Лабораторна робота №3

Структури даних Pandas

Мета роботи: Ознайомитись з основними структурами даних бібліотеки Pandas: Series DataFrame, операціями над ними. Навчитись використовувати групування.

Короткі теоретичні відомості

Об'єкт Series бібліотеки Pandas - одновимірний масив індексованих даних. Його можна створити зі списку або масиву наступним чином:

a = pd.Series([1,2,3,4])

У той час як індекс масиву NumPy, який використовується для доступу до значень, - цілочисельний і описується неявно, індекс об'єкта Series бібліотеки Pandas описується явно і зв'язується зі значеннями. Цей індекс не обов’язково має бути цілим числом, а може складатися з значень будь-якого типу.

a = pd.Series([1, 2, 3, 4],index=['I', 'II', 'III', 'IV'])

Для уникнення плутанини з прямими та непрямими індексами через те, що індекси можуть мати назви, що співпадають з номерами, існують спеціальні атрибути, які прямо вказують на тип індексації.

Для прямої індексації використовується атрибут loc:

b= pd.Series(['2', '8', '16', '20'], index=[1, 4, 8, 10])

b.loc[1:4]

Для непрямої індексації використовується iloc:

b.iloc[1:4]

Наступна базова структура бібліотеки Pandas - об'єкт DataFrame.

Якщо об'єкт Series - аналог одновимірного масиву з гнучкими індексами, об'єкт DataFrame - аналог двовимірного масиву з гнучкими індексами рядків і гнучкими іменами стовпців.

a=np.random.randint(50,size=(4))

b=np.random.randint(50,size=(4))

frame = pd.DataFrame({'a': a, 'b': b })

Результатом методу pd.read_csv() є об’єкт DataFrame.

До об’єкту DataFrame можна додавати нові стовпці, виділяти підмасиви

data = pd.read_csv('Fish.csv')

data['ratio'] = data['Width']/(data['Height'])

data[(data.Weight>200)&(data.Weight<300)]

Можна створити індекс зі значень стовпця:

data = pd.read_csv('Fish.csv’)

data.set_index(['Species'])

Для об’єкту DataFrame можна викликати різні функції агрегування, в тому функції дескриптивної статистики, наприклад, mean чи std. Якщо одночасно потрібні всі основні статистичні характеристики, можна використати метод describe. Цей метод має параметр include.

data.describe()

Якщо є потреба отримати значення функцій агрегування (а також деяких інших) відносно підмасивів, виділених за певною ознакою, можна використати метод groupby. Сам метод groupby по суті створює об’єкт DataFrameGroupBy, у якого вже можна викликати інші методи.

data = pd.read_csv(‘Fish.csv’)

data.groupby(‘Species’).mean()

Об’єкт DataFrameGroupBy має метод aggregate() або agg(), що дозволяє, наприклад, виконувати кілька операцій агрегування:

data.groupby(‘Species’)[‘Weight’].agg([sum,np.prod,np.mean])

Або застосовувати різні операції до різних стовпців:

data.groupby(‘Species’).agg({‘Weight’: max,’Height’:np.median})

Функція pd.merge () реалізує безліч типів з'єднань: «один-до-одного», «багато-до-одного» і «багато-до-багатьох». Всі ці три типи з'єднань доступні через один і той же виклик pd.merge (), тип виконуваного з'єднання залежить від форми вхідних даних.

Якщо назви стовпців відрізняються, то вони вказуються в параметрах left_on та right_on. Зайвий стовпець можна видалити за допомогою методу drop.

a = pd.DataFrame({'City': ['A', 'B', 'C', 'D'],'population': [1000000, 245653,355223,4532434]})

b = pd.DataFrame({'Town': ['A', 'C', 'D', 'B'],'region': ['a', 'b', 'c', 'c']})

c=pd.merge(a, b, left_on="City", right_on="Town").drop('Town',axis=1)

Можна виконувати злиття за індексом за допомогою методу join(). Той самий результат вийде, якщо встановити як True параметри left_index, right_index методу merge().

Параметр how методу merge вказує яка саме операція виконується над вмістом стовпця, за яким відбувається злиття. За замовчанням його значення 'inner', тобто перетин. Інші можливі значення - 'outer', 'left’ та 'right’.

Якщо дані містять однакові назви стовпців (за якими не відбувається злиття), то результат буде містити обидва ці стовпця з автоматично доданим суфіксом. Суфікс можна додати самостійно за допомогою параметру suffixes.

Завдання до лабораторної роботи

Створити програму, яка за даними файлу відповідно до варіантів лабораторної №2, виконує наступні завдання:

1. Виділити один зі стовпців (на вибір) з файлу як об’єкт Series, виділити з нього підмасив. Задати назви індексів цього об’єкту. Виділити підмасиви за допомогою прямої та непрямої індексацій.
2. До об’єкту DataFrame, в який записано вміст файлу, додати новий стовпець, що є результатом операцій над іншими стовпцями. Також продемонструвати додавання та видалення рядків, видалення стовпців.
3. Встановити один зі стовпців індексом. Визначити основні статистичні характеристики та типи даних всіх стовпців. Змінити тип даних для одного з стовпців. Згрупувати дані за одним зі стовпців, застосувати кілька агрегуючих функцій, виділити підмасив за певними ознаками.
4. Створити декілька власних об’єктів DataFrame за такою ж тематикою, що й файл. Наприклад, якщо тема файлу – жаби, можна створити об’єкти, що містять розміри жаб, вагу, стать, кількість особин в популяції і т.д. Використати описані в теоретичних відомостях параметри методів merge та concat для різних видів злиття та об’єднання даних цих об’єктів.

Оформити звіт. Звіт повинен містити:

— титульний лист;
— код програми;
— результати виконання коду.

Продемонструвати роботу програми та відповісти на питання стосовно теоретичних відомостей та роботи програми.