УДК 004.657

АНАЛИЗ МАССИВОВ ДАННЫХ С ИСПОЛЬЗОВАНИЕМ БИБЛИОТЕКИ PANDAS ДЛЯ PYTHON

Ильичев В.Ю., Юрик Е.А.

[1]Калужский филиал ФГОУ ВО «Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)», Калуга, e-mail: patrol8@yandex.ru

Целью работы, описанной в данной статье, являлась разработка алгоритма обработки больших массивов данных с помощью свободно распространяемого языка программирования Python с подключением библиотеки Pandas, предназначенной для работы с базами данных. Для решения этой задачи использована информация, размещённая на форумах сети Интернет, а также в документации к перечисленным программным продуктам. При создании алгоритма ставилась цель формализации последовательности команд обращения к библиотеке Pandas и действий, производимых этими командами для создания выборки только тех данных из массива, которые необходимы исследователю. После этого рассмотрен процесс сортировки значений из полученной выборки. Такое пошаговое описание алгоритма необходимо для понимания последовательности процесса обработки данных при создании научных и прикладных программных продуктов на языке Python. Использование разработанного алгоритма продемонстрировано на примере обработки массива данных по продажам крупных американских компаний за один год. Для вывода результатов обработки в виде диаграммы применены команды управления графической библиотекой Matplotlib для Python. Ключевой особенностью полученного алгоритма является простота его применения для обработки данных любого типа, представленных в разных форматах – как популярных, так и экзотических. Показаны уникальные возможности библиотеки Pandas, которыми не обладают традиционные средства обработки баз данных, основанные на SQL (языке структурированных запросов, работающем только с реляционными базами данных). Дано заключение о проделанной работе и рекомендации по сферам применения её результатов. В частности, предполагается дальнейшее использование и развитие данного алгоритма для расчёта мощности фотоэлектрических солнечных и ветряных электростанций по статистической информации о характеристиках солнечного излучения и атмосферных потоков в определённом регионе с подключением дополнительных библиотек.

Ключевые слова: база данных, SQL, алгоритм, программа, язык Python, библиотеки Python, библиотека Pandas

ANALYSIS OF DATA ARRAYS USING THE PANDAS LIBRARY FOR PYTHON

Ilichev V.Yu., Yurik E.A.

Kaluga Branch of Bauman Moscow State Technical University, Kaluga, e-mail: patrol8@yandex.ru

The purpose of the work described in this article was to develop an algorithm for processing large amounts of data using a freely distributed Python programming language with a Pandas library attached to work with databases. Information posted on Internet forums, as well as documentation for these software products, is used to solve this problem. When creating the algorithm, the goal was to formalize the sequence of commands to access the Pandas library and the actions these commands perform to create a sample of only the data from the array that the researcher needed. The process of sorting the values of the obtained sample is then considered. This step-by-step description of the algorithm is necessary to understand the sequence of the data processing when creating scientific and application software products in Python. The use of the developed algorithm is demonstrated by the example of processing an array of data on sales of large American companies in one year. The Matplotlib graphics library management commands for Python are used to display the results of processing as a chart. A key feature of the resulting algorithm is its ease of use for processing data of any type presented in different formats – both popular and exotic. Shows the unique capabilities of the Pandas library that traditional SQL-based database processing tools do not have (a structured query language that only works with relational databases). An opinion on the work done and recommendations on the application of its results were given. In particular, it is expected to further use and develop this algorithm to calculate the capacity of photovoltaic solar and wind farms based on statistical information on the characteristics of solar radiation and atmospheric flows in a certain region with the connection of additional libraries.

Keywords: database, SQL, algorithm, program, Python language, Python libraries, Pandas library

Современный уровень развития технологий, производств и экономик предполагает обработку огромного количества информации. Часто такая информация представляется в виде очень больших массивов числовых и прочих символьных данных, нередко не отсортированных по каким-либо признакам. Задачей статистического анализа является сортировка и прочие виды обработки массивов с целью обнаружения и наглядного отображения характерных закономерностей, тенденций, выделения основных признаков исследуемого процесса или объекта. По результатам такого анализа обычно строят графики, представляемые для обсуждения как специалистам в данной отрасли, так и широкому кругу неспециалистов.

Традиционно для данных целей до недавнего времени широко использовался так называемый язык структурированных запросов SQL. Однако ему присущи следующие недостатки [1]: громоздкость программ, сложность изучения синтакси-

са и, главное, строгая ориентированность на определённую структуру загружаемых данных (реляционные базы данных). В то же время сейчас всё чаще используются данные, объединённые совершенно разными структурами. Также в SQL сложно реализовать обработку неоднородных типов данных или массивов с пропущенными данными [2].

Более простым в освоении, удобным и гибким в использовании для целей обработки данных как разных традиционных форматов (файлов Microsoft Excess, Excel и др.), так и нестандартных форматов представляется библиотека Pandas для языка программирования общего назначения Python [3]. В данном случае весь процесс создания приложения, использующего файлы данных, происходит с использованием одного языка программирования, позволяющего проектировать пользовательский интерфейс, обрабатывать и анализировать массивы и представлять результаты в виде наглядного и качественного графического материала. Кроме того, язык Python очень быстро развивается, при этом к настоящему времени существуют постоянно совершенствуемые и дополняемые библиотеки для решения широкого круга как универсальных, так и прикладных, узкоспециализированных задач; к тому же появляется много новых библиотек.

Сейчас на сайтах сети Интернет можно найти множество статистики по протеканию процессов и совершению действий в различных отраслях: на производстве, в медицине, в транспортных перевозках, энергетике, торговле и других областях окружающего мира. Статистическая информация, накопленная за длительный промежуток времени, обычно представляется в виде файлов большого объёма. Ручная обработка таких файлов не представляется возможной, так как потребует больших временных и трудовых затрат и связана с появлением субъективных ошибок.

Машинная обработка данных связана традиционно с привлечением специалистов именно в отрасли работы с данными, обладающими высокой квалификацией и опытом программирования на SQL. Специалистам в других отраслях полноценное освоение SQL представляется крайне проблематичным.

Раньше инструментальных возможностей SQL вполне хватало для анализа реляционных массивов данных – быстрого поиска нужных значений и создания по ним отчёта. В настоящее же время получили распространение данные, оформленные в виде разных форматов и структур. Это могут быть csv-файлы, обычный текст, форматы Parquet, HDF5 и другие [4]. Библиотека Pandas языка Python, являющаяся надстройкой над библиотекой Numpy, может работать со всеми этими форматами.

Целью описываемой работы является разработка последовательности достаточно простых операций (алгоритма) по обработке массивов данных, представленных в распространённых форматах, а также демонстрация примера такой обработки с использованием распространённого сейчас языка Python и подключаемых библиотек. Формального и доступного описания такого алгоритма не удалось найти ни в зарубежных, ни тем более в отечественных источниках информации.

Материалы и методы исследования

Алгоритм обработки данных разрабатывался с помощью изучения документации к библиотеке Pandas [5], примеров программ, приведённых на сайте [6] и зарубежных форумах программистов, а также апробации изложенных принципов на произведённых авторами практических вычислениях.

Разработанный алгоритм включает в себя следующие этапы:

1) чтение файла данных определённого формата;

2) выделение нескольких строк данных для обработки (по умолчанию 5) – так называемого кадра данных (data frame) для исключения переполнения памяти;

3) выбор из кадра данных только необходимых для исследования столбцов – определение столбцов, содержащих названия исследуемых объектов и значения исследуемого фактора;

4) группировка одинаковых объектов и определение правил обработки значений факторов для каждого объекта (например, суммирование численных значений фактора или счёт количества строк, содержащих информацию по объекту) и применение данного правила кадра для всех строк массива;

5) сортировка полученных строк результирующего массива по возрастанию или убыванию численных значений какого-либо выбранного столбца, содержащего сумму значений фактора исследования;

6) переименование выбранных столбцов, содержащих названия объектов и суммы факторов, для удобства дальнейшего их анализа и представления;

7) выбор определённого количества первых отсортированных объектов для дальнейшего отображения значений сумм фактора по ним;

account number	name	sku	quantity	unit price	ext price	date
740150	*Barton LLC*	*B1-20000*	*39*	*86,69*	*3380,91*	*2014-01-01 07:21:51*
714466	*Trantow-Barrows*	*S2-77896*	*-1*	*63,16*	*-63,16*	*2014-01-01 10:00:47*
218895	*Kulas Inc*	*B1-69924*	*23*	*90,7*	*2086,1*	*2014-01-01 13:24:58*
307599	*Kassulke, Ondricka and Metz*	*S1-65481*	*41*	*21,05*	*863,05*	*2014-01-01 15:05:22*
412290	*Jerde-Hilpert*	*S2-34077*	*6*	*83,21*	*499,26*	*2014-01-01 23:26:55*
714466	Trantow-Barrows	S2-77896	17	87,63	1489,71	2014-01-02 10:07:15
218895	Kulas Inc	B1-65551	2	31,1	62,2	2014-01-02 10:57:23
729833	Koepp Ltd	S1-30248	8	33,25	266	2014-01-03 06:32:11
714466	Trantow-Barrows	S1-50961	22	84,09	1849,98	2014-01-03 11:29:02
737550	Fritsch, Russel and Anderson	S2-82423	14	81,92	1146,88	2014-01-03 19:07:37
146832	Kiehn-Spinka	S2-82423	15	67,74	1016,1	2014-01-03 19:39:53
688981	Keeling LLC	S2-00301	7	20,26	141,82	2014-01-04 00:02:36
786968	Frami, Hills and Schmidt	S2-23246	6	61,31	367,86	2014-01-04 06:51:53
307599	Kassulke, Ondricka and Metz	S2-10342	17	12,44	211,48	2014-01-04 07:53:01
737550	Fritsch, Russel and Anderson	B1-53102	23	71,56	1645,88	2014-01-04 08:57:48

Рис. 1. Кадры данных исследуемого массива

8) выбор типа графического отображения данных;

9) построение графика либо диаграммы по выбранному отсортированному набору данных.

Большое количество массивов актуальных и прошлых данных, сформированных для разных интересных для исследования областей окружающего мира, к примеру, можно найти на сайте kaggle.com.

В качестве примера применения разработанного алгоритма рассмотрим обработку массива данных по продажам крупнейших американских компаний за 2014 г. Файл данных расположен по адресу https://github.com/chris1610/pbpython/blob/master/data/sample-salesv3.xlsx и представляет собой лист Microsoft Excel из 1500 строк. Три кадра данных (названия столбцов и 15 строк с данными) из данного файла приведены на рис. 1.

Рассмотрим пошаговое выполнение вышеуказанного алгоритма, применённого к данному примеру.

1. Выполняется команда обращения к массиву данных, содержащая путь к файлу (файл может быть в сети или в локальной папке на компьютере) и преобразующая формат Excel в формат Pandas.

2. Выполняется команда, загружающая в оперативную память 5 первых строк данных (на рис. 1 выделены курсивом).

Далее выполняется команда Pandas, одновременно реализующая пункты 3–5 алгоритма:

3. Так как в качестве результата исследования мы хотим выделить компании, получившие максимальную прибыль за 2014 г., то для анализа нам нужны только два столбца данных – столбец «name», содержащий названия компаний (объекты), и столбец «ext price», содержащий доход за каждый день продаж (фактор) – указываем это в команде.

4. Производим группировку строк по названиям компаний (объектов) – результат выполнения данной операции для трёх кадров показан на рис. 2. Затем по каждой компании суммируем доходы за весь год (сумма значений фактора).

name	ext price
Barton LLC	3380,91
Frami, Hills and Schmidt	367,86
Fritsch, Russel and Anderson	1146,88
Fritsch, Russel and Anderson	1645,88
Jerde-Hilpert	499,26
Kassulke, Ondricka and Metz	863,05
Kassulke, Ondricka and Metz	211,48
Keeling LLC	141,82
Kiehn-Spinka	1016,1
Koepp Ltd	266
Kulas Inc	2086,1
Kulas Inc	62,2
Trantow-Barrows	-63,16
Trantow-Barrows	1489,71
Trantow-Barrows	1849,98

Рис. 2. Результат выбора столбцов данных для исследования и группировки объектов

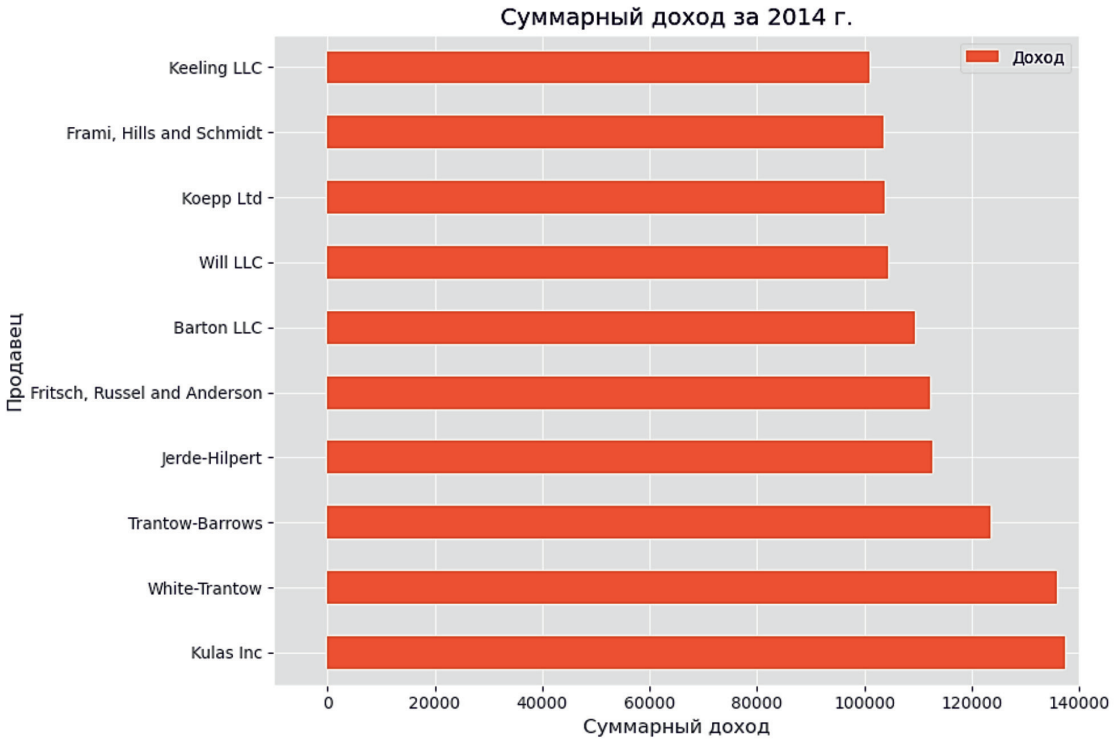

Рис. 3. Графический результат обработки массива данных

5. После выполнения команды сортировки максимальных сумм доходов по всему массиву данных в оперативной памяти формируется новый, результирующий, массив.

6. Следующей командой для удобства отображения переименовываем полученные столбцы «name», «ext price» следующим образом: «Компания», «Доход».

7. Выбираем 10 компаний с максимальным суммарным годовым доходом.

8. Этот и последующий пункт реализуется с подключением дополнительной графической библиотеки Matplotlib. В качестве типа графического отображения данных выбираем стиль «ggplot», реализующий популярный на данный момент набор графических элементов для подготовки иллюстраций научных публикаций, заимствованный из проекта обработки статистических данных «R Project». Аббревиатура «gg» переводится как «grammar of graphics», то есть определенная система строгих правил, позволяющих описывать и строить графики [7].

9. Задаём название графика, имена осей, нижний и верхний пределы отображаемых значений сумм факторов и выводим график на экран. Также пишем команду сохранения графика в виде файла рисунка.

Результаты исследования и их обсуждение

Результат выполнения программы приведён на рис. 3.

Полученный график очень наглядно показывает результат обработки массива данных и выборки наиболее успешных торговых компаний США, исходя из их суммарного годового дохода за 2014 г.

Подобные графики можно использовать для составления презентаций для научных докладов, для размещения на сайтах и т.п. Библиотека Matplotlib предоставляет практически неограниченные возможности создания любого пользовательского стиля отображения элементов графика, набора цветов, шрифтов и др.

Также имеется возможность вывода результатов в виде таблицы, которую можно использовать для дальнейшего анализа и слияния данных.

Обсуждению особенностей и преимуществ применения как языка Python, так и библиотек, расширяющих его возможности для решения всё новых задач, посвящено множество учебников и учебных пособий [8; 9], разделов и тем на форумах сети Интернет. Также рассмотрению этой тематики уделяется внимание на научных

конференциях, в том числе онлайн. При этом немного хаотичное, но очень активное развитие данного программного продукта объясняется тем, что он является некоммерческим, свободно распространяемым и используется для создания программ для решения задач в абсолютно разных областях науки и для прикладного использования. Эта возможность позволяет как множеству программистов, так и неспециалистов, вкладывать в прогресс этого программного продукта что-то новое. Опыт использования языка Python и подключаемых библиотек затем систематизируется в научных статьях и прочих изданиях. Можно сказать, что язык развивается «методом последовательных приближений и уточнений». Данная статья также преследует цель систематизации знаний, в основном по функционированию и практическому использованию библиотеки обработки и анализа данных Pandas.

Процесс написания и тестирования прикладного компьютерного приложения показал доступность и простоту реализации алгоритма обработки массива данных с помощью языка Python.

Основное предназначение разработанного алгоритма заключается в достижении лучшего понимания функционирования библиотеки Pandas. Демонстрация её базовых возможностей на примере призвана показать простоту использования и наглядность получаемых результатов.

Заключение

По результатам разработки алгоритма обработки и анализа массивов данных с использованием языка программирования Python и подключаемых библиотек Pandas и Matplotlib можно выделить следующие преимущества использования данной «связки» программных продуктов:

– разработка всей программы, начиная с загрузки данных и заканчивая выводом результата в графическом виде, осуществляется в одном приложении;

– возможность использования массивов данных, представленных практически в любом формате;

– широкие возможности по созданию выборки из массива только тех данных, которые необходимы исследователю;

– возможность сортировки данных по любому признаку;

– наглядное и качественное представление результатов исследований в виде графического материала (либо в виде таблиц, если необходимо);

– открытость и бесплатность программных кодов, простота их применения.

С учётом рассмотренных достоинств разработанный алгоритм можно рекомендовать для использования во всех отраслях научных и прикладных исследований. В частности, авторами планируется его применение для обработки баз данных по параметрам солнечного излучения и потоков воздуха в атмосфере определённых регионов с целью определения изменения мощности фотоэлектрических солнечных и ветровых электростанций. Для этого потребуется подключение дополнительных библиотек, которые уже имеются на профильных сайтах.

Список литературы

1. Быченков В.Ф. Кроссплатформенный подход к построению типовых решений обработки данных для SQL-ориентированных СУБД. Информатизация образования. 2012. № 1 (66). С. 56–73.

2. Erland Sommarskog. Arrays and Lists in SQL Server. [Электронный ресурс]. URL: http://www.sommarskog.se/arrays-in-sql.html (дата обращения: 29.05.2020).

3. Дж. Вандер Плас/ Python для сложных задач: наука о данных и машинное обучение. СПб.: Питер, 2018. 572 с.

4. Tiago Antao. Bioinformatics with Python Cookbook: Learn how to use modern Python bioinformatics libraries and applications to do cutting-edge research in computational biology, 2nd Edition. Packt Publishing Ltd, 2018. 360 p.

5. Pandas documentation. Date: Mar 18, 2020. Version: 1.0.3. [Электронный ресурс]. URL: https://pandas.pydata.org/docs/pandas.pdf (дата обращения: 29.05.2020).

6. Документация по библиотекам Python с примерами. [Электронный ресурс]. URL: https://pythonru.com/biblioteki (дата обращения: 29.05.2020).

7. Самсонов Т.Е. Визуализация и анализ географических данных на языке R. М.: Географический факультет МГУ, 2020. [Электронный ресурс]. URL: https://tsamsonov.github.io/r-geo-course (дата обращения: 29.05.2020).

8. Хайбрахманов С.А. Основы научных расчётов на языке программирования Python: учебное пособие. Челябинск: Изд-во Челябинского государственного. университета, 2019. 96 с.

9. Сысоева М.В., Сысоев И.В. Программирование для «нормальных» с нуля на языке Python. Учебник. В двух частях. Часть 1. М.: ООО «МАКС Пресс», 2018. 176 с.