ДЕРЕКТЕРДІ ИНТЕЛЛЕКТУАЛДЫ ТАЛДАУ

Дәріс 2

Индекстеу, сүзгілеу және түрлендірулер

Оспан Ә.Ғ., PhD, доцент-зерттеуші

ҚазҰУ, Жасанды интеллект және Big Data кафедрасы

2026-2027 оқу жылы

Дәрістің мақсаты мен мазмұны

Мақсат және оқу нәтижелері

- Белгілер, орындар және шарттар бойынша таңдауды ажырату.
- Шекараларды, ішкі таңдама көлемін және экстремумның барлық тең мәнін тексеру.
- Сәйкестікті жоғалтпай белгілер құру және жазбаларды сұрыптау.

Дәріс мазмұны

1. **loc және iloc**
2. **Логикалық маскалар**
3. **Шектер**
4. **Экстремумдар**
5. **Түрлендірулер**

Шартты сұрақ бірнеше әрекеттен тұрады

1

Жиынды анықтау

Шек толық файл не жарияланған топ бойынша анықталады.

2

Шекті есептеу

Мәнді және оның мағынасын бір рет сақтап алу.

3

Маска құру

Шекараларды және таңдалған жол санын тексеру.

4

Көрсеткішті есептеу

Қажетті белгінің орташа мәнін есептеп, қорытынды шектеуін сипаттау.

- Амалдардың реті — аналитикалық сұрақтың бір бөлігі.
- Қорытынды санды есептемей тұрып таңдама құрамын тексереміз.

loc және iloc: белгілер мен орындар

Сұрақ	Жазылуы	Не таңдалады
Жол белгісі және өріс атауы	df.loc[20, 'Age']	Бір ұяшық
Екінші жол және бірінші өріс	df.iloc[1, 0]	Ағымдағы реттегі орын
Алғашқы екі жол	df.iloc[:2]	0 және 1 орындары
10-нан 30-ға дейінгі белгілер	df.loc[10:30]	Қолайлы индексте екі шекара да кіреді
Бір жолды кесте түрінде	df.loc[[20], ['Age']]	Пішіні 1 де 1 DataFrame

Бір жазбаға апаратын екі жол

Python мысалы

```
import pandas as pd

demo = pd.DataFrame({
    'Name': ['Ada', 'Boris', 'Chen', 'Dina'],
    'Age': [19, 27, 22, 31],
    'Wage': [10, 40, 20, 30],
    'SprintSpeed': [80, 65, 75, 60]
}, index=[10, 20, 30, 40])
print(demo.loc[20, 'Name'])
print(demo.iloc[1, 0])
```

Нәтиже: Boris; Boris

Белгі бойынша

- 20 — индекстің мәні.
- Өріс атауы екінші таңдауды анықтайды.

Орын бойынша

- 1 екінші жолды білдіреді.
- 0 бірінші бағанды білдіреді.

Логикалық маска: әр жолға арналған шешім

Маска құрылымы

- True және False мәндері бар Series.
- Индекс бастапқы жолдармен байланысқан.
- sum ақиқат шарттар санын береді.

Қолдану

- df.loc[mask, columns] жолдар мен өрістерді анықтайды.
- Таңдау бастапқы кестені өзгертпейді.
- Іріктеуден кейін көлем мен мәндерді тексереміз.

Әр жол жарияланған іріктеу ережесіне дәл сәйкес логикалық шешім алуы тиіс.

Шарттарды біріктіру: «және» мен «немесе»

Python мысалы

```
young_fast = (demo['Age'] < 25) & (demo['SprintSpeed'] >= 75)
edge_age = (demo['Age'] < 20) | (demo['Age'] > 30)
print(demo.loc[young_fast, 'Name'].tolist())
print(demo.loc[edge_age, 'Name'].tolist())
print(young_fast.sum())
```

Нәтиже: ['Ada', 'Chen']; ['Ada', 'Dina']; 2

Қиылысу: &

- Екі шарт та ақиқат болуы тиіс.
- Әр салыстыруды жақшаға аламыз.

Бірігу: |

- Бір ақиқат шарт жеткілікті.
- ~ элемент бойынша терістеуді береді.

Аралықтар, санаттар және терістеу

Міндет	Тәсіл	Нені нақтылау керек
Аралық ішінде	s.between(20, 30)	Шекаралардың кіруі
Санаттардың бірі	s.isin(['GK', 'ST'])	Санаттардың дәл жазылуы
Сәйкестіктерден басқасы	~mask	Белгісіз мәндердің мағынасы
Максимумға дәл тең	s.eq(s.max())	Барлық тең максимумды сақтау

Толық есептеуді бастамай тұрып шекаралық жолды қолмен тексерген пайдалы.

Ішкі таңдама орташа мәні оның құрамына тәуелді

mean(selected) = sum(x_i) / n_selected

Шек

- Wage.mean() толық кесте бойынша есептеледі.
- Қатаң топтарға орташа мәнге теңдер кірмейді.

Тексеру

- Шектен жоғары, төмен және оған тең сандар қажет.
- Бос топтың орташа мәні анықталмаған.

demo ішінде: орташа Wage = 25; шектен жоғары SprintSpeed = 62.5, төмен = 77.5.

football.csv ішінде шартты есептеуді қайталаймыз

Python мысалы

```
from pathlib import Path

football = pd.read_csv(Path('data') / 'football.csv')
threshold = football['Wage'].mean()
mask = football['Wage'] > threshold
selected = football.loc[
    mask, ['Name', 'Wage', 'SprintSpeed']
]
print('Жол саны:', len(selected))
print('Орташа рейтинг:', selected['SprintSpeed'].mean())
```

Алдымен іріктеу

- Шек толық файл бойынша алынған.
- Нәтижеде қажетті өрістер қалады.

Содан кейін түсіндіру

- Жазба санын және орташа рейтингті көрсетеміз.
- Салыстыру себепті дәлелдемейді.

Экстремум: бір белгі ме, барлық көшбасшы ма?

Бір жазба

- idxmax бір белгі қайтарады.
- Жарияланған ереже бойынша бір жол қажет болса жарайды.
- Өздігінен тең максимумдарды тізбелемейді.

Барлық сәйкестік

- s.eq(s.max()) көшбасшылар маскасын береді.
- Аттарды және тексерілетін мәнді таңдаймыз.
- Топ үшін максимум сол топ ішінде есептеледі.

Экстремум таңдалған жиын ішінде анықталады және барлық сәйкестік сақталады.

Екі орташа мәннің қатынасы: бөлуден бұрын тексеру

1

Топтарды анықтау

Мысалы, алдымен Position == GK, содан кейін дағды максимумдары.

2

Тең максимумдарды сақтау

Қажетті максимумға тең барлық жолды қосу.

3

Сандарды тексеру

Екі топ та бос емес; бөлімдегі орташа мән нөлге тең емес.

4

Бөлу және сипаттау

Екі орташа мәнді, жазба сандарын және қатынас ретін көрсету.

- Топтар шарттарға сәйкес қиылысса, бұған жол беріледі.
- NaN орнына нөл қойып, нәтиженің жоқтығын жасыруға болмайды.

Жаңа белгі және толық жазбаларды сұрыптау

Python мысалы

```
work = football[['Name', 'Age', 'Wage']].copy()
work['AgeCentered'] = work['Age'] - work['Age'].mean()
work['IsYoung'] = work['Age'] < 23
ranked = work.sort_values(
    ['Wage', 'Name'], ascending=[False, True]
)
print(ranked.head(3))
print(round(work['AgeCentered'].mean(), 10))
```

Нәтиже: AgeCentered орташа мәні 0-ге жақын.

Түрлендіру

- Көшірме жұмыс кестесін бөледі.
- Жаңа бағанға формула нәтижесі айқын беріледі.

Рет

- sort_values толық жолдарды ауыстырады.
- .iloc енді жаңа ретке қатысты.

Таңдалған көрсеткіштердің мағынасын тексеру

Penalties

- Пенальти орындау дағдысының рейтингі.
- Гол санауға оқиғалар мен бақылау кезеңі қажет.

Максимумның 90%-ы

- Шек көрсетілген топ бойынша 0.9 * max тең.
- Бұл рейтинг > 90 шартынан өзгеше.

Wage және Value

- Бірліктері мен кезеңі расталмаған.
- Салыстыру тарихи файлға қатысты.

Жауапты сақтар алдында нені тексеру керек?

Логика

- Әр салыстырудың айналасында жақша бар.
- Маска ағымдағы кестеге қатысты.

Шекаралар

- Шектің қатаңдығы айқын жазылған.
- Бос топтар мен тең максимумдар ескерілген.

Өзгерістер

- Шартты меншіктеуге бір .loc қолданылады.
- Дөңгелектеу соңында орындалады.

Есепте шарт, жазба саны және есептелген көрсеткіштің мағынасы көрінуі тиіс.

Шартты талдаудың бес ережесі

1. loc белгілерді, iloc ағымдағы реттегі орындарды қолданады.
2. Құрама маска &, | және әр салыстыруға жеке жақша арқылы құрылады.
3. Шек пен экстремум алдын ала көрсетілген жиын бойынша есептеледі.
4. Бос топтар, тең максимумдар және нөлдік бөлім қорытындыға дейін тексеріледі.
5. Жаңа белгі мен сұрыптау жазбаның барлық өрісінің байланысын сақтауы тиіс.

Іріктеу мен түсіндіруді тексеріңіз

1. Неліктен loc[20] және iloc[20] әртүрлі әрекетті білдіруі мүмкін?
2. Орташадан жоғары топ орташадан кем емес топтан қалай ерекшеленеді?
3. Максимал мәннің барлық иесін қалай сақтаймыз?
4. Екі топтық орташа мәннің қатынасынан бұрын нені тексеру қажет?
5. Неліктен Penalties қосындысы гол санын бермейді?

Өзіндік жұмыс

Екі тең максимумы және сүзгілеуден кейін бос тобы бар кесте құрыңыз. Барлық көшбасшыны алыңыз және бос топтың орташа мәні орнына түсінікті хабар шығарыңыз.

Қол жеткізу мен түрлендіру дереккөздері

Құжаттама

1. pandas: индекстеу және іріктеу

https://pandas.pydata.org/docs/user_guide/indexing.html

2. pandas: негізгі амалдар

https://pandas.pydata.org/docs/user_guide/basics.html

3. pandas: жетіспейтін мәндер

https://pandas.pydata.org/docs/user_guide/missing_data.html

Дәрістің негізі

- Берілген 2-дәрістің презентациясы
- Берілген 2-зертханалық жұмыстың тапсырмалары
- Берілген 2-зертханалық жұмыстың жауаптары бар нұсқасы