Оценка качества научных гипотез в виртуальных экспериментах в областях с интенсивным использованием данных

© Е.А. Тарасов[1] © Д.Ю. Ковалев[2]

[1]Московский государственный университет имени М.В. Ломоносова,
[2]Федеральный исследовательский центр «Информатика и управление» РАН,
Москва, Россия

tarasov@outlook.com dkovalev@ipiran.rue

Аннотация. Исследованы подходы, позволяющие оценить качество модели, реализующей гипотезу в рамках виртуального эксперимента. Математические модели, порождающие гипотезы, активно используются в областях с интенсивным использованием данных. К одной из таких областей можно отнести исследования многофазных потоков жидкости. Модель реализует гипотезу о скорости потока жидкости в трубе. Подход оценки качества осуществляется в рамках общей классической теории детектирования сигнала. Оценка представляет собой бинарный показатель. В качестве аппарата проверки гипотез используются два метода: частотный и Байесовский. Обработка входных данных осуществляется в потоковом режиме. Из данных, поступающих на вход системе, на этапе предобработки извлекаются признаки, которые затем подаются на вход исследуемой модели. С определенной периодичностью происходит перерасчет оценки качества с учетом изменяющихся параметров среды. Таким образом, отслеживается момент, когда модель начинает плохо предсказывать поведение физического явления. В работе описана реализация данного функционала на распределенной вычислительной системе.

Ключевые слова: виртуальный эксперимент, управление гипотезами, частотный подход, Байесовский подход, многофазное течение жидкости.

Estimation of Scientific Hypotheses Quality in Virtual Experiments in Data Intensive Domains

© Evgeny Tarasov[1] © Dmitry Kovalev[2]

[1] Lomonosov Moscow State University,
[2] Federal Research Center Computer Science and Control of the Russian Academy of Sciences,
Moscow, Russia

tarasov@outlook.com dkovalev@ipiran.rue

Abstract. In this paper, we investigate approaches that allow us to estimate the quality of model implementing hypotheses within a virtual experiment. One of the areas of DID under study is the multiphase fluid flow analyses. The quality estimation approach is carried out within the framework of the general classical detection theory. The estimate is a binary indicator. As a instrument for testing hypotheses, two approaches are used: frequency and Bayesian. Feature Extraction is carried out in the streaming mode. With a certain periodicity, the quality assessment is recomputed taking into account the changing environmental parameters. Thus, the moment when the model begins to poorly predict the behavior of the physical phenomenon is captured. This paper describes the implementation of this approach within a distributed computing framework.

Keywords: virtual experiment, hypothesis management, frequency approach, Bayesian approach, multiphase fluid flow.

1 Введение

Данные в современных исследованиях имеют определяющую роль [11]. Они могут быть представлены как в неструктурированном, так и в полуструктурированном виде. Таким образом, акцент в работе ученого смещается с проведения реального физического эксперимента на обработку данных в рамках виртуального эксперимента, моделирующего поведение физического явления. Данная парадигма работы получила название исследований с интенсивным использованием данных (ИИИД) [20].

Труды XIX Международной конференции «Аналитика и управление данными в областях с интенсивным использованием данных» (DAMDID/ RCDL'2017), Москва, Россия, 10–13 октября 2017 года

К одной из областей ИИИД можно отнести течение многофазных потоков жидкости [8]. Данные собираются с множества сенсоров, например, выполняется дискретная запись акустического сигнала, температур, давления, течения жидкости. Накопленные объемы данных служат для характеристики многофазных потоков и их режимов. Важным приложением интерпретации акустического сигнала и накопленных мета-данных является предсказание скорости потока жидкости.

Несмотря на значительные успехи в интерпретации данных с сенсоров, проблемы построение сложных моделей, объясняющих динамику течения жидкости, остаются открытыми. Проведение точного физического эксперимента требует от исследователя серьезных усилий, т. к. необходимо обеспечить множество специальных условий [32]. Численное моделирование, особенно для недостаточно хорошо изученных потоков, часто требует калибровки с экспериментом [25, 31]. Разработка подходов к анализу данных для интерпретации потоковых данных виртуального эксперимента является важной и перспективной проблемой для исследования.

Параметры модели течения жидкости не являются статичным элементом. Внешние условия изменчивы и влияют на неё. Из-за этого предсказательная способность модели может начинать резко ухудшаться [30]. Это приводит к необходимости проведения её повторной калибровки. Важно отследить тот момент, когда модель начинает выдавать заведомо плохой результат. Таким образом, платформа, обрабатывающая вычисления, должна уметь работать с потоковыми данными в режиме, близком к реальному времени. Это обуславливает особое отношение к такому роду задач.

Одной из таких особенностей является требование распределенности. Система должна быть легко масштабируема, чтобы обрабатывать модель любой сложности с постоянно возрастающим объемом данных за разумный промежуток времени. Готовых открытых систем в области гидродинамики, удовлетворяющих данному требованию, по сведению авторов, нет.

Всё больше задач формируется не в рамках одной области, а междисциплинарно. Таким образом, возрастает роль онтологических спецификаций [37]. Это позволяет как различным ученым в рамках одной области, так и ученым из различных областей использовать общие понятия, что необходимо для ускорения проведения совместных исследований. В данной работе область применения исследования лежит как в контексте изучения физических явлений, а именно, течения жидкости, так и управления гипотезами [10].

Новизной данной работы является распределенная реализация метода бинарной оценки качества модели, работающей с потоковыми данными.

Статья организована следующим образом. В разделе 2 представлен сравнительный обзор платформ, на базе которых может быть осуществлен виртуальный эксперимент. Раздел 3 определяет онтологическую спецификацию двух предметных областей: течения жидкости и управления гипотезами. В разделе 4 выполнена концептуальная спецификация данных, используемых в виртуальном эксперименте. Раздел 5 описывает формат сырых входных данных, данных, поступающих на вход модели, и извлечённые значимые признаки модели течения жидкости. Раздел 6 содержит описание метода оценки качества с использованием двух подходов: частотного и Байесовского. Раздел 7 раскрывает архитектуру использованного для расчётов программно-аппаратного комплекса. В разделе 8 представлен реализованный поток работ. Раздел 9 описывает результаты, полученные на практике.

2 Обзор платформ

2.1 Критерии выбора

Одним из ключевых элементов ИИИД, наряду с машинным обучением, является явное использование гипотез в определении виртуального эксперимента [10]. Многие исследователи скептически относятся к подходу с использованием машинного обучения, так как он дает низкую интерпретируемость полученных результатов, так как многие методы в нем используются как черный ящик [18]. Подход же на основе гипотез лишен данного недостатка. Гипотезы в математическом виде описывают априорные знания об исследуемом явлении, которые проверяются в рамках виртуального эксперимента.

На сегодняшний день отсутствует единая методология работы с потоковыми данными [36]. Программные продукты, которые в той или иной мере реализуют представления отдельных групп ученых на то, как нужно работать с ними, не являются развитыми и стабильными.

В рамках исследования существующих решений по поставленной перед нами цели рассмотрим платформы для обработки данных в таких области научной деятельности как:

- средства управления гипотезами и проведения виртуального эксперимента;
- средства управления и обработки потоковых данных.

В качестве требований, предъявляемым при сравнительном анализе систем, будем применять следующие положения:

1. Система должна соответствовать онтологической спецификации предметной области. Это значит, что на базе нее возможно:
   a. реализовать модель;
   b. провести статистическое тестирование;
2. Система должна уметь работать с большим объемом потоковых данных;
3. Распределенность и скорость, т. е. должна легко масштабироваться в зависимости от вычислительной нагрузки. Результат должен получаться в режиме, близком к реальному времени;

Таблица 1 Сравнение платформ управления гипотезами

Название	Ключевые элементы	Слабые стороны	Сильные стороны
Hephaestus	Собственный SQL-подобный язык запросов для описания эксперимента. Основной элемент – виртуальный эксперимент. Построение вероятностно-причинных графов. Мета-система – работает над существующими базами данных.	Интеграция данных не предоставляется из коробки. Плохое описание модуля поиска корреляций. Ориентированность применения – здравоохранение, ведет к своей интерпретации определения виртуального эксперимента.	Тестирование и ранжирование гипотез на основе частотной статистики. Граф знаний может визуализировать результаты и помогать интегрировать новые гипотезы. Работает с наборами гипотез.
FCCE	Использование NoSQL БД. Основной элемент – концепция функций. API поддержки для хранения, извлечения, оценки корреляции по признакам. Комплексная многоуровневая система агрегации данных.	Не описан модуль корреляций. Ориентированность применения – анализ поведения сети. Не оперирует математическими формулами.	Сосредоточение на минимизации задержек. Поддержка доступа к сырым данным. Быстрый модуль поиска корреляций. Программно реализован.
ϒ-DB	Основной элемент – поддержка научных исследований. Вероятностная БД. Работа с отдельными гипотезами. Байесовский подход. Гипотезы в формате MathML.	Взаимосвязь гипотез выходит за рамки системы. Проблемы с масштабируемостью.	СУБД на базе SQL. Автоматически пересчитывает вероятность после получения новых данных или гипотезы. Работает с формулами.

4. Открытость, т. е. система должны быть с открытым исходным кодом;
5. Стабильность работы также является важным критерием, т. к. многие средства были разработаны ещё совсем недавно и не прошли полного цикла отладки.

2.2 Управление гипотезами

В настоящее время в рамках работы в областях с интенсивным использованием данных многие исследователи приходят к выводу о необходимости унификации подходов построения виртуальных экспериментов. Отдельные научные группы разрабатывают свои программные продукты, реализующие видение своих авторов к данной проблематике. Проанализируем некоторые из них – такие продукты, как: Hephaestus [10, 38], Features Collection and Correlation Engine (FCCE) [26, 38], ϒ-DB [11, 12, 38]. Сводная информация по сравнению платформ управления гипотезами представлена в Таблице 1.

2.3 Потоковые системы

Для сравнения выберем открытые системы, которые являются наиболее популярными с точки зрения применимости в практических задачах. К таким продуктам можно отнести: Storm [5, 21, 15], Flink [1, 2, 15], Spark Streaming [4, 15, 27, 13].

Серьёзной проблемой выбора потоковых фреймворков [36, 29] является отсутствие в настоящее время единых и объективных критериев оценки производительности [36]. В научной литературе существуют публикации, авторы которых проводят определённые сравнения, однако проблемой являются узкая специализация и ограниченность применения этих тестов [9]. В связи с отсутствием единой методологии целесообразно получать сравнительную производительность систем на данных конкретной исследовательской задачи для всех анализируемых платформ.

Однако скорость является не единственным критерием выбора платформы. Сводные данные [13, 15] по потоковым системам представлены в Таблице 2.

2.4 Выбор платформы

Исходя из всех выше приведенных обзоров, можно сделать следующие выводы.

В рамках данной работы невозможно использовать никакую из существующих систем по управлению гипотезами. Это обусловлено тем, что они:

- ориентированы на работу с статическими данными, хранящимися в базе данных;
- нет гибкого инструмента построение виртуального эксперимента; имеется ориентаций на свои области применения;
- не являются открытыми программными продуктами;
- не носят законченного характера; некоторые модули имеют только описательный характер без практической реализации;
- плохая документированность.

Таблица 2 Сравнение потоковых систем

Критерий	**Storm**	**Flink**	**Spark Streaming**
Поддержка языков программирования	Java	Java, Scala	Java, Scala, Python
Режим работы	Потоковый (tuple-wise)	Потоковый и микро-пакеты	Микро-пакеты (micro-batch)
Обработка сообщений	По крайней мере один раз (at least once)	Строго один раз (exactly once)	Строго один раз (exactly once)
Управление окном	Нет, встроенными средствами	На основе: времени, строк, приходящих данных	Только на основе времени
Управление ресурсами	YARN, Mesos, Built-in	YARN, Built-in	YARN, Mesos, Built-in
Задержка обработки	Низкая	Низкая	Средняя
Механизмы обеспечения отказоустойчивости	ACK записи	Распределенные снапшоты	Микро-пакет
Управление потоком	Проблематично	Естественно	Проблематично
Операции с сохранением состояния	Нет	Да	Да
Потоковая примитива	Tuple	DataStream	DStream
Пропускная способность	Низкая	Высокая	Высокая

В качестве платформы оценки качества виртуального эксперимента выбрана платформа на базе Spark Streaming. Этот выбор обусловлен следующими положениями:

- текущая реализация модели течения жидкости представлена на Python;
- имеется возможность расширения функционала программирования проверки статистических гипотез за счет установки дополнительных библиотек;
- возможность управления окном на основе времени.

Из сравнительной таблицы потоковых систем видно, что Spark Streaming не является самой производительной системой. Её выбор обусловлен поддержкой языка Python и обеспечением минимально необходимого функционала.

3 Онтологическая спецификация

3.1 Течение жидкости

Онтологическая спецификация течения жидкости представлена на Рис. 1. Данная онтология описывает проведение реального эксперимента, при котором анализируется течение жидкости в трубе [8]. В качестве жидкости могут использоваться: вода, масло, их смесь с газом. Жидкость в рамках данного эксперимента обладает следующими свойствами: температура, скорость потока, давление. В рамках модели акустический шум и спектр являются функциями давления. Спектр в свою очередь характеризуется амплитудой и частотой, которые являются входными данными для определения величины частотного пика. Давление измеряется гидрофонами, установленными в гидродинамической трубе. Также для измерения температуры скорости потока используются дополнительные сенсоры. Данные с гидрофонов и сенсоров поступают на аналогово-цифровой преобразователь, который, в свою очередь, усиливает и дискретизирует поступающий сигнал в соответствии с заданными свойствами. В эксперименте частота дискретизации составляет 100 кГц. Обработанный сигнал записывается в результирующий файл эксперимента.

Онтология предметной области соответствует реальному физическому эксперименту. Каждый эксперимент выполняется в течение 25 секунд. Затем происходит некоторое изменение в параметрах модели, затем эксперимент повторяется снова. Так на выходе получается набор результирующих файлов. Поток данных формируется искусственно с использованием этих файлов.

3.2 Управление гипотезами

Онтологическая спецификация управления гипотезами представлена на Рис. 2.

Определим некоторые термины предметной области. Виртуальный эксперимент – деятельность по применению набора гипотез для воспроизведения симуляций близких к наблюдаемому явлению. Гипотеза – формальная спецификация свойств исследуемого объекта или явления, имеющих математическое представление. Модель – алгоритм, реализующий гипотезы. Оценка качества – характеристика модели, позволяющая сделать вывод о её соответствии реальному явлению.

Статистическое тестирование – это процесс принятия решения о том, противоречит ли рассматриваемая статистическая гипотеза выборке данных. Частотный подход – аппарат проверки гипотез, базирующийся на частотном определении вероятности, т. е. вероятности как предела относительной частоты наблюдения некоторого события в серии однородных независимых испытаний. Байесовский подход – аппарат проверки гипотез, базирующийся на байесовском определении вероятности, для которой имеются некоторые априорные знания о наблюдаемом явлении. Потоковая система – программное средство, позволяющее анализировать непрерывно поступающие на ее вход данные.

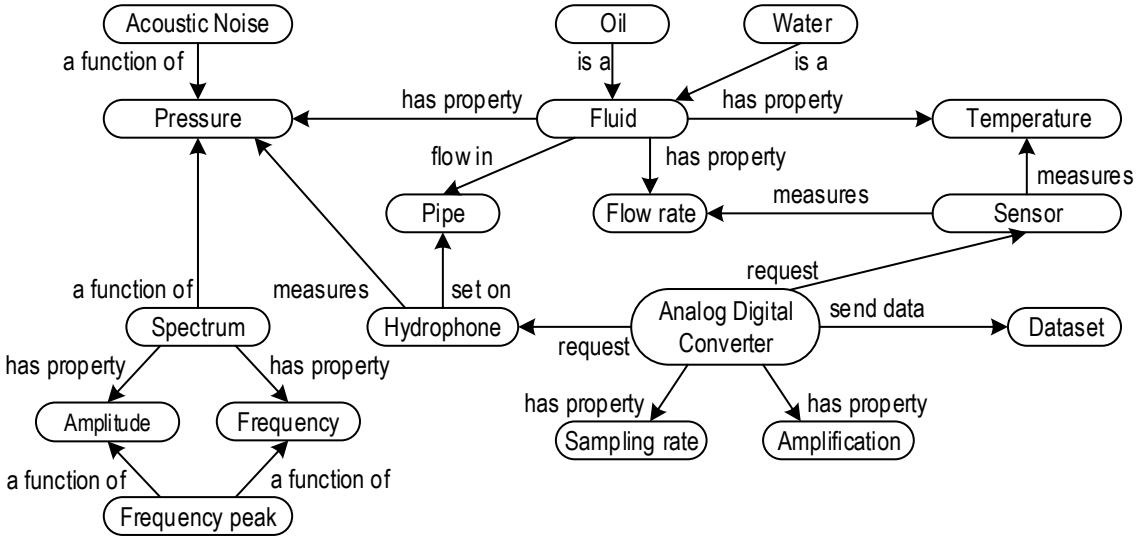

Рисунок 1 Онтология течения жидкости

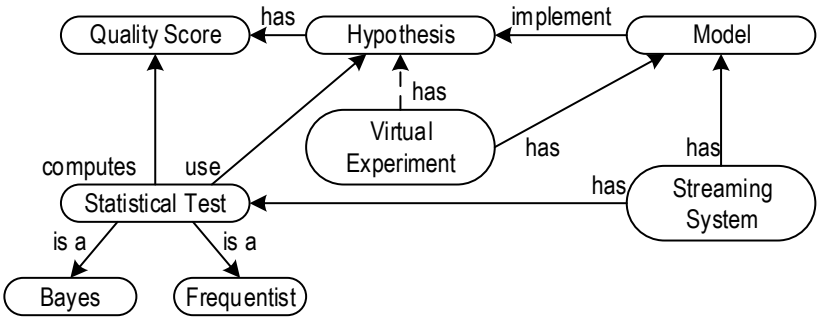

Рисунок 2 Онтология управления гипотезами

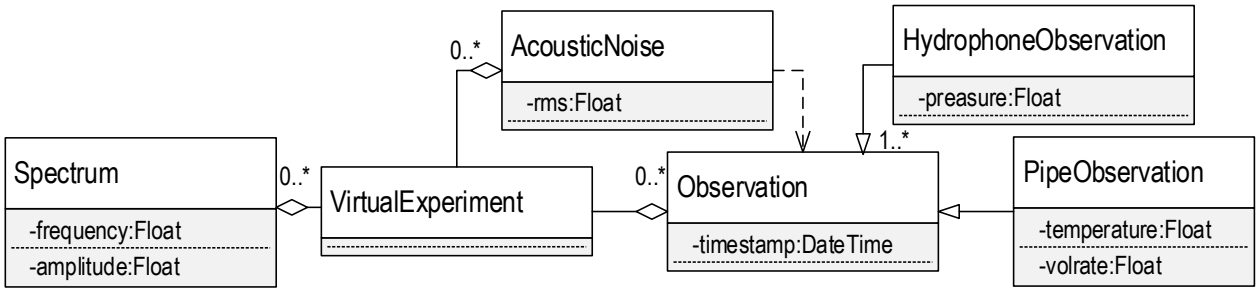

Рисунок 3 Концептуальная спецификация

На базе потоковой системы реализована модель, описывающая течение жидкости в трубе. Также данная система имеет модули выполнения проверки статистических тестов. На базе результата работы модели и входных данных сформулированы гипотезы, позволяющие получить оценку качества модели. Оценка высчитывается на базе статистических тестов, заложенных в потоковой системе. В виртуальном эксперименте рассматриваются частотный и Байесовский подходы для получения оценки качества.

4 Концептуальная спецификация

Концептуальная спецификация виртуального эксперимента представлена на Рис. 3.

В рамках исследуемой модели имеется априорное знание о зависимости скорости потока от температуры, значении частоты пика спектра, шума потока. Также известно, что имеются корреляции между показателями гидрофонов, установленных в гидродинамической трубе. Данные разделяются на полученные из наблюдения и вычисленные из данных наблюдения. К непосредственно получаемым данным относится информация от гидрофонов и сенсоров, а именно: давление, температура, скорость потока. К вычисляемым относятся: спектр, амплитуда, акустический шум. Все эти характеристики рассчитываются на основе наблюдаемых показателей давления. Все данные, кроме временной метки, имеют вещественный тип.

5 Формат данных

5.1 Входные данные

Данные, полученные в рамках реального

физического эксперимента, хранятся в CSV (Comma Separated Value) файлах на распределенной файловой системе (HDFS). На вход системе поступает два вида файлов: данные, полученные с гидрофонов; данные, полученные с сенсоров.

Файл с информацией, полученной с гидрофонов, имеет следующее описание. Заголовок файла состоит из порядкового номера *nm*,– указатель канала гидрофона $h\{i\}$, где i – число от 0 до 2. Столбцы разделены знаком табуляции. Заголовок файла выглядит так: `nm\t\h0\h1\h2`.

Частота дискретизации для сбора акустических данных составляет 100 кГц. Один эксперимент длится 25 секунд. Таким образом, каждый CSV файл содержит 2,5 миллиона строк.

Файл с информацией, полученной с сенсоров, имеет следующее описание: заголовок файла состоит из timestamp – временной метки, temp – температуры, vol_rate – скорости потока, file – файла, соответствующего показателям гидрофонов. Столбцы разделены знаком ";". Заголовок файла выглядит так: `timestamp;temp;vol_rate;file`

Показатели температуры и скорости потока собираются один раз в секунду. Вопросы синхронизации показателей различных файлов решаются через информацию временных меток и сопоставления имен файлов в соответствующих полях.

5.2 Обрабатываемые данные

Данные для обработки в потоковой системе преобразуются в формат RDD (Resilient Distributed Dataset) – отказоустойчивый набор элементов, обработка которых может выполняться параллельно [28]. Вся логика работы с данными происходит в рамках этого концепта. Существуют два способа получения такого набора:

- параллелизация последовательного массива в рамках текущей программы с помощью вызова метода parallelize();
- на этапе извлечения данных из внешних источников, таких, как HDFS, HBase, Streaming Context.

При создании распределенного набора можно как явно задать число, показывающее, сколько параллельных разделов использовать при работе с этими данными, так и использовать значение по умолчанию.

В рамках виртуального эксперимента самые «тяжелые» в вычислительном плане задачи ложатся на набор данных, поступивших с гидрофонов. Так как используется информация из трёх каналов, то степень параллелизма на этапе предобработки устанавливается также равной трём.

Концепт RDD имеет свой API и поддерживает два вида операций:

- трансформацию – получение нового набора из существующего;
- действие – запускает задание на выполнение.

При написании программы важно отслеживать область действия переменных в рамках данных видов операций.

Важными характеристиками RDD являются:

- распределенность – операции над данными выполняются на различных узлах кластера;
- ленивое исполнение ("lazy") – трансформация не выполняется прямо сейчас; система хранит последовательность операций над набором; выполнение происходит, только если в коде программы встретилась операция действия;
- управление состоянием – возможность выбора, из какого хранилища (память или диск) они будут повторно использоваться.

6 Подход оценки качества

6.1 Описание метода

Рассматриваемый нами метод оценки качества виртуального эксперимента опирается на классическую теорию детектирования сигнала [14]. Эта теория выступает как средство количественной оценки возможности различать информационную составляющую сигнала от шума [24]. Базовые компоненты теории детектирования представлены на Рис. 4.

Первым ее элементом является источник, который генерируют выходной сигнал. Его выход может быть одним из нескольких вариантов. В самом простом случае это гипотезы: H_1 и H_0. Второй и третий компоненты соответственно: механизм вероятностного перехода и пространство наблюдений. Механизм перехода может рассматриваться для определения, какая гипотеза истина. На основе этих знаний он генерирует точку в пространстве наблюдений в соответствии с некоторым законом вероятности. Независимая дискретная случайная величина n, чья плотность вероятности нам известна, добавляется к выходу источника.

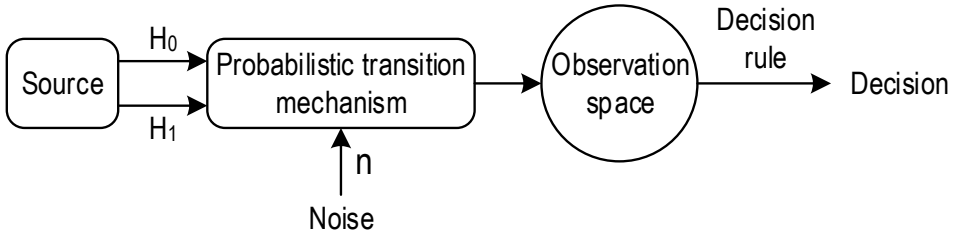

Рисунок 4 Компоненты теории детектирования

Четвертым компонентом теории детектирования сигнала является правило принятия решения. После получения наблюдаемой переменой в пространстве наблюдений мы должны угадать какая гипотеза была истинной. Правило принятия решений сопоставляет каждой точке наблюдений одну из гипотезПодходящий выбор правил будет зависеть от многих факторов, которые определяются постановкой конкретной исследовательской задачи.

Мы рассматриваем оценку качества как бинарное событие, которое в математическом представлении определено в виде гипотез:

H_0: модель корректна;

H_1: происходит нарушение работы модели.

Чтобы сделать предположение о качестве модели, необходимо выполнить статистическую проверку данных.

Одним из параметров, передаваемых на вход нашей системы один раз в секунду, является измеренная величина Y скорости потока. В рамках проведения виртуального эксперимента модель выдавала вычисленную величину $\hat{Y}$ скорости потока. Оценивая их разность $\hat{Y} - Y$ на протяжении 25 секунд, можно сделать вывод об оценке качества модели при заданных условиях эксперимента.

Известно, что разность $\hat{Y} - Y$ подчиняется нормальному закону распределения [8]. Таким образом, в качестве аппарата проверки гипотез могут быть использованы следующие подходы: классический или частотный метод; Байесовский метод.

6.2 Частотный подход

В рамках классического подхода вероятность определяется как относительная частота наступления события. Все события имеют независимый характер. Общая методика проведения статистического тестирования широко представлена в литературе (см., например, [35]). В рамках настоящей работы решалась задача в следующей постановке.

В исследуемой модели течения жидкости используем следующие положения:

- остатки подчиняются нормальному закону распределения;
- дисперсия сигнала неизвестна.

Таким образом, в рамках частотного подходы будет использован одно-выборочный Т-тест с двухсторонней альтернативой.

В качестве исследуемой случайной величины возьмем разность $X_n = \hat{Y} - Y$ контрольного значения скорости потока, поступающего с датчиков, и величины скорости потока, полученной в результате работы модели. Таким образом, получим выборку

$$X = (x_1, \dots, x_n) \in R, \quad X_n \sim N(\mu, \sigma^2).$$

Проверим гипотезу, что выборочное среднее равно заданному числу, против альтернативной гипотезы, что это не так: $H_0: \bar{X} = \mu$, $H_1: \bar{X} \neq \mu$. В нашем случае примем $\mu = 0$. В качестве правила принятия решения возьмем критерий Стьюдента [34]. Статистика критерия имеет распределение Стьюдента с $n-1$ степенями свободы:

$$T(X) = \frac{(\bar{X} - \mu)\sqrt{n}}{S} \sim st(n-1),$$

где выборочное среднее $\bar{X} = \frac{1}{n}\sum_{i=1}^{n} x_i$, выборочная дисперсия $S^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{X})^2$.

Пусть задан уровень значимости $\alpha = 0.05$. В нашей задаче используем двухстороннюю альтернативу. Таким образом, статистический критерий будет иметь вид $|T| > T_{\alpha/2}$, где T_α – α-квантиль распределения Стьюдента с $n-1$ степенями свободы. Если $|T| > T_{\alpha/2}$, то нулевая гипотеза H_0 отвергается.

6.3 Байесовский подход

Байесовский подход отличается от классического тем, что в своей основе имеет другое определение вероятности (она интерпретируется как мера незнания, а не как объективная случайность [33]). В общем виде вероятность определяется как степень уверенности в истинности суждения. Мы имеем некоторое априорное знание о наблюдении, которое уточняется в процессе эксперимента.

Байесовская проверка в рамках теории детектирования базируется на двух предположениях:

- выходы источника регулируются вероятностным присвоением, они обозначаются P_1 и P_0 и называются априорными вероятностями, которые представляют собой информацию об источнике до проведения эксперимента;
- каждому из возможных исходов присваивается стоимость; обозначим стоимости 4-х исходов C_{00}, C_{10}, C_{01}, C_{11}; первый индекс указывает на выбранную гипотезу, второй – на ту, которая истинна; после каждого эксперимента стоимости могут уточняться.
- Правило принятия решения должно быть таким, чтобы средняя стоимость была как можно меньше. В общем виде данный подход характеризуется формулой [14]:

$$\frac{p_{r|H_1}(R|H_1)}{p_{r|H_0}(R|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \frac{P_0(C_{10}-C_{00})}{P_1(C_{01}-C_{11})}.$$

Величина в левой части называется коэффициентом правдоподобия и обозначается $\Lambda(R)$. Величина в правой части формулы является пороговым значением теста и обозначается η.

Таким образом, Байесовский критерий приводит нас к проверке неравенств

$$\Lambda(R) \underset{H_0}{\overset{H_1}{\gtrless}} \eta \quad \text{или} \quad \ln\Lambda(R) \underset{H_0}{\overset{H_1}{\gtrless}} \ln\eta.$$

Эмпирическая шкала [20] доказательной силы Байесовского критерия приведена в Таблице 3.

Таблица 3 Шкала критерия правдоподобия

$\lvert \ln \eta \rvert$	η	Доказательная сила
< 1.0	< 3:1	Не убедительная
1.0	~ 3:1	Слабое доказательство
2.5	~ 12:1	Среднее доказательство
5.0	~ 150:1	Сильное доказательство

Доказательная сила говорит о том, можем мы или нет отвергнуть гипотезу H_0. Так как гипотеза H_0 предполагает, что наша модель корректна, то ее отвержение служит оценкой качества и указывает на то, что модель начинает плохо описывать наблюдения.

Дадим постановку задачи. В качестве исследуемой случайной величины, как и в частотном подходе, используем $X_n = \hat{Y} - Y$ (разность контрольного значения скорости потока, поступающего с датчиков, и величины скорости потока, полученной в результате работы модели). Таким образом, задана выборка

$$X = (x_1, \dots, x_n) \in R, X_n \sim N(\mu, \sigma^2).$$

Проверим гипотезу о том, что разность средних наблюдаемого и моделируемого сигналов равна нулю плюс имеется дополнительная составляющая некоторого шума, против альтернативной гипотезы о том, что кроме шума имеется ненулевая составляющая, информирующая, что сигнал ушел с базовой линии:

$$H_0: r_i = 0 + n_i = n_i, \ \ i = 1,2, \dots, N,$$
$$H_1: r_i = \mu + n_i, i = 1,2, \dots, N.$$

Шумовая составляющая подчиняется нормальному закону распределения. Таким образом,

$$p_i(X) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{X^2}{2\sigma^2}\right).$$

Плотность вероятностей r_i при каждой гипотезе вычисляется следующим образом:

$$p_{r_i|H_1}(R_i|H_1) = p_{n_i}(R_i - \mu) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(R_i-\mu)^2}{2\sigma^2}\right),$$

$$p_{r_i|H_0}(R_i|H_0) = p_{n_i}(R_i) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{{R_i}^2}{2\sigma^2}\right).$$

Поскольку n_i статистически независимы, совместная плотность вероятностей r_i является простым произведением индивидуальных плотностей вероятности. Таким образом, приведенные формулы можно записать в следующем виде:

$$p_{r|H_1}(R|H_1) = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(R_i-\mu)^2}{2\sigma^2}\right),$$

$$p_{r|H_0}(R|H_0) = \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{{R_i}^2}{2\sigma^2}\right).$$

Подставив полученные результаты в формулу коэффициента правдоподобия, получим

$$\Lambda(R) = \frac{\prod_{i=1}^{N} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(R_i-\mu)^2}{2\sigma^2}\right)}{\prod_{i=1}^{N} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{{R_i}^2}{2\sigma^2}\right)}.$$

Сравнив полученное значение коэффициента правдоподобия с эмпирической шкалой, сделаем вывод о качестве модели в рамках виртуального эксперимента.

7 Описание архитектуры

Для проведения вычислительного эксперимента в рамках поставленных задач был собран программно-аппаратный вычислительный комплекс: аппаратная часть построена на базе серверов и коммутатора фирмы Quanta [23]; программная часть – программный кластер на базе продукта HDP (Hortonworks Data Platform) [16].

7.1 Аппаратная часть

Аппаратная архитектура состоит из шести серверов, представленных в форм-факторе OpenRack и объединенных одним коммутатором по интерфейсу 10Gb Ethernet. Каждый сервер имеет свои локальный SSD-диск с установленными на нём операционной системой и программными компонентами HDP кластера, а также свою полку с HDD-дисками, подключенную напрямую по интерфейсу SATA. Каждый сервер выполняет определенную роль в кластере, которая определяет его технические характеристики. Технические характеристики управляющих и рабочих узлов представлены в Таблице 4.

Таблица 4 Характеристики узлов

Характеристика	Значение
Тип процессора	Genuine Intel 2.30GHz/ Intel Xeon E5-2630L
Количество ядер	40/24
Память	32/64ГБ
Тип дисков	SSD, HDD
Объем дисков	240ГБ, 2/4ТБ
Резервирование дисков	RAID-1/JBOD
Подключение дисков	SATA
Операционная система	CentOS 6.9
Сетевые интерфейсы	10GbEthernet

Роли серверов в кластере:

- m1, m2 – управляющие узлы, к ним предъявляются повышенные требования в плане производительности и надежности; на них установлены компоненты, отвечающие за распределение задач по рабочим узлам кластера, такие, как Name Node сервиса HDFS, ресурс менеджер YARN и др.; критические компоненты зарезервированы в режиме Active-Passive; в качестве Active сервера для большинства из них выступает узел m1;
- s1, s2, s3, s4 – рабочие узлы; основной вычислительный элемент кластера обеспечивает выполнение программного кода приложений в распределенной среде; выход из строя такого узла не является критическим, так как их состояние постоянно отслеживается управляющими узлами, которые в случае падения перезапустят задачу на оставшихся доступных серверах, однако это приведет к замедлению выполнения расчетов на кластере.

Доступ к компонентам управления платформы осуществляется из общей сети лаборатории за счет

подключения коммутатора кластера с общим маршрутизатором по интерфейсу 1GbEthernet.

7.2 Программная часть

Для построение вычислительной системы была использована платформа HDP версии 2.6. Эта версия является последней актуальной на момент написания этой статьи. На аппаратные сервера был установлен минимально необходимый набор программных компонент для проведения виртуального эксперимента:

- HDFS [7] – отказоустойчивая распределенная файловая система;
- YARN [22] – менеджер ресурсов, основной планировщик задач, запускаемых на кластере;
- Storm [5], Spark (с модулем Streaming) [4], Flink [2] – системы потоковой обработки данных;
- Kafka [3] – виртуальная очередь;
- Ambari [16] – веб-интерфейс для управления кластером;
- ZooKeeper [17] – сервер координации работы распределенных приложений;
- Zeppelin [16] – среда написания и отладки программного кода.

Все представленные ниже компоненты, кроме Apache Flink, входят в установочный пакет HDP кластера. Flink установлен дополнительно как сервис над YARN. Установка одновременно трёх потоковых систем в рамках одного кластера обусловлена тем, что в настоящий момент нет единой методологии оценки. Поэтому типовым является сценарий, когда существующая практическая задача в тестовом виде реализуется одновременно на всех платформах, а уже затем сравнивается производительность, полученная на практике. Также на момент развертывания кластера были неизвестны ограничения всех систем, и соответственно не было принято решение о применимости конкретного продукта для реализации виртуального эксперимента.

8 Поток работы

Поток работ виртуального эксперимента представлен на Рис. 5.

Виртуальный эксперимент заключается в одновременной непрерывной работе следующих компонент над потоком входных данными:

- Producer – программа, занимающаяся извлечением данных из csv-файла и отправляющая сообщения в очередь Kafka;
- Kafka – обслуживает прием, промежуточное хранение, репликацию данных;
- Spark Streaming – выполняет роль Consumer, извлекает данные из очереди в формате RDD и передает их на обработку в ядро Spark.
- Spark – выполняет вычисления с данными, вызывает модель течения жидкости, проводит статистические тесты оценки качества модели.

Этапы потока работ таковы:

1. Данные, полученные в результате физического эксперимента, хранятся в распределенной файловой системе. Producer представляет собой программный модуль, реализованный на Python. Этот компонент первым шагом извлекает данные из CSV-файлов, содержащих показатели гидрофонов, значения температуры и скорости потока и формирует два массива строк;
2. Подготовка данных к отправке заключается в разбиении полученных массивов строк на партии по 100 тыс. значений для показателей гидрофонов и одного – для температуры и скорости потока;
3. Отправка сообщений осуществляется с помощью вызова метода send() из загруженной библиотеки kafka-python [19]. Этот метод является ассиметричным. Для увеличения скорости отправки значение Ack выставлено в 0, чтобы Producer не ждал от Kafka-пакета подтверждения доставки. Так как кластер находится в изолированном сетевом сегменте и на серверах используются высокоскоростные сетевые интерфейсы, то потерь пакетов не происходит;
4. Из-за ограничений Spark Streaming как потоковой системы [6], а именно:
    a) имеется только временное управление окном;
    b) реализованы чтение данных из всех разделов очереди и запись их в одну RDD;
    c) недостаточна производительность работы ядра Spark из-за использования промежуточного слоя Python-интерпретатора и самой архитектуры Spark;
    d) не реализован функционал Backpressure стандартными средствами;

    возникла необходимость в оптимизации гиперпараметров модели. Изначально предполагалось подавать на вход модуля обработки данных из очереди 100 тыс. сообщений в секунду. За данный интервал времени Spark должен их успевать обрабатывать. Однако, таких скоростей: как обработки, так и подачи в очередь достигнуть не удалось. Поэтому возникла необходимость в увеличении временного интервала отправки партии сообщений;
5. Слияние полученных признаков в один кортеж для отправки его на вход модели;
6. Расчет модели на базе полученных входных данных. Распределение задания расчета модели по кластеру выполняется ядром Spark, исходя из своей внутренней логики работы;
7. Проведение статистического теста. Для оценки качества модели мы используем одно-выборочный Т-тест с двух сторонней альтернативой или Байесовский подход. На

выходе мы имеем строку с результатом в формате:

a. число – для оценки в случае байесовского критерия;
b. True/False – для оценки статистического критерия;

8. Вывод результата в консоль.

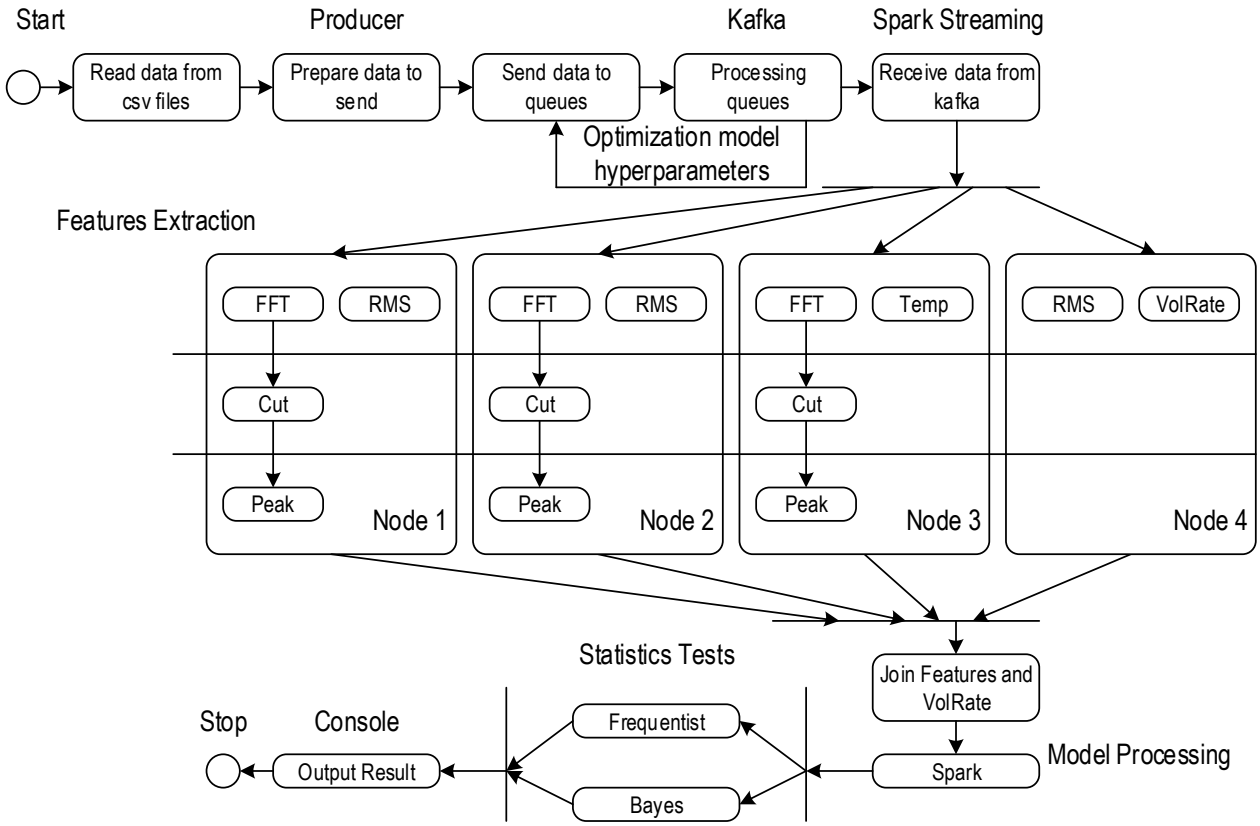

Рисунок 5 Поток работ

9 Полученный результат

По результатам проведенного эксперимента получены данные, представленные в Таблице 6. Цветом подсвечены результаты оценки качества виртуального эксперимента различными методами. Использованы следующие обозначения:

- белый – модель корректно описывает входные данные (Гипотеза H_0);
- серый – модель не корректна (Гипотеза H_1).

Таблица 6 Полученный результат

Байесовский	**Частотный**
0,0002	False
0,0296	False
0,6713	False
0,0053	False
0,0030	False
0,0006	False
1,1213	False
3,3046	True
5,4079	True
9,1870	True
20,0469	True
24,3965	True

По полученным результатам можно определить момент, когда модель начинает некачественно описывать поведение течения жидкости по входным данным. Результаты обоих методов схожи, но всё же немного отличаются, так как мы попали в граничные области Баейсовского критерия (3,3046), которые имеют слабую доказательную силу, поэтому отвергнуть гипотезу H_0 о том, что модель корректна, нельзя.

10 Заключение

Представлен подход, позволяющий оценить качество научных гипотез на примере области течения жидкости. Его идея базируется на классической теории детектирования, в рамках которой в качестве правил принятия решения выступают критерий Стьюдента и критерий правдоподобия.

Разработаны онтология предметной области и концептуальная схема виртуального эксперимента для исследования характеристик течения жидкости в трубе на основе его акустического шума. Произведен анализ существующих инструментов организации распределенной обработки потоковых данных.

Выполнено исследование подходов к организации методов проверки гипотез и оценки

качества моделей, реализующих соответствующие гипотезы.

Создана архитектура распределенной системы для оценки качества модели и проверки гипотез. Разработан масштабируемый программный модуль для существующей кластерной инфраструктуры.

Результаты эксперимента показали, что различные подходы проверки научных гипотез по оценке качества модели позволяют получить схожие результаты, незначительно отличающиеся на граничных областях критерия качества. Таким образом, частотный и Байесовский методы могут в равной степени быть применены для оценки качества виртуального эксперимента в рассматриваемой предметной области – течения жидкости.

Поддержка

Работа выполнена при поддержке РФФИ (грант 16-07-01028).

Литература

[1] Alexandrov, A., Bergmann, R., Ewen, S., Freytag, J., Hueske, F., Heise, A., Kao, O., Leich, M., Leser, U., Markl, V., Naumann, F., Peters, M., Rheinländer, A., Sax, M., Schelter, S., Höger, M., Tzoumas, K., Warneke, D.: The Stratosphere Platform aor Big Data Analytics. The VLDB J., 23 (6), pp. 939-964 (2014)

[2] Apache Flink: Scalable Batch and Stream Data Processing. https://flink.apache.org/

[3] Apache Kafka is a Distributed Streaming Platform. https://kafka.apache.org/intro

[4] Apache Spark – Lightning-Fast Cluster Computing. https://spark.apache.org/

[5] Apache Storm. https://storm.apache.org/

[6] Armbrust, M., Das, T., Davidson, A., Ghodsi, A., Or, A., Rosen, J., Stoica, I., Wendell, P., Xin, R., Zaharia, M.: Scaling Spark in the Real World: Performance and Usability. Proc. of the VLDB Endowment, 8 (12) (2015)

[7] Borthakur, D.: HDFS Architecture Guide. Hadoop Apache Project (2008)

[8] Brennen, C.E.: Fundamentals of Multiphase Flow. Cambridge University Press (2005)

[9] Chintapalli, S., Dagit, D., Evans, B., Farivar, R., Graves, T., Holderbaugh, M., Liu, Z., Nusbaum, K., Patil, K., Peng, B., Poulosky, P.: Benchmarking Streaming Computation Engines: Storm, Flink and Spark Streaming. Proc. of the IEEE Int. Parallel and Distributed Processing Symposium Workshops (2016)

[10] Duggan, J., Brodie, M.: Hephaestus: Data Reuse for Accelerating Scientific Discovery. Proc. of 7th Biennial Conf. on Innovative Data Systems Research (CIDR'15). USA (2015)

[11] Goncalves, B., Porto, F.: Managing Large-Scale Scientific Hypotheses as Uncertain and Probabilistic Data With Support for Predictive Analytics. Proc. of the IEEE Computing in Science and Engineering, 17 (5), pp. 35-43 (2015)

[12] Goncalves, B., Silva, F., Porto, F.: ϒ-DB: A System for Data-Driven Hypothesis Management and Analytics (2014). http://arxiv.org/abs/1411.7419

[13] Hagedorn, S., Götze, P., Saleh, O., Sattler, K.: Stream Processing Platforms for Analyzing Big Dynamic Data. Information Technology, 58 (4), pp. 195-205 (2016)

[14] Harry, L. van Trees, Bell, K., Tian, Z.: Detection, Estimation, and Modulation Theory. Part 1 - Detection, Estimation, and Filtering Theory. Second Edition. Willey, 1175 p. (2013)

[15] Hesse, G., Lorenz, M.: Conceptual Survey on Data Stream Processing Systems. Proc. of the IEEE 21st Int. Conf. on Parallel and Distributed Systems (2015)

[16] Hortonworks Data Platform. https://hortonworks.com/products/data-center/hdp/

[17] Hunt, P., Konar, M., Junqueira, F., Reed, B.: ZooKeeper: Wait-free Coordination for Internet-Scale Systems. Proc. of the USENIX Annual Technical Conf. (2010)

[18] Ioannidis, J.P.: Why Most Published Research Findings Are False. PLoS Medicine, 2 (8) (2005)

[19] Kafka-python. http://kafka-python.readthedocs.io/en/master/index.html

[20] Kalinichenko, L., Kovalev, D., Kovaleva, D., Malkov, O. Methods and Tools for Hypothesis-driven Research Support: a Survey. Informatica and Appications, 9 (1), pp. 28-54 (2015)

[21] Marz, N., Warren, J.: Big Data: Principles and Best Practices of Scalable Realtime Data Systems. Manning Publications Co. USA. 1st edition (2015)

[22] Mathiya, B., Desai, V.: Apache Hadoop Yarn Parameter Configuration Challenges and Optimization. Proc. of the Int. Conf. on Soft-Computing and Network Security (ICSNS -2015). Coimbatore. India (2015)

[23] Quanta Cloud Technology. http://qct.io/

[24] Rysak, A., Litak, G., Mosdorf, R.: Analysis of Non-stationary Signals by Recurrence Dissimilarity. Recurrence Plots and Their Quantifications: Expanding Horizons (2016)

[25] Salvetti, M.V., Geurts, B., Meyers, J., Sagaut, P.: Quality and Reliability of Large-Eddy Simulations. Springer. Netherlands (2008)

[26] Schales, D., Hu, X., Jang, J., Sailer, R., Stoecklin M., Wang, T.: FCCE: Highly Scalable Distributed Feature Collection and Correlation Engine for Low Latency Big Data Analytics. Proc. of 2015 IEEE 31st Int. Conf. on Data Engineering, pp. 1316-1327 (2014)

[27] Spark documentation. Pyspark.streaming module. https://spark.apache.org/docs/1.6.3/api/python/pyspark.streaming.html

[28] Spark Programming Guide. https://spark.apache.org/docs/1.6.3/programming-guide.html

[29] Stonebraker, M., Çetintemel, U., Zdonik, S.: The 8 Requirements of Real-time Stream Processing. SIGMOD Rec., 34 (4), pp. 42-47 (2005)

[30] Ünalmis, Ö.H.: Subsea Multiphase Flowmeter: Performance Tests in Multiphase Flow Loop. Society of Petroleum Engineers

[31] Wilcox, D.C.: Turbulence modeling for CFD. D C W Industries (2006)

[32] Zagarola, M.V., Smits, A.J.: Mean-flow Scaling of Turbulent Pipe Flow. J. of Fluid Mechanics, 373, pp. 33-79 (1998)

[33] Байесовский подход к теории вероятностей. Примеры Байесовских рассуждений. Глава 6. 2007. http://www.machinelearning.ru/wiki/images/4/43/BayesML-2007-textbook-2.pdf

[34] Критерий Стьюдента. http://www.machinelearning.ru/wiki/index.php?title=Критерий_Стьюдента

[35] Проверка статистических гипотез. http://www.machinelearning.ru/wiki/index.php?title=Проверка_статистических_гипотез

[36] Самарев, Р.С.: Обзор состояния области потоковой обработки данных. Труды ИСП РАН, (1), сс. 231-260 (2017)

[37] Скворцов Н.А., Аввакумова, Е.А., Брюхов, Д.О., Вовченко, А.Е., Вольнова, А.А., Длужневская, О.Б., Кайгородов, П.В., Калиниченко, Л.А., Князев, А.Ю., Ковалева, Д.А., Малков, О.Ю., Позаненко, А.С., Ступников, С.А.: Концептуальный подход к решению задач в астрономии. Астрофизический бюллетень, 71 (1), сс. 122-133 (2016)

[38] Тарасов, Е.А.: Сокращение числа виртуальных экспериментов с помощью оценки корреляций параметров взаимодействующих гипотез. Сб. трудов XVIII Межд. конф. DAMDID/RCDL, сс. 383-388 (2016)