызықтық регрессиядан (Simple) айырмашылығы неде?
- Логистикалық регрессия "регрессия" деп аталғанымен, неліктен ол классификация (жіктеу) үшін қолданылады? Сигмоидтық функцияның рөлі қандай?
- Шешім ағашы (Decision Tree) деректерді бөлу үшін ең жақсы "сұрақты" қалай таңдайды?
- Кездейсоқ орман (Random Forest) дегеніміз не және ол бір ғана шешім ағашына қарағанда неліктен жиі жақсырақ нәтиже береді?
- Кездейсоқ орманда "Bootstrap sampling" және "дауыс беру" процестері қалай жұмыс істейді?

Қысқаша мазмұны (тезистер):

Бұл дәріс **KNN,** Аңғал Байес және қарапайым сызықтық регрессиядан кейінгі күрделірек модельдерді қарастырады. Бұл алгоритмдер болжам жасауға және себеп-салдарлық байланыстарды түсінуге көмектеседі.

Бірнеше сызықтық регрессия

Бұл бір тәуелді айнымалы (target) мен бірнеше тәуелсіз айнымалылар (features) арасындағы сызықтық қатынасты модельдейтін әдіс. Қарапайым регрессиядан айырмашылығы, ол бір уақытта бірнеше айнымалының әсерін ескереді. Модельдің мақсаты – әр белгінің салмағын (b) таба отырып, орташа квадраттық қатені (MSE) азайту.

Логистикалық регрессия

Бұл тәуелді айнымалының екі немесе одан да көп кластары арасындағы ықтималдықтарды болжауға арналған жіктеу моделі. Ол сызықтық комбинацияның нәтижесін алып, оны сигмоидтық (логистикалық) функция арқылы 0-ден 1-ге дейінгі ықтималдыққа түрлендіреді. Егер ықтималдық 0.5-тен жоғары болса, объект 1-классқа, төмен болса 0-классқа жатқызылады. Ол тек сызықтық шешімдер қабылдай алады, бірақ түсіндіруге оңай және жылдам.

Шешім ағаштары **(Decision Trees)**

Бұл ағаш құрылымы арқылы шешім қабылдайтын, көптеген ережелер мен шарттарға негізделген модель. Ол деректерді бірқатар шарттар арқылы бөледі. Ағаш Түбірден (Root) басталып, Тармақтар (Branches) арқылы өтіп, Түбегейлі түйіндерде (Leaf Nodes) шешім қабылдайды. Деректерді бөлу үшін ең жақсы "сұрақты" таңдау үшін Ақпараттық ұтыс (Information Gain) және Энтропия сияқты математикалық принциптерді қолданады.

Кездейсоқ орман **(Random Forest)**

Бұл бірнеше шешім ағаштарының ансамблі (бірігуі) болып табылатын әдіс. Ол бірнеше ағаштың нәтижелерін біріктіру арқылы дәлірек және тұрақты болжам жасайды.

Жұмыс істеу механизмі:

- Bootstrap sampling: Әрбір ағаш үшін бастапқы деректерден кездейсоқ іріктеу арқылы жаңа деректер жиынтығы жасалады.
- Кездейсоқ мүмкіндіктерді таңдау: Әрбір түйінде ағаш барлық белгілерді емес, тек олардың кездейсоқ таңдалған бөлігін ғана ескереді.
- Дауыс беру / Орташалау: Классификацияда әр ағаш "дауыс береді" (ең көп дауыс алған класс таңдалады), ал регрессияда барлық ағаштардың болжамдарының орташа мәні алынады.

Кіріспе

Машиналық оқыту (Machine learning) — бұл деректердегі заңдылықтарды автоматты түрде анықтай алатын және болашақ нәтижелерді болжай алатын модельдерді құру процесі.

Алдыңғы дәрістерде біз K-жақын көршілерді (KNN), Аңғал Байесті (Naïve Bayes) және қарапайым сызықтық регрессияны (Simple Linear Regression) талдадық.

Енді келесі деңгейге өтіп, күрделі модельдерді қарастырайық — бірнеше регрессия, логистикалық регрессия, шешім ағаштары және кездейсоқ ормандар. Бұл алгоритмдер Data Science, эконометрика, қаржы, денсаулық сақтау, әлеуметтік талдау сияқты салаларда кеңінен қолданылады.

Олар болжам жасауға ғана емес, сонымен қатар себеп-салдарлық байланыстарды түсінуге және модельді түсіндіруді қамтамасыз етуге көмектеседі (interpretability).

Бірнеше сызықтық регрессия

Бірнеше сызықтық регрессия (Multiple Linear Regression) - бұл бірнеше тәуелсіз айнымалылар (feature) мен бір тәуелді айнымалы (target) арасындағы сызықтық қатынасты модельдейтін әдіс. Бұл модель мәліметтердегі бірнеше айнымалылардың әсерін ескере отырып, тәуелді айнымалыны болжау үшін қолданылады. Классикалық сызықтық регрессия тек бір тәуелсіз айнымалымен жұмыс істейтін болса, бірнеше сызықтық регрессия бір уақытта бірнеше тәуелсіз айнымалыларды ескереді және олардың тәуелді айнымалыға әсерін есептейді.

Математикалық түрде:

$$y^{=}b0 + b1x1 + b2x2 + ... + bnxn$$

y ^ – модель болжаған мән,

$b0$ – еркін мүше(intercept),

bi – әр белгінің (feature) салмағы,

$x2$ – түсіндірмелі айнымалылар.

немесе матрица түрінде:

$$y^{=}X\beta$$

X — n × (p + 1) дизайн-матрица (бірінші баған-бірліктер, содан кейін әрбір баған-айнымалылар), $\beta = (\beta 0, \beta 1, \ldots, \beta p)T$ -коэффициенттердің векторы.

Модельдің мақсаты - нақты мәндер y мен болжамдар y ^ арасындағы айырмашылықты азайту: $L = 1n\sum i = 1n(yi - y^{i})2$

Бұл функция MSE деп аталады, яғни орташа квадраттық қате.

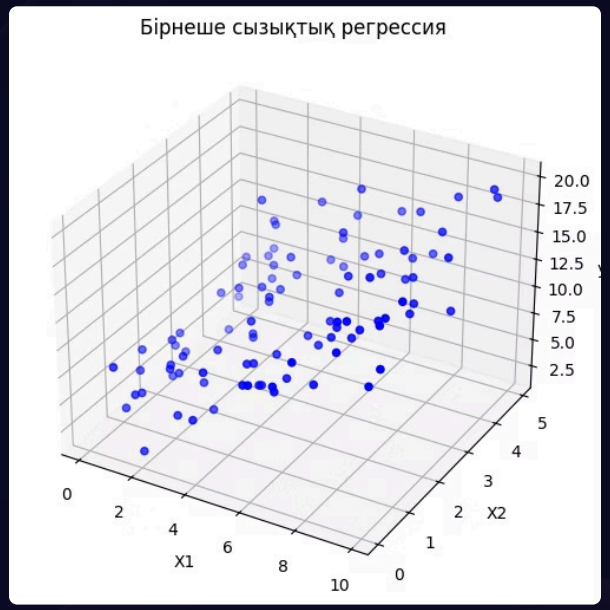
Есептеу әдісі

Регрессия коэффициенттері аналитикалық түрде есептеледі:

$$\beta = (XTX) - 1XTy$$

X – деректер матрицасы, y – мақсаттық вектор.

Модельдің мақсаты тәуелді айнымалыны болжау үшін тәуелсіз айнымалылардың әрқайсысы үшін өз салмағын табу болып табылады. Бұл салмақтар деректерге негізделген (яғни олар максималды ықтималдыққа сәйкес келеді). Модельдің параметрлері (салмақтары) оптимизациялау әдістері арқылы анықталады, мысалы, ең кіші квадраттар әдісі (Ordinary Least Squares, OLS).



Логистикалық регрессия

Логистикалық регрессия - тәуелді айнымалының екі немесе одан да көп кластары арасындағы ықтималдықтарды болжауға арналған статистикалық модель. Бұл модельдің негізгі мақсаты әртүрлі деректер негізінде бір классты немесе санатты болжау болып табылады. Логистикалық регрессия көбінесе жіктеу мәселелерін шешу үшін, яғни белгісіз объектілердің белгілі категорияға жататындығын анықтау үшін қолданылады.

Логистикалық регрессияның жұмыс принципі:

Логистикалық регрессия - бұл сызықтық модель, бірақ оның нәтижесі сызықтық емес болады. Болжам жасамас бұрын, модельде тәуелсіз айнымалылардың сызықтық комбинациясы қолданылады, бірақ соңында ол логистикалық функцияның (sigmoida) ықтималдығына айналады.

Логистикалық функция **(sigmoida):**

Логистикалық регрессияның негізі \[0, 1\] интервалындағы мәндерді ықтималдықтарға түрлендіретін сигмоидтық функция болып табылады.

Сигмоидтық функция келесідей есептеледі:

$$\sigma(z) = 11 + e - z$$

z-сызықтық комбинация (мысалы, $z = \omega 0 + \omega 1x1 + \omega 2x2 + ... + \omega nxn$).

Шешім қабылдау:

Логистикалық регрессия нәтижесінде шығыс мәні 0 немесе 1 болады, яғни ықтималдылықты есептеп, оны белгілі бір шекпен салыстырыңыз (әдетте 0,5). Егер ықтималдық 0,5-тен жоғары болса, Модель объектіні 1-классқа, ал төменгісі 0-классқа жатқызады.

Қолданылуы:

Екілік классификация: логистикалық регрессия тек екі мәнге тәуелді айнымалыларды болжауда қолданылады (мысалы, 0 және 1, Иә және жоқ, ауру және денсаулық).

Модельді оқыту үшін көптеген әртүрлі әдістер қолданылады (мысалы, ықтималдықтың максималды әдісі).

Формуласы:

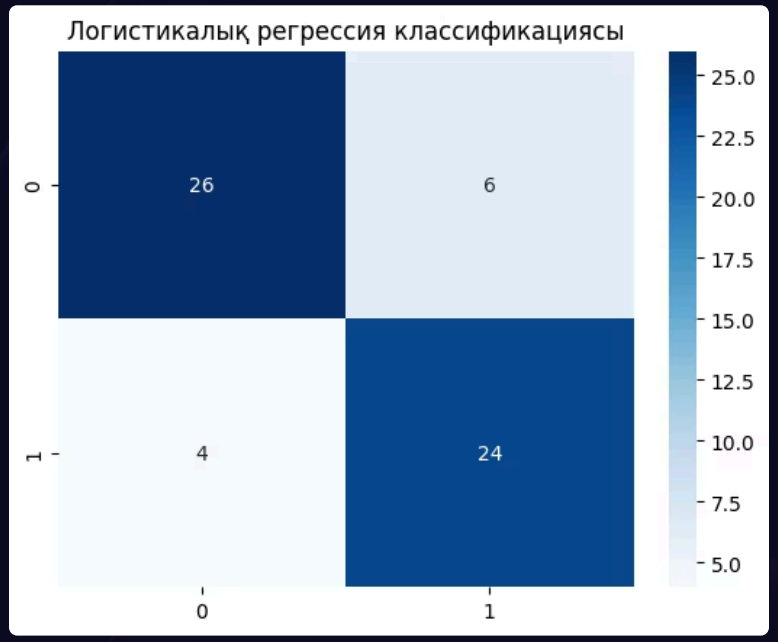
$$P(y = 1 \mid x) = 11 + e - (\beta 0 + \beta 1x1 + ... + \beta nxn)$$

Мұнда логистикалық функция (сигмоид) нәтижені 0 — ден 1 аралығында шектейді, яғни ықтималдық береді.

Мақсаты

Модельдің мақсаты-логарифмдік шығын функциясын(log-loss) минимизациялау:

$$L = -1n \sum i = 1n[yilog(yi^) + (1 - yi)log(1 - yi^)]$$



Шешім ағаштары (Decision Trees)

Шешім ағашы - бұл көптеген ережелер мен шарттарға негізделген болжамдарды жүзеге асыратын деректерді жіктеу немесе регрессиялау үшін қолданылатын модель түрі. Бұл модельді ойлап табу процесінде негізгі идея - деректерді бірқатар шарттар арқылы оңай және түсінікті түрде бөлу. Шешім ағашы түсінікті және интуитивті, өйткені ол ағаш құрылымы арқылы шешім қабылдайды.

Шешім ағашы келесі элементтерден тұрады:

- Түбір (Root): шешім ағашының ең бірінші бөлігі. Бұл нүкте барлық деректердің біріктірілген күйін білдіреді. Әдетте, осы түбірде деректерді шығару үшін ең тиімді атрибут таңдалады.
- Тармақтар (Branches): әрбір ішкі тамырдан шығатын сызықтар. Олар ағаштың бұтақтарын қосымша жағдайлар мен сұрыптаулар арқылы бағыттайды.
- Түбегейлі түйіндер (Leaf Nodes): белгілі бір шешім немесе болжам жасалатын шешім ағашының соңғы түйіндері. Әрбір радикалды түйіннің белгілі бір класы немесе сандық мәні болады.
- Шарттар мен бөліну ережелері: әрбір ішкі түйінде деректер белгісіне негізделген бөлу ережесі бар. Мысалы, көрсетілген атрибуттардың нақты мәніне байланысты деректер екі немесе одан да көп бөлікке бөлінеді.

Математикалық негізінде:

Шешім қабылдау ақпараттық ұтыс (Information Gain) принципіне негізделеді:

$$IG = Entropy(parent) - \sum iNiNEntropy(childi)$$

мұндағы энтропия:

$$Entropy = -\sum k = 1Kpklog2(pk)$$

Шешім ағашын классификация (класстарды бөлу) және регрессия (сандық мәндерді болжау) тапсырмаларында да қолдануға болады. Бұл модельдің басты ерекшелігі - ол бірқатар сұрақтар мен ережелер негізінде түйіндер мен тармақтар арқылы деректерді бөледі, нәтижесінде түпкілікті шешім қабылданады.

Кездейсоқ орман

Кездейсоқ орман (Random Forest) - бұл бірнеше шешім ағаштарының ансамблі болып табылатын статистикалық оқыту әдісі. Оның басты ерекшелігі - бірнеше шешім ағаштарын құру және әр ағаштың нәтижелерін біріктіру арқылы болжам жасау. Бұл модель әртүрлі ағаштардың пікірлерін біріктіріп, ортақ шешім қабылдайды. Әрбір ағаштың өзіндік сипаттамалары болғанымен, оларды біріктірген кезде модельдің болжамы дәлірек және тұрақтылыққа ие болады.

Негізгі идея - көптеген ағаштардың әртүрлілігін пайдалану, осылайша тек бір ағаштың күшті және әлсіз жақтарын теңестіру.

1. Кездейсоқ деректерді іріктеу(Bootstrap sampling):

Әрбір шешім ағашы үшін кездейсоқ деректер бөлігі таңдалады. Бұл әдіс Bootstrap деп аталады, онда әрбір ағаш үшін кездейсоқ жасалған деректер үлгісі пайдаланылады. Бұл таңдау әрбір жеке шешім ағашы үшін әртүрлі деректер опцияларын алуға мүмкіндік береді.

2. Кездейсоқ мүмкіндіктерді таңдау:

Әрбір шешім ағашында бөлуді құру кезінде функциялардың кездейсоқ жиынтығы ғана қарастырылады. Яғни, әрбір түйінде барлық белгілердің орнына тек бірнешеуін таңдап, бөлу керек. Бұл тәсіл модельдің әртүрлілігін арттырып, ағаштар арасындағы корреляцияны азайтуға көмектеседі.

3. Дауыс беру (классификация) немесе ағаштардың орташа есебі (регрессиясы):

- Классификация: егер модель жіктеу тапсырмасын орындаса, әр ағаш өз болжамын жасайды және ең көп дауыс жинаған сынып таңдалады (шешім көпшілік дауыспен қабылданады).
- Регрессия: егер модель регрессия тапсырмасын орындаса, барлық ағаштардың болжамды мәндерінің орташа мәнін алу керек.

Формуласына келетін болсақ:

$$y\hat{} = mode(f1(x), f2(x), \ldots, fm(x))$$

Бұл формула моданы (mode) қолдануды көрсетеді, яғни жиынның ең көп тараған мәнін алу үшін пайдалануды көрсетеді:

y ^ - болжамды немесе алынған нәтиже (болжам), $(f1(x), f2(x), \ldots, fm(x))$ - m түрлі модельдердің немесе функциялардың болжамды мәндері.

Дәрісті меңгеруге арналған бақылау сұрақтары:

- Бірнеше сызықтық регрессияның математикалық формуласын қарапайым сызықтық регрессиямен салыстырыңыз. Мұнда b1, b2 нені білдіреді?
- Логистикалық регрессия қалайша 0 мен 1 арасындағы ықтималдықты алады? Бұл процесте сигмоидтық функцияның рөлі қандай?
- Шешім ағашының құрылымын сипаттаңыз (Түбір, Тармақ, Жапырақ).
- Кездейсоқ орман (Random Forest) бірнеше ағаштың пікірін қалай біріктіреді? (Классификация және регрессия үшін бөлек түсіндіріңіз).
- Неліктен Кездейсоқ орман бір ғана шешім ағашына қарағанда Overfitting-ке (артық үйренуге) азырақ бейім?

Әдебиеттер тізімі:

1. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press. **https://www.deeplearningbook.org**
2. Hastie, T., Tibshirani, R., & Friedman, J. (2009). The Elements of Statistical Learning. Springer.**https://hastie.su.domains/ElemStatLearn/**
3. Sarker, I. H. (2021). Data Science and Analytics: An Overview from Data-Driven Smart Computing and Decision Making. SN Computer Science, Springer.
4. Pedregosa, F. et al. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825–2830. https://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html