Научная статья

Original article

УДК 004.89

СРАВНИТЕЛЬНЫЙ АНАЛИЗ СИСТЕМ ОБРАБОТКИ ДАННЫХ JUPYTER NOTEBOOK И ANAPLAN

COMPARATIVE ANALYSIS OF DATA PROCESSING SYSTEM: JUPYTER NOTEBOOK AND ANAPLAN

Абдрашитова Анжела Наилевна, студентка МГТУ им. Н.Э. Баумана, 3 курса кафедры «Системы обработки информации и управления», тел. 8(917)8768712, abdrashitova.anzhela@yandex.ru.

Муханов Ержан, студент МГТУ им. Н.Э. Баумана, 3 курса кафедры «Системы обработки информации и управления», тел. 8(925)0085249, mukhanov-yerzhan@mail.ru.

Anzhela. N. Abdrashitova, Bauman Moscow State Technical University third-year student of department « Information processing and management systems», tel. 8(917)8768712, abdrashitova.anzhela@yandex.ru.

Yerzhan Mukhanov, Bauman Moscow State Technical University third-year student of department « Information processing and management systems», tel. 8(925)0085249, mukhanov-yerzhan@mail.ru.

Аннотация. В статье рассмотрены два способа анализа данных: BI[1]-платформа Anaplan и Jupyter Notebook. Anaplan - облачная платформа, которая позволяет

[1] Business Intelligence

автоматизировать процессы бюджетирования и прогнозирования и предоставляет понятный для пользователя интерфейс. Jupyter Notebook - мощнейший инструмент для работы с данными, позволяющий обрабатывать, анализировать и визуализировать результаты. В данной работе мы рассматривали Anaplan не с точки зрения бизнес-аналитики, а в качестве инструмента для анализа данных. В последнее время платформа набрала большую популярность и используется для построения объемных моделей, в результате чего возникла необходимость проверки ее быстродействия. Цель исследования: проверить, действительно ли Anaplan осуществляет облачные вычисления, как при этом задействованы ресурсы ПК и какова скорость построения графиков. Для этого были созданы аналогичные модели в Jupyter Notebook и Anaplan, собраны метрики и проведены тесты. Датасет для построения моделей был взят с сайта Kaggle[2].

Abstract. The article discusses two ways to analyze data: BI-platform Anaplan and Jupyter Notebook. Anaplan - is a cloud computing platform, which provides an opportunity to automate processes of budgeting and business-planning and provides a convenient interface for users. Jupyter Notebook is a powerful tool for data processing which allows to process, analyze and illustrate results. In this work, we considered Anaplan not from the point of view of business-intelligence, however as a tool for data analysis. Recently, the platform has gained great popularity and is used to build large models, as a result of which it became necessary to test its performance. The purpose of the study: to check whether Anaplan really does cloud computing, how it uses PC resources and what is the speed of plotting. For this, similar models were created in Jupyter Notebook and Anaplan, metrics were collected and tests were carried out. The dataset for building models was taken from the Kaggle website.

Ключевые слова: BI-платформа, действия, датасет, маппинг, метрики, мониторинг, запрос.

Keywords: BI-platform, action, dataset. mapping, metrics, monitoring, query.

[2] https://www.kaggle.com/

Используемые системы обработки данных

Прежде чем начать строить модель в Anaplan, стоит рассказать о базовых правилах данной платформы[3]. Списки, модули и дэшборды - “3 кита”, на которых строится любая модель. Каждый модуль содержит line items[4] (далее - элементы), в которые можно поставить формулы для расчетов. Весь модуль в целом и каждый элемент по отдельности имеют dimension (далее - измерение), в качестве которого могут выступать временная шкала или списки. Список может быть flat (плоским) или содержать иерархию. Каждая ступень иерархии имеет родительский список, к которому мы добавляем необходимые значения. Готовая пользовательская страница показана на Рис. 1. Готовые графики в Jupyter Notebook иллюстрированы на рисунках 2, 3.



Рис. 1. Отображение графиков в Anaplan

[3] Введение в Anaplan - community.anaplan.com
[4] Названия указаны на английском из-за отсутствия русскоязычной версии сайта и интерпретации названий

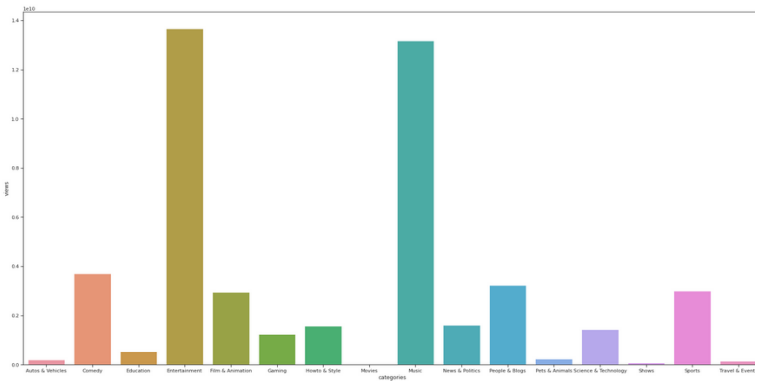


Рис. 2. Столбчатая диаграмма в Jupyter Notebook

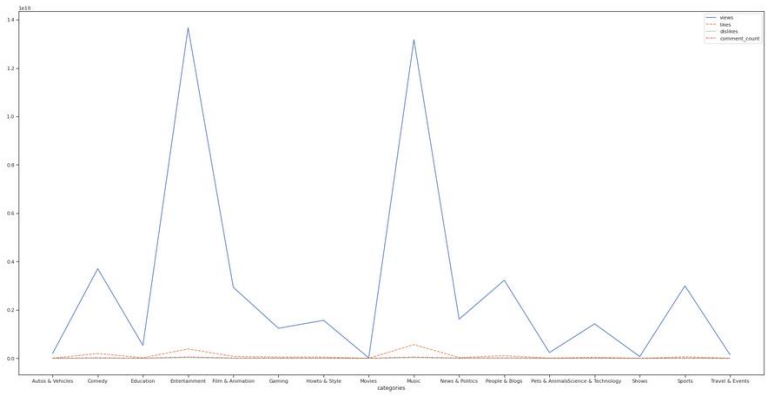


Рис. 3. Диаграмма распределения в Jupyter Notebook

Сбор метрик

Характеристики хост-машины:

- процессор - Intel® Core™ i5-9300H CPU @ 2.40GHz × 8
- оперативная память - DDR4 8 GB
- графический адаптер - GeForce GTX 1050 3 GB Max-Q

Для сбора метрик с систем анализа данных был выбран стек:

- Node exporter - небольшое приложение, собирающее метрики операционной системы и предоставляющее к ним доступ по HTTP.

Prometheus собирает данные с одного или нескольких экземпляров Node Exporter.

- Victoria metrics - получает метрики из разных сервисов и собирает их в одном месте. Запросы осуществляются удобным языком - PromQL. Выбрана в сравнении с Prometheus, поскольку использует значительно меньше ресурсов хоста.
- Grafana отображает данные из Prometheus в виде графиков и диаграмм, организованных в дашборды.

Было решено провести исследование времени работы Anaplan и Jupyter Notebook при помощи Python. Для взаимодействия систем анализа данных были использованы HTTP REST API, описанные в документации к обеим платформам[5]. Стоит отметить, что для работы API с Jupyter Notebook и запуском ядра и ячеек ноутбуков, необходимо было развернуть Jupyter kernelgateway.

Платформа Anaplan поддерживает REST API, для взаимодействия необходимо получение токенов доступа и обновления токена к веб-сервису, а также id целевых процессов и actions.

Алгоритм создания валидного запроса в Anaplan состоит из получения токена доступа, id процесса, запуска процесса.

Для получения нагрузки на машину (отправки PromQL[6] запроса на VictoriaMetrics), необходимо знать время получения данных и построения графиков. В результате работы создаются файлы с собранными метриками, которые будут использованы для обработки гипотез.

Полученные датасеты

Для проверки гипотез в каждый момент времени была вычислена разница между текущим и начальным значениями. Эти данные были записаны в столбцы cpu_difference, network_difference, ram_difference на рисунках 4 - 6.

[5] Источники указаны в Списке литературы

[6] Prometheus Query Language - язык запросов в Prometheus

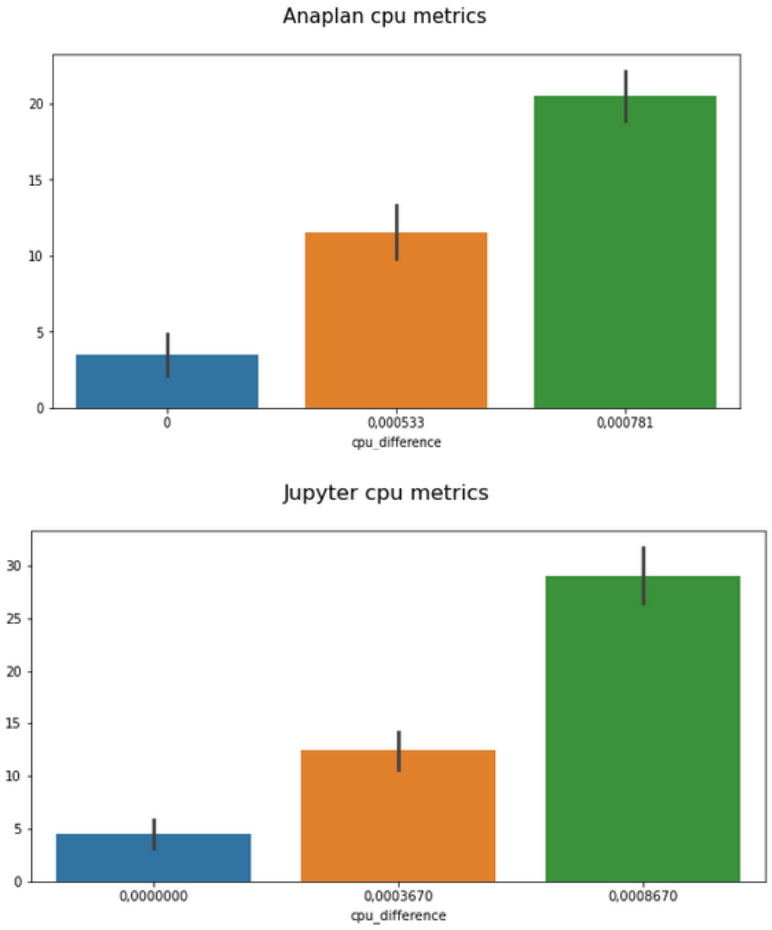


Рис. 4. Сравнение нагрузки на процессор в Anaplan и Jupyter Notebook

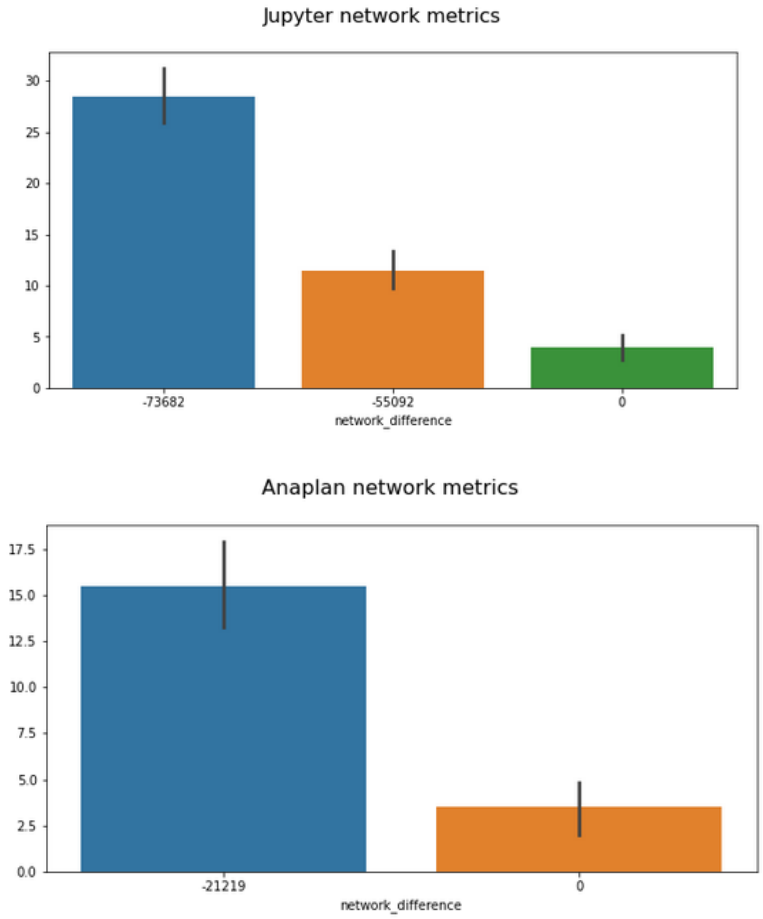


Рис. 5. Сравнение нагрузки на сеть в Anaplan и Jupyter Notebook

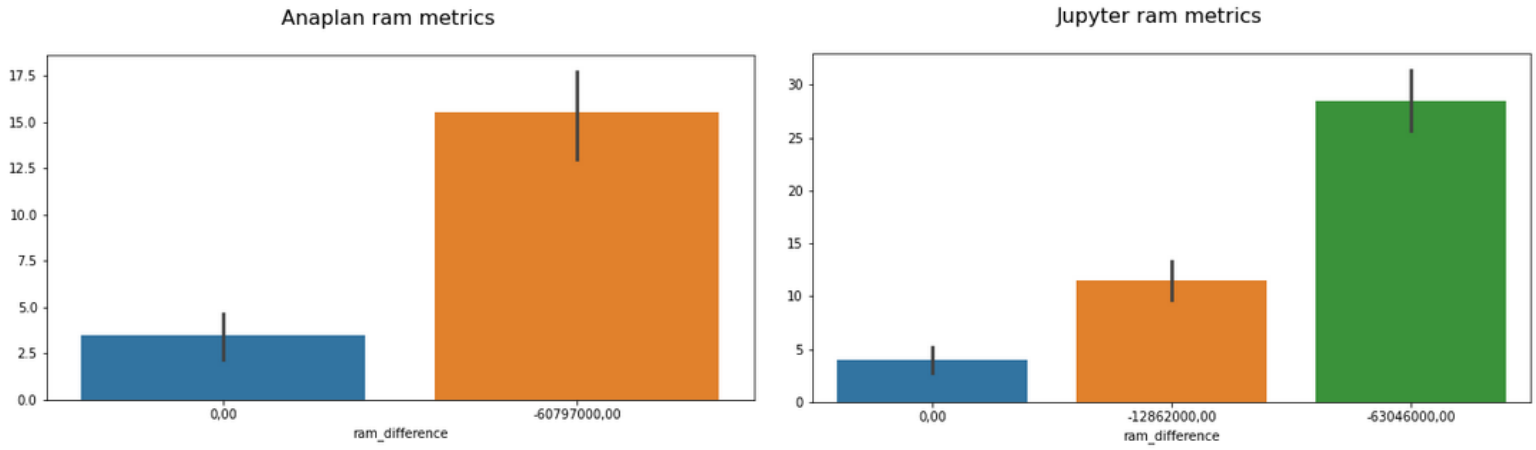


Рис. 6. Сравнение нагрузки на оперативную память в Anaplan и Jupyter Notebook

Обработка готовых результатов

Для обработки результатов проверим следующие гипотезы:

1. Гипотеза о нагрузке на процессор:

H0: Anaplan и Jupyter оказывают одинаковую нагрузку на процессор

H1: Anaplan и Jupyter оказывают разную нагрузку на процессор

2. Гипотеза о нагрузке на сеть:

H0: Anaplan и Jupyter оказывают одинаковую нагрузку на сеть

H1: Anaplan и Jupyter оказывают разную нагрузку на сеть

3. Гипотеза о нагрузке на оперативную память:

H0: Anaplan и Jupyter оказывают одинаковую нагрузку на оперативную память

H1: Anaplan и Jupyter оказывают разную нагрузку на оперативную память

Указав гипотезы, мы должны определить уровень значимости. Выберем уровень значимости 5 %. Наша выборка содержит 20 измерений. Такие маленькие выборки делают невалидным допущение о том, что стандартная ошибка нормально распределена. Поэтому для ее оценки использовался t-критерий.

Проверка гипотезы 1:

t_ test показал значение 0.406. Как писали ранее, порог уровня значимости - 0.05. Что говорит о том, что мы не можем отвергнуть данную гипотезу. Итак, мы доказали, что Anaplan и Jupyter оказывают одинаковую нагрузку на процессор.

Проверка гипотезы 2:

В ходе обработки результатов мы получили среднеквадратичную ошибку, при которой невозможно использовать t-тест . Большая среднеквадратичная говорит о том, что данные слишком далеко расположены друг о друга. В этом случае мы не можем проверить гипотезу. Однако заметим, что нагрузка Anaplan на сеть была постоянной и менялась с течением времени. Нагрузка на сеть Jupyter была значительно меньше и уменьшалась в ходе выполнения эксперимента.

Проверка гипотезы 3:

Аналогичную ситуацию со среднеквадратичной ошибкой мы получили при проверке 3 гипотезы. Обратившись к датасету, мы видим, что нагрузка на оперативную память Anaplan не меняется с течением времени, в то время как Jupyter нагружает процессор в разной степени.

Анализ времени выполнения

В ходе выполнения эксперимента было измерено время, за которое аналогичные действия производились Anaplan и Jupyter. Anaplan на обработку данных (импорт, пересчет и построение графиков) понадобилось 12 секунд, Jupyter - 20.

Выводы

В ходе исследовательской работы удалось получить следующие выводы:

1. Jupyter - универсальное средство, позволяющее обрабатывать и анализировать данные. В Anaplan данные должны приходить уже подготовленными.
2. Нагрузка на оперативную память Anaplan стабильна и не изменяется с течением времени, что объясняется облачными вычислениями. Jupyter же потребляем больше ресурсов при загрузке данных.
3. Нагрузка на сеть Anaplan также стабильна и выше, чем у Jupyter, что связано с использованием веб-приложения.
4. Anaplan и Jupyter оказывают одинаковую нагрузку на процессор
5. Anaplan выполняет анализ данных быстрее, что снова может быть связано с облачными вычислениями на более мощных серверах, чем использованный ПК.

Результаты исследования доказали, что Anaplan оказывает минимальную нагрузку на ПК, которая уменьшается с момента запуска процесса. Это не только доказывает, что платформа осуществляет облачные вычисления, но и

обосновывает ее востребованность. Сами создатели уверяют, что высокая производительность обусловлена использованием Polaris Calculation Engine™[7]. Кроме того, возможности платформы позволяют пользователям включать огромные объемы данных в свои варианты использования для решения сложных бизнес-сценариев и легко корректировать прогнозы.

Литература:

1. Monitoring as Code на базе VictoriaMetrics и Grafana. URL: https://habr.com/ru/post/568090/
2. A comprehensive guide to convolutional neural network. URL: https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53
3. Серия статей “Python и статистический вывод”. URL: https://habr.com/ru/amp/post/556852/
4. Документации к сервисам. URL:https://ndres.me/post/jupyter-notebook-rest-api/; https://jupyter-kernel-gateway.readthedocs.io/en/latest/getting-started.html https://help.anaplan.com/3a4a2905-3d55-4199-a980-d1a89ffdcb7e-Use-basic-authentication
5. Описание jupyter kernel gateway. URL:https://towardsdatascience.com/expose-endpoints-using-jupyter-kernel-gateway-e55951b0f5ad
6. Ю.Е. Гапанюк “Машинное обучение”. URL:https://github.com/ugapanyuk/ml_course_2022/wiki/COURSE_TMO

References

1. Monitoring as Code based on VictoriaMetrics and Grafana. URL:https://habr.com/ru/post/568090/

[7] Технология производства ядра, запатентованная Anaplan. https://www.anaplan.com/stories/introducing-next-generation-anaplan-polaris-calculation-engine/

2. A comprehensive guide to convolutional neural network. URL: https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53
3. “Python и статистический вывод”. URL: https://habr.com/ru/amp/post/556852/
4. Documentation. URL: https://ndres.me/post/jupyter-notebook-rest-api/;https://jupyter-kernel-gateway.readthedocs.io/en/latest/getting-started.html https://help.anaplan.com/3a4a2905-3d55-4199-a980-d1a89ffdcb7e-Use-basic-authentication
5. Description jupyter kernel gateway. URL: https://towardsdatascience.com/expose-endpoints-using-jupyter-kernel-gateway-e55951b0f5ad
6. U.E. Gapanyuk “Machine learning”. URL: https://github.com/ugapanyuk/ml_course_2022/wiki/COURSE_TMO

© Абдрашитова А.Н., Муханов Е., 2022 Научно-образовательный журнал для студентов и преподавателей «StudNet» №6/2022.

Для цитирования: Абдрашитова А.Н., Муханов Е. СРАВНИТЕЛЬНЫЙ АНАЛИЗ СИСТЕМ ОБРАБОТКИ ДАННЫХ JUPYTER NOTEBOOK И ANAPLAN// Научно-образовательный журнал для студентов и преподавателей «StudNet»6/2022