Пәні: Деректерді интеллектуалды талдау және визуализациялау

Тірек векторлар әдісі. "Жақын көрші" әдісі

Дәріс №9

Шаяхметова А.С., PhD, қауымдастырылған профессор
Жасанды интеллект және Big Data кафедрасы
2026-2027 оқу жылы

Мақсаты

Тірек векторлар әдісінің (SVM) мәнін және сызықты SVM принципін түсіндіру. "Жақын көрші" және k-жақын көршілер әдістерімен таныстыру. Әдістердің артықшылықтары мен кемшіліктерін салыстыру.

Жоспары

1. Тірек векторлар әдісі (SVM): мәні мен принципі
Шекаралық әдістер, шешім жазықтығы, тірек векторлар

2. Сызықты SVM және ядро операторы
Гиперплан, тәуекел, сызықты емес жағдай

3."Жақын көрші" әдісі және прецеденттер
Nearest neighbour, CBR, прецедент кезеңдері

4. k-жақын көршілер: жіктеу және болжау
Көпшілік дауыс, орташа мән, k параметрі

1. Тірек векторлар әдісі (SVM): мәні мен принципі

Шекаралық әдістер тобына жатады

Бинарлы классификация есептерін шешеді

Шешім жазықтығына негізделген

SVM анықтамасы

Тобы
Шекаралық әдістер тобына жатады

Принципі
Облыстардың шекараларын қолданып кластарды анықтайды

Есебі
Бинарлы классификация тапсырмаларын шешуге арналған

Негізі
Шешім жазықтығы (plane) туралы түсінік жатыр

SVM = Support Vector Machine – тірек векторлар әдісі

Шешім жазықтығы: схема

Сары класс (овал)

Қоңыр класс (тікбұрыш)

Бөлу сызығы

Әр объект екі өлшеммен сипатталады

Тірек векторлар

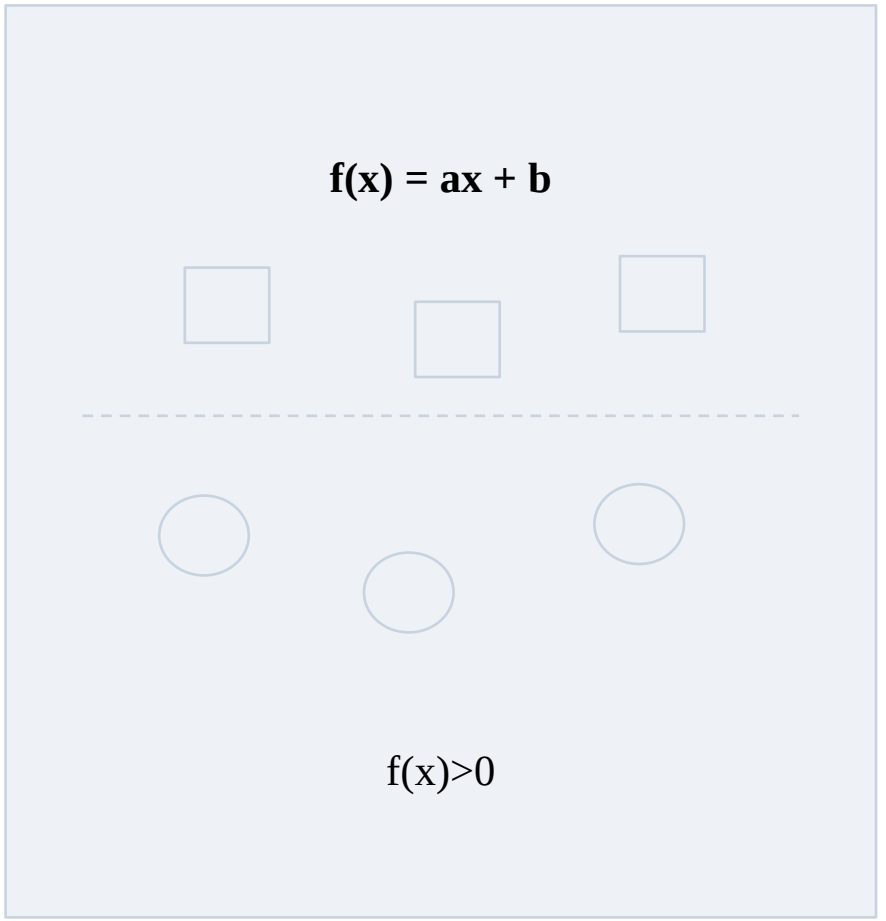
Ұғым	Мағынасы
Мақсат	Екі объектіні бөлетін жазықтықты табу
Тірек векторлар	Кластар шекарасында орналасқан үлгілер
Жақсы жіктеу	Шекаралар арасындағы аймақ бос болады
Әдіс екі класс арасындағы шекарада орналасқан үлгілерді, яғни тірек векторларын іздейді	

2. Сызықты SVM және ядро операторы

Гиперплан: f(x)=ax+b

Күтілетін тәуекел

Сызықты емес жағдай



Сызықты SVM

Мақсат - деректерді екі класқа бөлетін сызықтық функцияны табу

Бір класс үшін f(x)<0, екінші класс үшін f(x)>0

Гиперплан «a» векторымен және «b» мәнімен анықталады: f(x)=ax+b

Жаңа объектінің таңбасы (оң немесе теріс) оның класын көрсетеді

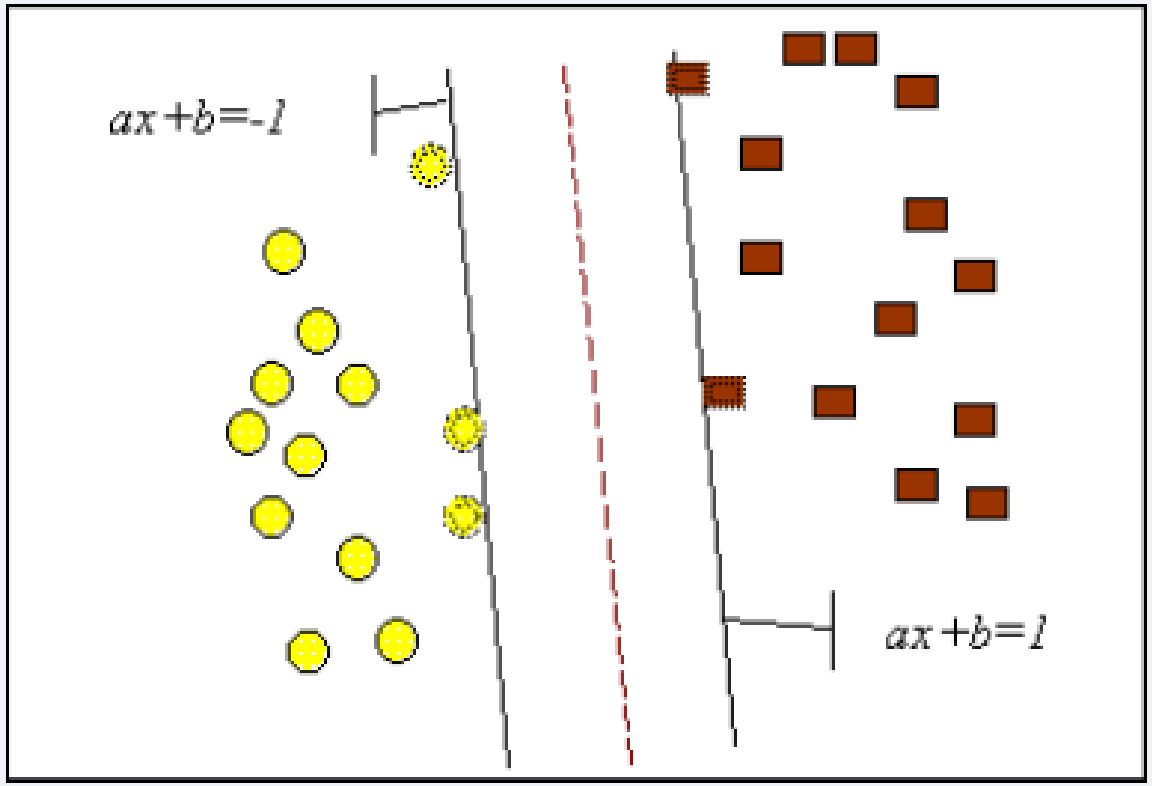
Гиперплан

$$f(x) = ax + b$$

$f(x) > 0 \rightarrow$ бірінші класс

$f(x) < 0 \rightarrow$ екінші класс

Сызықты SVM



ax+b = 1
бір класстың шекарасы

ax+b = 0
бөлуші гиперплан

ax+b = −1
екінші класстың шекарасы

Шекаралар арасындағы аймақ бос

Жіктеу функциясын құру: схема

Жаттығу жиыны

белгілі кластары бар үлгілер

→

f(x)=ax+b

жіктеу функциясы

→

Эмпирикалық тәуекел

жаттығу жиынындағы қате

→

Күтілетін тәуекел минимум

жаңа деректердегі қате ең аз

Эмпирикалық тәуекелді азайту әрқашан күтілетін тәуекелді азайтуға әкелмейді, әсіресе жаттығу деректері аз болғанда

Сызықты емес жағдай: ядро операторы

2D: сызықтық шекара жоқ

→

Ядро операторы: өлшемді арттыру

$\varphi(x)$

→

3D: гиперплан кластарды бөледі

Өлшем жоғары болса, модельмен жұмыс қиындайды: алдымен ең маңызды компоненттерді анықтап, өлшемді азайтады, содан кейін SVM қолданады

SVM артықшылықтары мен кемшіліктері

Артықшылығы	Кемшілігі
Жіктеу үшін аз мәліметтер жиынтығы жеткілікті	Көп үлгінің тек шекарадағы бөлігі ғана қолданылады
Күтілетін тәуекелдің жоғарғы бағасы аз	Жоғары өлшемді деректермен жұмыс қиын
Сызықты емес деректермен жұмыс істей алады	Ядро операторын таңдау қажет

3. "Жақын көрші" әдісі және прецеденттер

Nearest neighbour: деректер жадта сақталады

Case Based Reasoning (CBR) - прецеденттер бойынша пайымдау

Прецедент - жағдайдың сипаттамасы және әрекеттер

"Жақын көрші" әдісі

Жаңа элементпен салыстыру үшін деректерді жадта сақтауға негізделген

Жаңа жазба сақталған деректермен салыстырылады

Ең ұқсас (жақын көрші) жазба сәйкестендіріледі

"Мұғалімсіз оқыту": мысалдар жиналған сайын сипаттамалар жақсарады

Барлық деректерді сақтау ыңғайлы емес, сондықтан кейде тек "типтік" жағдайлар сақталады

Мысал: банктің жаңа клиенті

Жаңа клиент

кіріс, жас және басқа сипаттамалар белгілі

→

Атрибуттарды салыстыру (кіріс, жас)

базадағы клиенттермен салыстырылады

→

k жақын көрші таңдау

ең ұқсас k клиент алынады

→

Шешім қабылдау

көршілердің шешімі алынады

"k-жақын көрші": k ең жақын көрші "жақын көршілер" жиыны ретінде қарастырылады

Прецедентке негізделген тәсіл кезеңдері

Міндет туралы ақпарат жинау

Базадағы прецеденттермен салыстыру

Ең жақын прецедентті таңдау

Шешімді мәселеге бейімдеу

Шешімнің дұрыстығын тексеру

Жаңа прецедентті базаға енгізу

"Жақын көрші" әдісі: артықшылығы мен кемшілігі

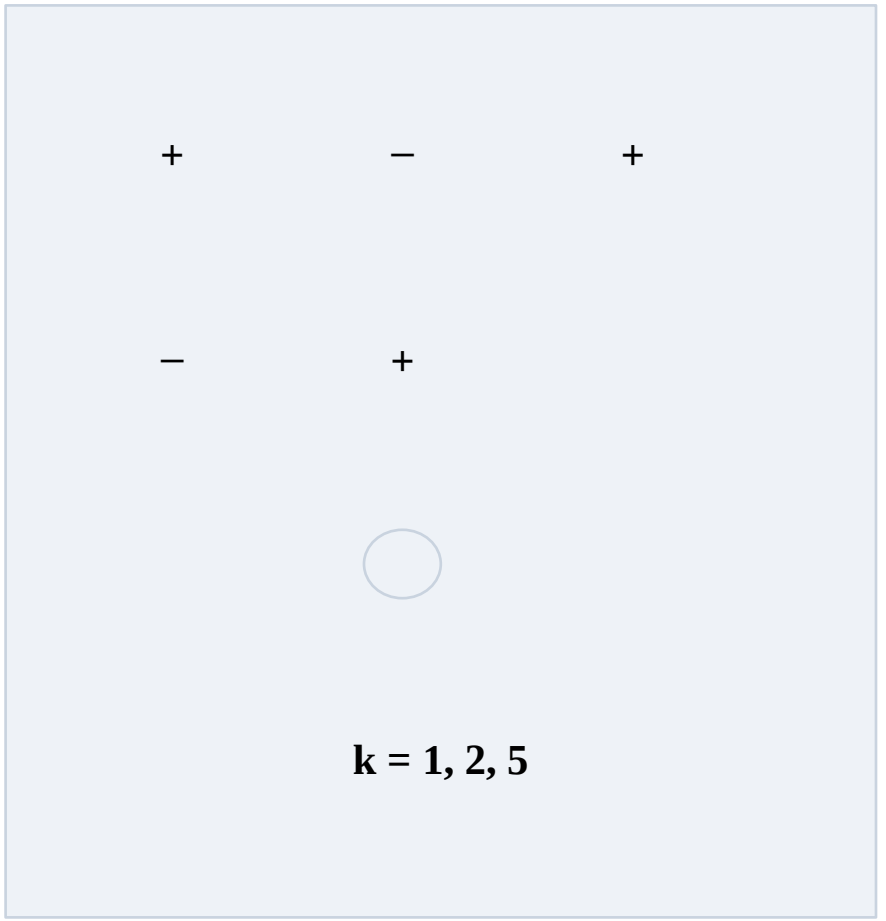
Артықшылықтары	Кемшіліктері
Нәтижені қолдану қарапайым	Жалпылайтын модель мен ереже жасалмайды
Шешімді басқа жағдайларда қолдануға болады	Метриканы таңдау қиын
Мақсат – мүмкін болатын ең жақсы шешім	Есептеу күрделілігі жоғары

4. k-жақын көршілер: жіктеу және болжау

Жіктеу: "көпшілік дауыс бойынша"

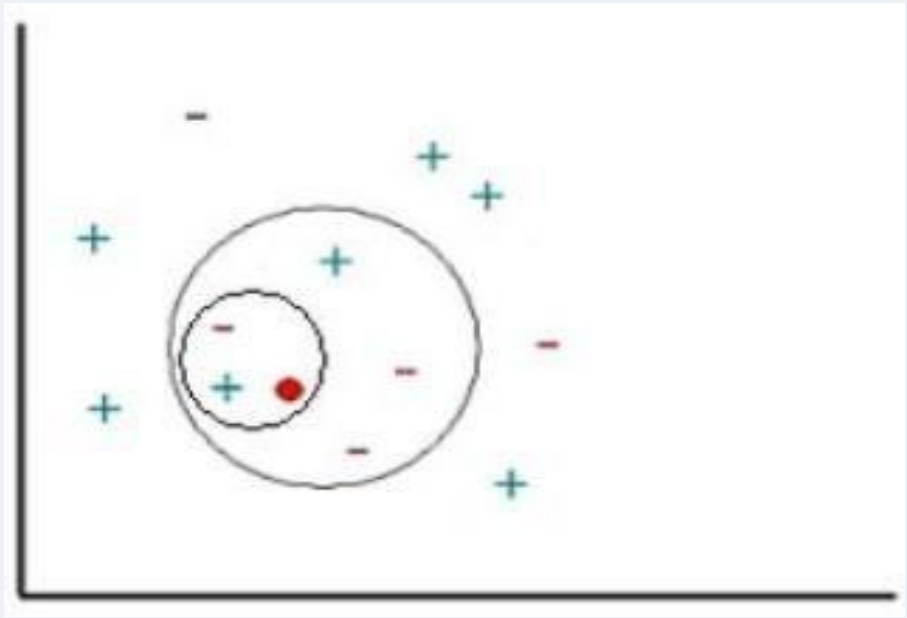
Болжау: k көршінің орташа мәні

k параметрін таңдау



Жіктеу есебі

k параметрінің мәні әртүрлі жиын объектілерін жіктеу



Мысалдар «+» немесе «-» таңбасымен белгіленген

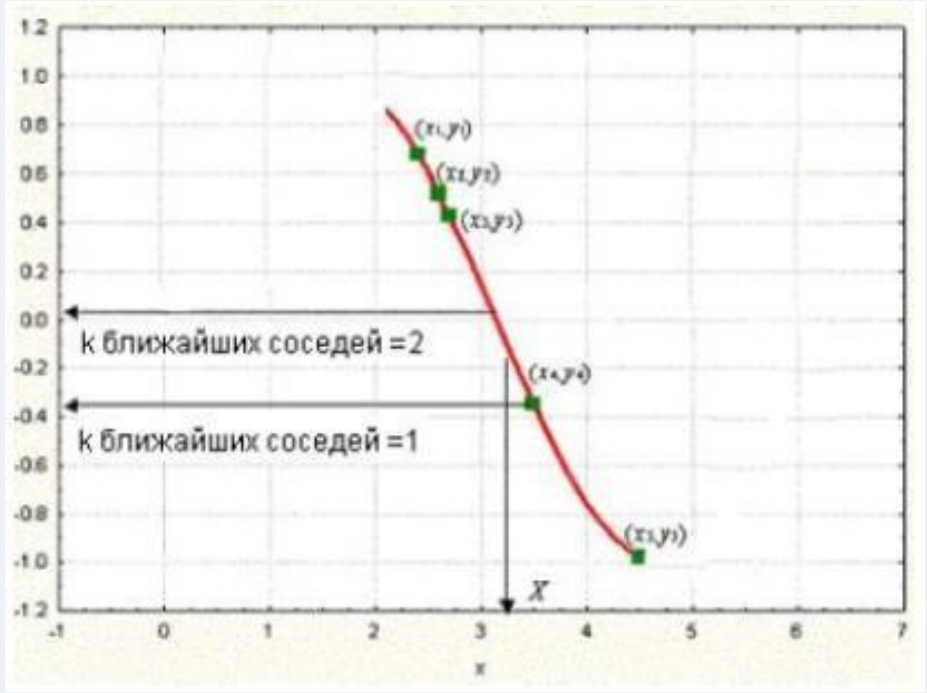
Қызыл нүкте - сұрау нүктесі

Жіктеу "көпшілік дауыс бойынша" жүзеге асады

Жіктеу нәтижесі k мәніне байланысты

k	Көршілер	Нәтиже
1	Жақын нүкте "+"	"+" класы
2	"+" және «-" тең	Жіктей алмайды
5	2 нүкте "+", 3 нүкте «-»	«-» класы
Сұрау нүктесі - жіктелуі қажет жаңа объект (қызыл шеңбермен белгіленген)		

Болжау (регрессия) есебі



k=1: Y = y4

k=2: Y = (y3+y4)/2

Жалпы: k көршінің орташа арифметикалық мәні

k параметрінің әртүрлі мәндері үшін болжау

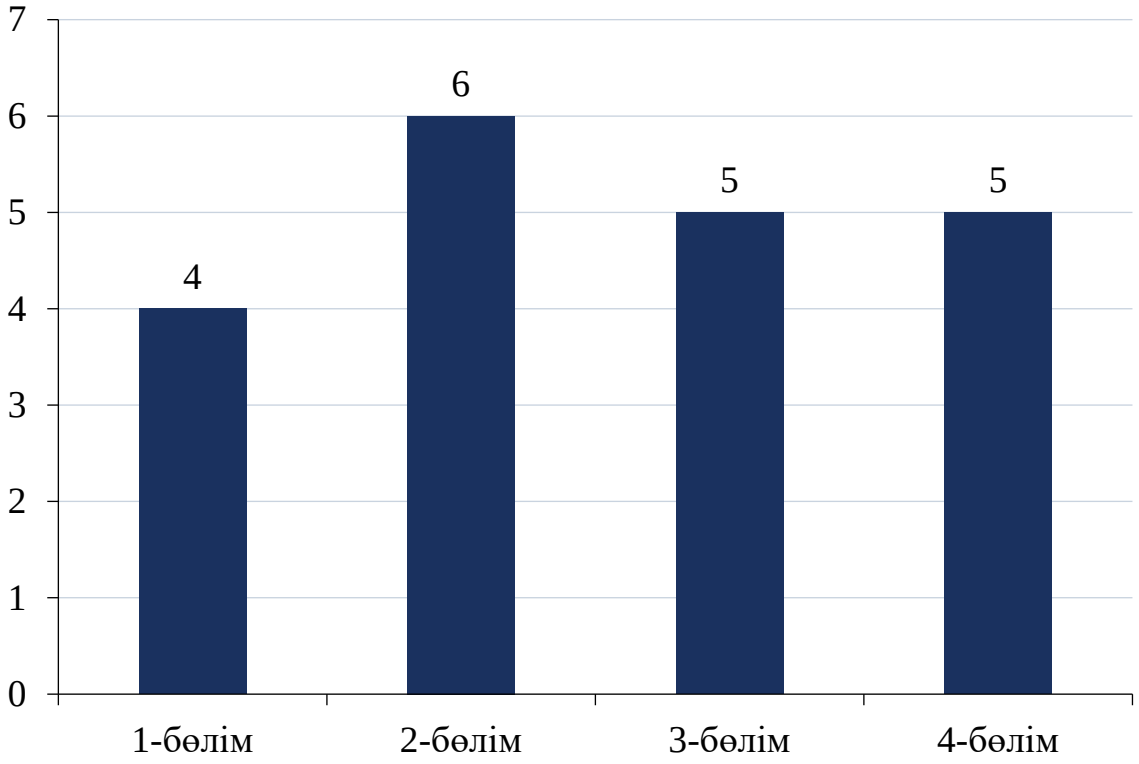
k параметрін таңдау

k тым аз	k тым үлкен	Оңтайлы k
Болжам мәндері үлкен таралады	Модель қатты өзгереді	Қате жіктеу ықтималдығы аз, көршілер сұрау нүктесіне жақын

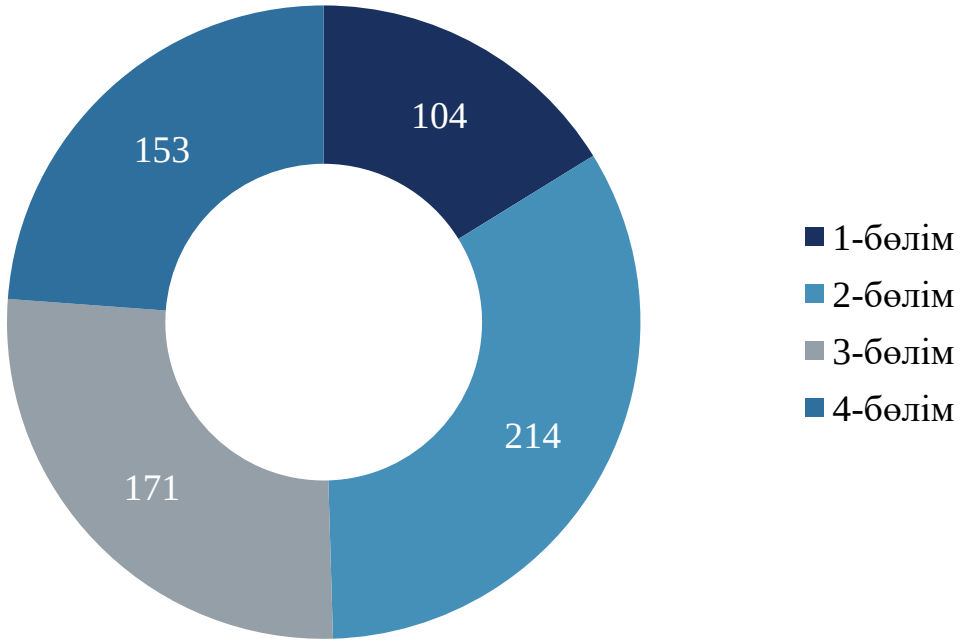
k - тегістеу параметрі: модельдің таралуы мен ығысуы арасындағы ымыра

Диаграммалар

Бөлімдер бойынша слайд саны



Бөлімдер бойынша мәтін көлемі (сөз саны)



Қорытынды

1. SVM екі класты бөлетін гиперпланды іздейді, тірек векторлар шекарада орналасады

2. Сызықты емес жағдайда ядро операторы өлшемді арттырады

3. k-жақын көрші әдісі жіктеуде көпшілік дауысты, болжауда орташа мәнді қолданады

4. k параметрінің оңтайлы мәні модель сапасын анықтайды

Тірек векторлар әдісі. "Жақын көрші" әдісі

Дәріс 9