ДЕРЕКТЕРДІ ИНТЕЛЛЕКТУАЛДЫ ТАЛДАУ

Дәріс 5

Кестелерді біріктіру: merge және concat

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

- Кілттерді, merge түрін және күтілетін кардиналдылықты таңдау.
- Сәйкестендірілмеген кілттер мен көбейген оқиғаларды табу.
- Схема мен индексті тексеріп, бөліктерді concat арқылы жинау.

Дәріс мазмұны

1. **Кілттер**
2. **merge түрлері**
3. **Кардиналдылық**
4. **Қайшылықтар**
5. **concat және файлдар**

Журнал мен анықтамалық әртүрлі ақпарат сақтайды

Оқиғалар журналы

- Нысан кілтінің қайталануына жол беріледі.
- Оқиғаның өз идентификаторы бар.
- Бір модель көптеген оқиғаға қатыса алады.

Нысандар анықтамалығы

- Нысан кілті әдетте бірегей.
- Түсіндірмелер мен тұрақты атрибуттарды сақтайды.
- Бірнеше нұсқа айқын таңдау ережесін қажет етеді.

Анықтамалықты жалғағаннан кейінгі жол деңгейі кез келген қосындыдан бұрын тексеріледі.

Кілт пен кардиналдылық байланыс талабын анықтайды

Байланыс	Сол жақтағы кілт	Оң жақтағы кілт	validate
Бірге бір	Бірегей	Бірегей	one_to_one
Көпке бір	Қайталануы мүмкін	Бірегей	many_to_one
Бірге көп	Бірегей	Қайталануы мүмкін	one_to_many
Көпке көп	Қайталанады	Қайталанады	many_to_many

merge алдында бір жолдың мағынасын және кілттің күтілетін қайталану санын атаймыз.

left сәйкестігі жоқ оқиғаларды сақтайды

Python мысалы

```
import pandas as pd

events = pd.DataFrame({
    'event_id': [101, 102, 103],
    'item_id': [1, 1, 3], 'amount': [10, 20, 30]
})
catalog = pd.DataFrame({
    'item_id': [1, 2, 4], 'label': ['Alpha', 'Beta', 'Delta']
})
joined = events.merge(
    catalog, on='item_id', how='left',
    validate='many_to_one', indicator=True
)
print(joined)
```

Нәтиже: event_id 103: label жоқ, _merge = left_only.

Сақталу

- Үш оқиға да нәтижеде қалады.
- 3 кілті сипаттаманың жетіспейтін мәнін алады.

Талап

- on кілтті айқын береді.
- validate оң жақтағы күтілетін бірегейлікті тексереді.

Бір мысалдағы merge төрт түрі

how	Не сақталады	Мысалдағы жол саны
left	Барлық сол жақ оқиға және олардың сәйкестігі	3
right	Барлық оң жақ нысан және олардың сәйкестігі	4
inner	Тек сәйкес кілттер	2
outer	Екі жақтың кілттері	5

Біріктіру түрі ыңғайлы көлемге емес, қажетті қамтуға сай таңдалады.

indicator біріктіруді тексеруге айналдырады

1

Біріктіру

on, how, validate және indicator=True параметрлерін айқын беру.

2

Мәртебелерді ажырату

both, left_only, right_only әртүрлі жағдайды білдіреді.

3

Ерекшелікті түсіндіру

Толық емес анықтамалық, оқиғалардың жоқтығы не басқа себеп.

4

Тексеру есебін сақтау

Кілт, жақ және түсіндірме тексерілетін күйде қалады.

- Оқиғасы жоқ оң жақ нысан жаңа сату жасамайды.
- Жетіспейтін идентификаторды ойдан шығарылған нөлдік ID-мен алмастырмаймыз.

Кілт қайталанса, қайталану сандары көбейтіледі

rows_for_key = n_left * n_right

Қайталану саны

- n_left: бір кілттің сол жақтағы жол саны.
- n_right: оң жақтағы сәйкестіктер саны.

Тәуекел

- Әр сол жақ жол оң жақ нұсқалар үшін қайталанады.
- Оқиға қосындылары жасанды ұлғаюы мүмкін.

2 сол жақ жол * 3 оң жақ жол = 6 үйлесім.

many-to-many нәтижесі байланыс мағынасы негізделгенде ғана орынды.

validate анықтамалық қайшылығын анықтайды

Python мысалы

```
left = pd.DataFrame({'key': [1, 1], 'x': ['a', 'b']})
right = pd.DataFrame({'key': [1, 1, 1], 'y': [10, 20, 30]})
expanded = left.merge(right, on='key', how='inner')
print(len(expanded))
try:
    left.merge(right, on='key', validate='many_to_one')
except pd.errors.MergeError:
    print('Оң жақ кілт бірегей емес')
```

Нәтиже: 6; Оң жақтағы кілт бірегей емес

Бақылау

- Талапсыз алты жол шығады.
- Жол санының өзі байланыс дұрыстығын түсіндірмейді.

Тексеру

- many_to_one оң жақ кілтінің бірегейлігін талап етеді.
- Қатені басу емес, себебін анықтау қажет.

Қайталанған кілт: алдымен себеп, кейін шешім

1

Қайталануларды табу

Айқын кілт бойынша duplicated, keep=False.

2

Өрістерді салыстыру

Дәл көшірмелер, әртүрлі нұсқалар не қайшылықтар.

3

Ережені анықтау

Расталған көшірме, күнге қатысты өзектілік не бөлек нысандар.

4

Қайта тексеру

Бірегейлік, сәйкестік мәртебелері және бақылау қосындылары.

- merge сәтті орындалуы үшін ғана қайшы нұсқаларды жоймаймыз.
- Таңдау ережесі қорытынды талдауға дейін негізделуі тиіс.

Ұқсас кілт бірдей нысанды әлі білдірмейді

Кілтті тексереміз

- Екі жақта да пәндік мағынасы бірдей.
- Типтері үйлесімді, бастапқы нөлдердің маңызы ескерілген.
- Бос кілттер біріктіруге дейін бөлінеді.

Атрибуттарды тексереміз

- on ортақ атаулардан болжанбай, айқын беріледі.
- suffixes бірдей өрістердің шығу тегін көрсетеді.
- Name ойыншының сенімді ID мәні деп саналмайды.

concat журнал бөліктерінің жолдарын жинайды

Python мысалы

```
part_a = events.iloc[:2].copy()
part_b = events.iloc[2:].copy()
all_events = pd.concat([part_a, part_b], ignore_index=True)
assert len(all_events) == len(part_a) + len(part_b)
assert all_events['event_id'].is_unique
print(all_events['amount'].sum())
```

Нәтиже: 60

Мақсаты

- Схемасы бірдей бөліктердің жолдарын жинау.
- ignore_index жаңа техникалық индекс жасайды.

Бақылау

- Жол саны бөліктер қосындысына тең.
- Оқиға кілттері күтпеген жерден қайталанбауы тиіс.

Бағандар бойынша concat индекстерді сәйкестендіреді

Амал	Сәйкестік негізі	Негізгі тексеру
merge	Кілт бағаны не өрістер жиыны	Кілт мағынасы және кардиналдылық
concat axis=0	Бағандардың бірдей схемасы	Бөліктердің қиылысуы және типтер
concat axis=1	Жолдардың индекс белгілері	Индекстің ортақ мағынасы мен сәйкестігі
Ұзындықтың жай теңдігі	Тек жол саны	Нысандарды байланыстыруға жеткіліксіз

Бірнеше файл: айқын тізім және ортақ схема

Python мысалы

```
from pathlib import Path

football = pd.read_csv(Path('data') / 'football.csv')
folder = Path('work_results') / 'lecture05_parts'
folder.mkdir(parents=True, exist_ok=True)
small = football[['Name', 'Age', 'Position']].head(12)
small.iloc[:6].to_csv(folder / 'part_01.csv', index=False)
small.iloc[6:].to_csv(folder / 'part_02.csv', index=False)
paths = sorted(folder.glob('part_*.csv'))
frames = [pd.read_csv(path) for path in paths]
assert frames
assert all(f.columns.tolist() == list(small) for f in frames)
combined = pd.concat(frames, ignore_index=True)
pd.testing.assert_frame_equal(combined, small)
```

Нәтиже: 12 жол және 3 баған қалпына келтірілді.

Файлдар

- Бөліктер жергілікті құрылады және ортақ схемасы бар.
- Файл реті сұрыптаумен бекітілген.

Жинау

- Алдымен кестелер тізімі, кейін бір concat.
- Салыстыру дәл қалпына келтіруді растайды.

Біріктіруден кейінгі тексерулер

merge

- Айқын кілт, how және validate.
- Сәйкестік мәртебелері және оқиғалар бірегейлігі.

Қосындылар

- Алдыңғы және кейінгі бақылау көлемі.
- Оқиғалардың үлесі жасанды қайталанбайды.

concat

- Схема, файл реті және бөліктердің қиылысуы.
- Индекс нақты сәйкестендіру ережесіне сәйкес.

Оқиғалардың сақталуын қорытынды кесте көлемімен ғана емес, кілттермен және қосындылармен тексеріңіз.

Деректерді біріктірудің бес ережесі

1. Біріктіру кілтінің екі кестедегі мағынасы бірдей болуы тиіс.
2. left сәйкестігі жоқ сол жақ жолдарды сақтайды; inner алып тастайды.
3. Кілт қайталануы қайталану сандарының көбейтіндісін береді, validate талапты тексереді.
4. Нұсқаларды еркін жоймай тұрып анықтамалық қайшылығын зерттейді.
5. concat келісілген схеманы, индексті және қиылысуды бақылауды талап етеді.

Кілттер мен оқиғалардың сақталуын тексеріңіз

1. left кезінде сәйкестігі жоқ сол жақ жолға не болады?
2. Бір кілттің екі сол жақ және үш оң жақ жазбасы неше жол береді?
3. validate="many_to_one" оқиғалар журналын қалай қорғайды?
4. Кілт бойынша drop_duplicates неліктен кез келген қайшылықты түзетпейді?
5. concat(axis=1) жолдарды не бойынша сәйкестендіреді?

Өзіндік жұмыс

Бір жолды жоғалтып, екіншісін көбейтетін, жалпы көлемді сақтайтын inner құрыңыз. Мәселені оқиға ID мәндері, indicator және бақылау қосындысы арқылы дәлелдеңіз.

merge, concat және жолдар жөніндегі дереккөздер

Құжаттама

1. pandas: merge және concat

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

2. pandas: DataFrame.merge параметрлері

https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.merge.html

3. Python: pathlib

https://docs.python.org/3/library/pathlib.html

Дәрістің негізі

- Берілген 5-дәрістің кестелерді біріктіру жөніндегі презентациясы
- Берілген 5-зертханалық жұмыстың тапсырмалары
- Берілген 5-зертханалық жұмыстың жауаптары бар нұсқасы