## Начальное знакомство со средой RapidMiner Studio. Лабораторная работа №1

## Методические указания

Целью выполнения лабораторной работы является получение начальных навыков работы с программным комплексом RapidMiner Studio.

* + 1. Теоретические основы

RapidMiner Studio (рисунок 1) [4]. представляет собой многопользовательскую программную платформу, которая служит для создания, передачи и обслуживания наукоемких данных. Платформа открыта и расширяема, ускоряет создание полных аналитических рабочих процессов от подготовки данных для моделирования до развертывания бизнеса в единой среде значительно повышая эффективность и сокращая время, необходимое для создания проектов в области данных.



Рисунок 1 – Эмблема одной из версий пакета RapidMiner Studio Скачать актуальную версию платформы можно с сайта [4].

Профессиональная лицензия платная. В стандартной лицензии AGPL доступно 10000 колонок и ограничение в один логический процесс. Существует версия для учебного процесса. Пакет RapidMiner Studio установлен в вычислительных лабораториях центра информационных технологий ГУАП в зданиях на улице Ленсовета и на улице Большая Морская.

* + 1. Описание инструмента

Начальное окно, появляющееся после загрузки платформы, представлено на рисунке 2. Оно предлагает выбрать режим создания нового проекта (рисунок 3) или шаблон для реализации неких ранее созданных разработчиком платформы процессов (рисунок 4). Эта страница на любом этапе работы системы может быть вызвана командой File -> New process.

Бесплатная версия системы позволяет в течение только одного месяца после первичной загрузки системы пользоваться возможностями режимов Turbo Prep, автоматизирующего процесс подготовки данных, и Auto Model, предназначенного для подключения типовых шаблонов обработки данных. В дальнейшем для активации этих режимов потребуется предпринять некоторые специальные действия, а для пользователей будет доступен только основной режим работы через Blank.

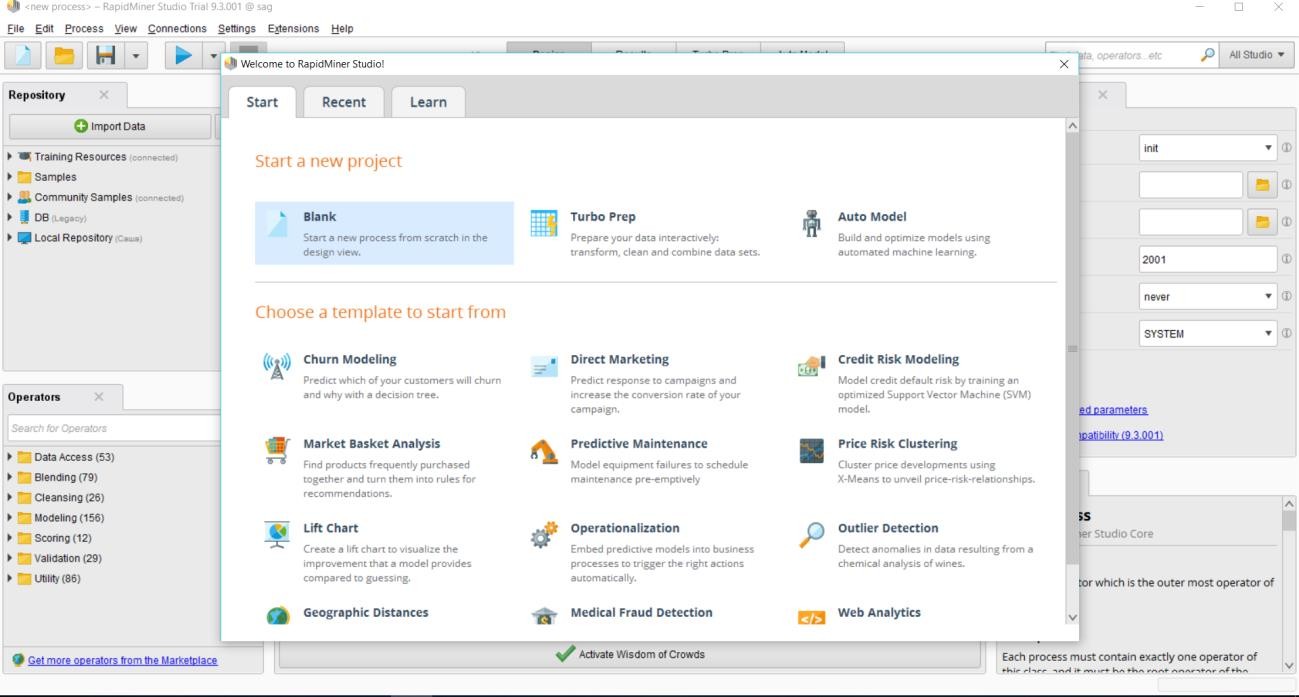


Рисунок 2 – Начальное окно запуска системы RapidMiner Studio Выбрав режим Start / Blank мы попадаем в главное окно (рисунок 5). В

этом случае появляется возможность конструировать обработку с самого начала. Кроме того, из этого окна существует возможность вернуться к предыдущему сеансу (Recent) и обратиться к системе пошагового обучения (Learn) (рисунок 2).

После установки в качестве примера пакет предлагает набор вариантов данных для обработки (рисунок 5 вкладка Samples). Так, например, используемый в некоторых обучающих примерах и поставляемый вместе с пакетом набор данных Titanic содержит сведения о пассажирах печально известного одноименного судна. При просмотре этого набора данных пользователю демонстрируется структура его атрибутов (колонок, полей) с указанием параметров каждого из них (рисунок 6). В наборе Titanic имеется

1309 экземпляров (записей, строк) каждый из которых имеет 12 атрибутов (колонок).

Ознакомиться с содержимым экземпляров существующего набора данных можно после того, как он будет открыт. Для этого надо щелкнуть правой клавишей мыши по имени набора и выбрать пункт Open всплывающего меню (рисунок 7). Обратите внимание на фильтр отсутствующих (missing) атрибутов и меток, который появляется в правом верхнем углу открывшейся панели набора. Одновременно у пользователя появляется возможность получения статистических сведений по данным набора (Statistics) и графической интерпретации данных (Visualizations).

|  |  |  |  |
| --- | --- | --- | --- |
| Начало работы  **Start a new project** |  | | |
|  | | Новый процесс |
|  |  | **Blank** |
|  | |
|  | Подготовка данных |
|  | **Turbo prep** |
|  |
|  | |
|  | Построение и оптимизации моделей с помощью машинного обучения |
|  | |
| **Auto model** |
|  | | |

Рисунок 3 – Предлагаемые варианты начала работы



**Churn Modeling**

Прогнозирование оттока клиентов с помощью дерева решений



**Direct Marketing**

Прогнозирование реакции на маркетинговую кампанию и повышение ее рейтинга



**Credit Risk Modeling** Моделирование кредитного риска с помощью машины опорных векторов



**Market Basket Analysis** Поиск ассоциативных правил



**Lift Chart**

Создание диаграммы прогнозирования



**Geographic Distances**

Расчет растояний по карте



**Price Risk Clastering**

Кластеризация ценовых рисков



**Outlier Detection** Обнаружение аномалий в результатах химического анализа



**Web Analytics**

Анализ посещаемости сайта



**Predictiv Maintenance** Предсказание отказов оборудования



**Operationalization**

Встраивание

прогнозирования в модель бизнес-процесса



**Medical Fraud Detection** Определение медицинских ошибок

Рисунок 4 – Предлагаемые шаблоны

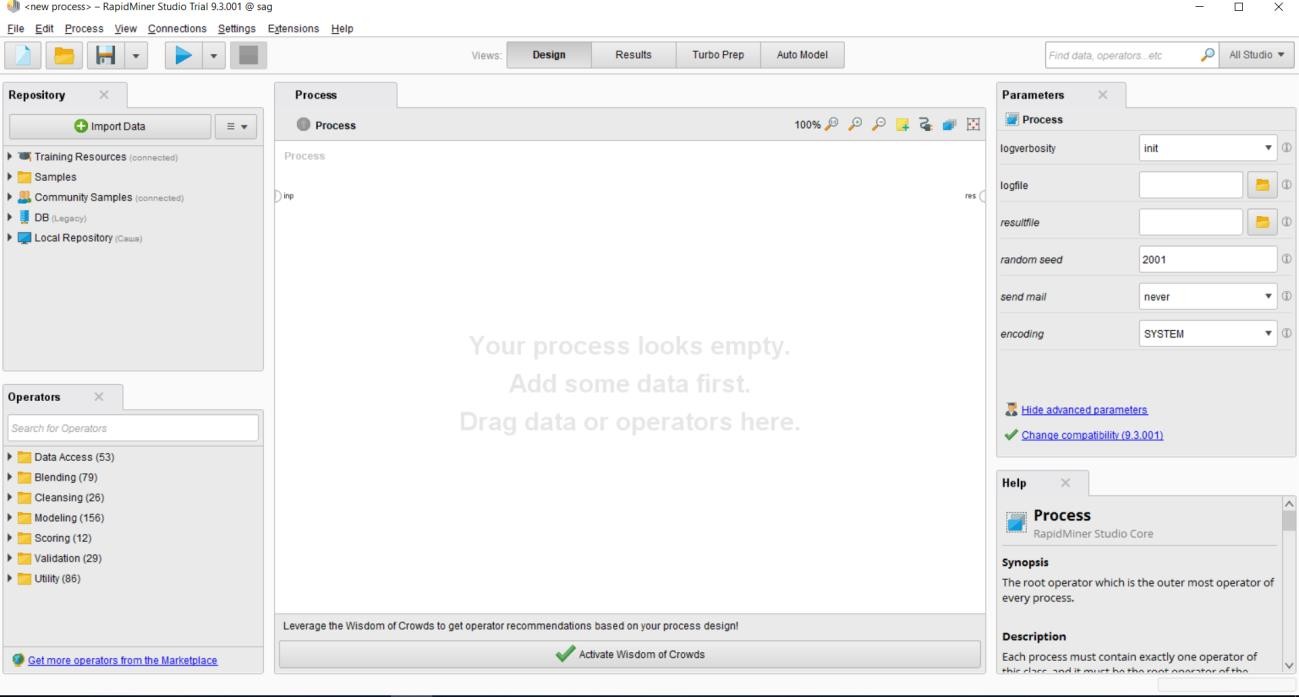


Рисунок 5 – Главное окно системы RapidMiner Studio для конструирования произвольного процесса

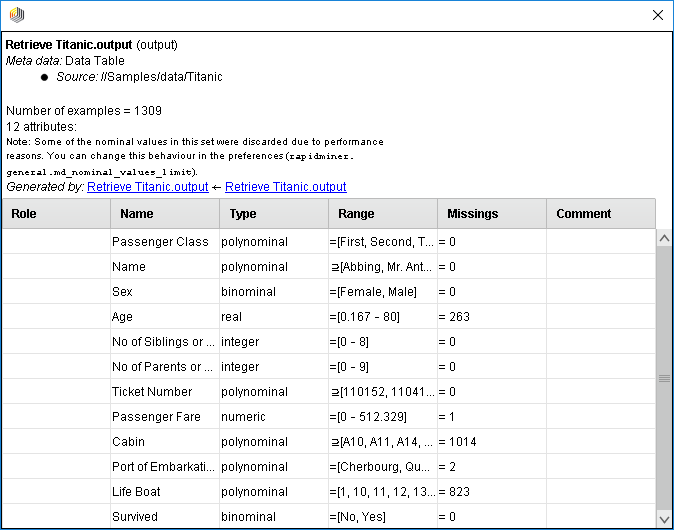


Рисунок 6 – Структура атрибутов набора данных Titanic

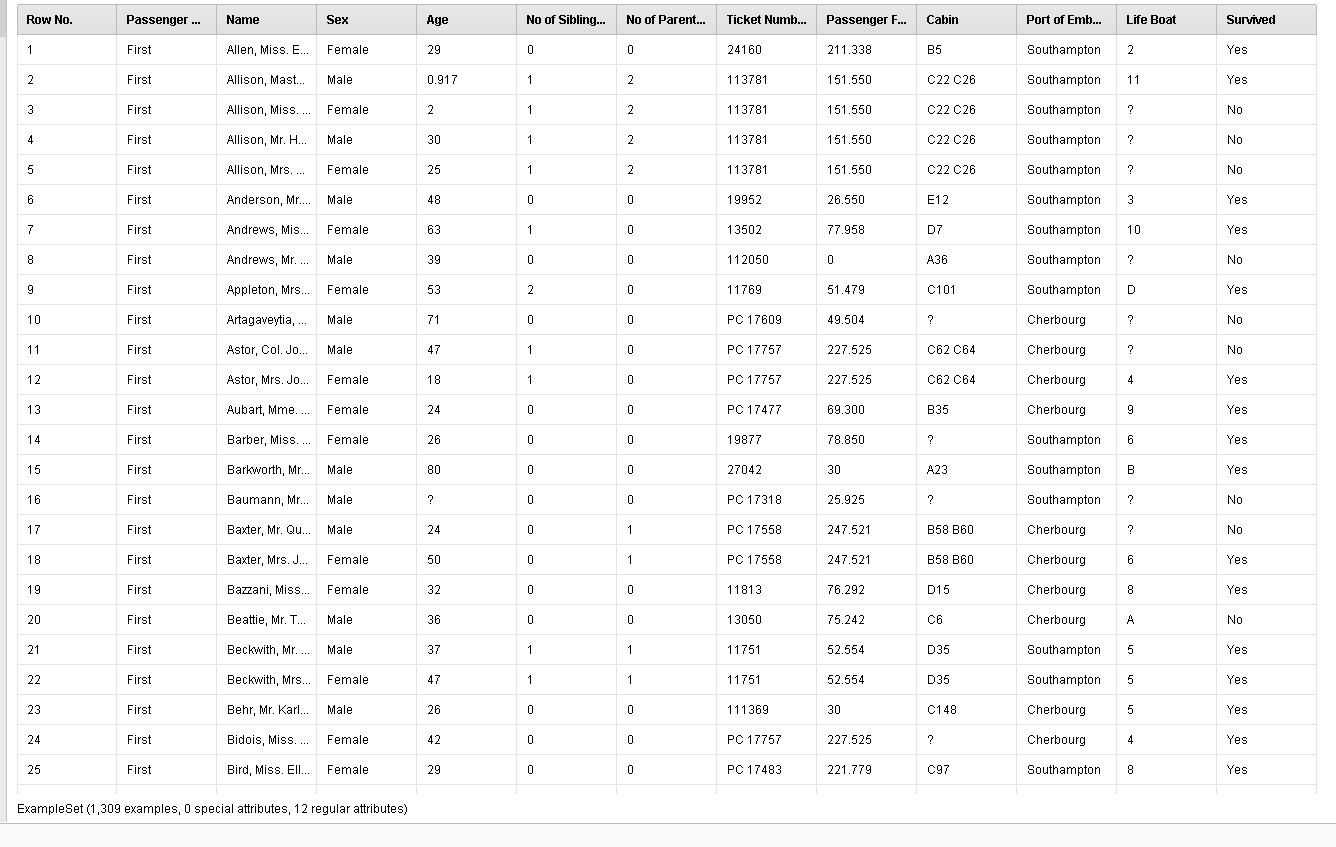


Рисунок 7 – Набор данных Titanic

Папки, предназначенные для хранения данных, в терминологии RapidMiner называются репозитории. Именно там в папке Samples хранились и данные Titanic. Для хранения ваших (пользовательских) данных предназначена папка Local Repository. Находящаяся в ней папка data предназначена для хранения наборов данных пользователя, которые могут быть подготовлены отдельно другими программными средствами. В качестве примера рассмотрим подготовленный нами средствами Excel на основе типовых математических функций набор данных из таблицы 1 (рисунок 8).

Таблица 1 – Детерминированные тестовые числовые последовательности

Детерминированные тестовые числовые

последовательности

200

150

100

50

0

-50

-100

Константа

Линейная

Квадратичная

Кубическая

Показательная Гармоническая d - импульсная Ступенчатая

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Номер варианта | Название | Математическое выражение | Параметры | Значения | | |
| 𝑎 | 𝑏 | 𝑐 |
| 1 | Константа | 𝑦[𝑛] = 𝑐 | 𝑐 |  |  | 50 |
| 2 | Линейная | 𝑥[𝑛] = 𝑎𝑛 + 𝑏 | 𝑎, 𝑏 | 1 | -50 |  |
| 3 | Квадратичная | 𝑥[𝑛] = 𝑎(𝑛 − 𝑏)2 + 𝑐 | 𝑎, 𝑏, 𝑐 | 0.01 | 100 | -50 |
| 4 | Кубическая | 𝑥[𝑛] = 𝑎𝑛3 + 𝑏𝑛 + 𝑐 | 𝑎, 𝑏, 𝑐, | 0.00001 | 0 | -30 |
| 5 | Показательная | 𝑥[𝑛] = 𝑎𝑛, где 𝑎 > 0, 𝑎 ≠ 1 | 𝑎 | 1.02 |  |  |
| 6 | Гармоническая | 𝑥[𝑛] = 𝑎𝑠𝑖𝑛(𝑏𝑛 + 𝑐) | 𝑎, 𝑏, 𝑐 | 20 | 0.1 | 3.14159 |
| 7 | d - импульсная | 𝑐, если 𝑛 = 𝑎;  𝑥[𝑛] = {0, если 𝑛 ≠ 𝑎. | 𝑎, 𝑐 | 70 |  | 100 |
| 8 | Ступенчатая | 𝑐, если 𝑛 ≥ 𝑎;  𝑥[𝑛] = {  0, если 𝑛 < 𝑎. | 𝑎, 𝑐 | 120 |  | 75 |

Рисунок 8 – Подготовленные в Excel тестовые данные Импортируем подготовленные данные в RapidMiner и разместим их в

1

7

13

19

25

31

37

43

49

55

61

67

73

79

85

91

97

103

109

115

121

127

133

139

145

151

157

163

169

175

181

187

193

199

виде таблицы в папке Local Repository. Для этого в главном окне системы (рисунок 5) во вкладке Repository нажмем кнопку +Import Data. Откроется панель диалога с вопросом Where is you data? Скорее всего, ваши данные

находятся на вашем же компьютере, поэтому нажмите кнопку My Computer. Откроется окно мастера, позволяющего найти файл на вашем компьютере и указать место и имя файла в репозитории RapidMiner. Мы присвоили ему имя Тестирование RM. Автоматически созданная структура атрибутов импортированного набора данных показана на рисунке 9, а их графическое представление на рисунке 10.

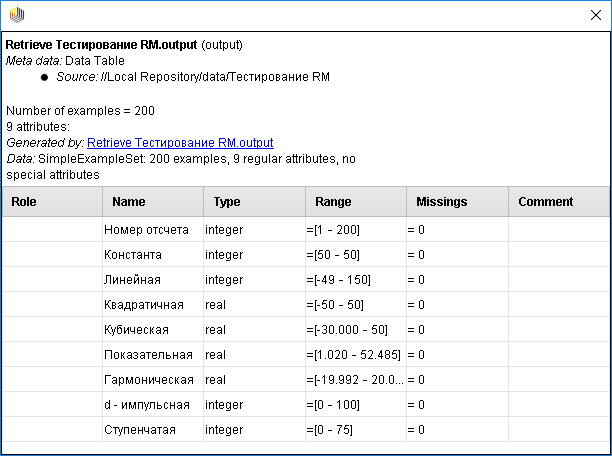


Рисунок 9 – Структура атрибутов тестового набора данных в RapidMiner

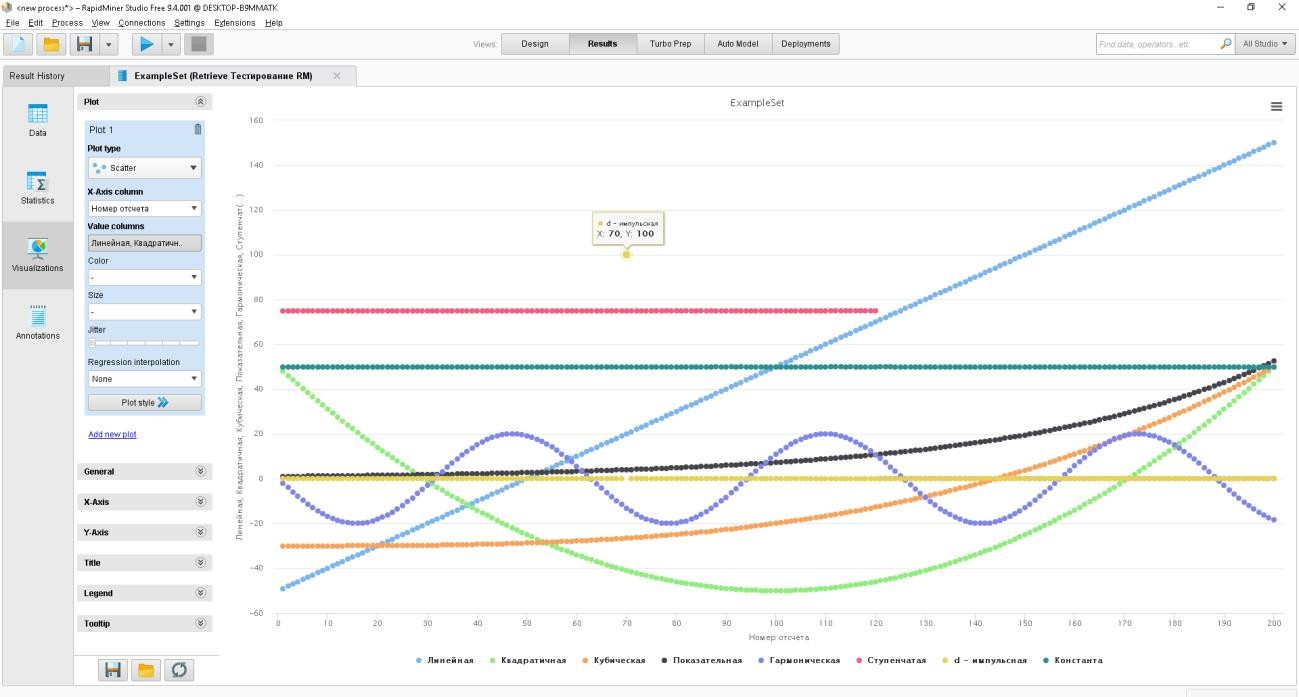


Рисунок 10 – Отображение результатов импорта данных из Excel средствами RapidMiner

## Исходные данные

Исходными данными являются предлагаемые разработчиками пакета RapidMiner Studio образцы данных (Samples), и тестовые данные, подготовленные студентом.

## Порядок выполнения

1. Установите у себя дома пакет пробную версию пакета RapidMiner или воспользуйтесь версией, установленной в лабораториях университета.
2. Ознакомьтесь с процедурой запуска пакета в работу, перейдите в начальное окно запуска системы RapidMiner (рисунок 2). Создайте новый процесс, для чего нажмите кнопку Blanc вкладки Start открывшегося окна Welcome to RapidMiner Studio.
3. Найдите вкладку Repository, ознакомьтесь с ее структурой и содержимым. Изучите структуры данных различных наборов данных из папки Samples и определите назначения атрибутов и полей. Изучите данные различных наборов, сравните между собой одноименные наборы (например, Titanic, Titanic Training и Titanic Unlabeled) и найдите отличия в их атрибутах и содержимом.
4. Ознакомьтесь со статистическими параметрами изученных наборов данных в режиме Statistics. Воспользуйтесь графической интерпретацией изученных вами наборов данных в режиме Visualizations.
5. Создайте индивидуальный набор данных в произвольной системе программирования. Импортируйте данные в RapidMiner, убедитесь в отсутствии ошибок импорта и в возможность использовать эти данные. Ознакомьтесь со статистическими параметрами вашего набора данных

в режиме Statistics. Выполните графическую интерпретацию ваших данных в режиме Visualizations.

## Пример создания и обработки индивидуального набора данных

Рассмотрим работу программы с реальными данными на примере данных о городских объектах наследия города Уэлленд, Канада [5]. Время от времени они обновляются, поскольку администрация города вносит изменения в список наследия. В столбцах таблицы рисунок 11 содержится информация о почтовом индексе объекта, номере постановления, на основании которого он признан объектом наследия, адресе расположения, названии, годе постройки и годе реставрации. Всего таблица содержит 30 записей (рисунок 12). Загружаем таблицу с данными в репозиторий.



Рисунок 11 – Таблица данных о городских объектах наследия города Уэлленд (Welland, Canada)

Статистическая обработка введенных данных дала результаты, представленные на рисунке 12.

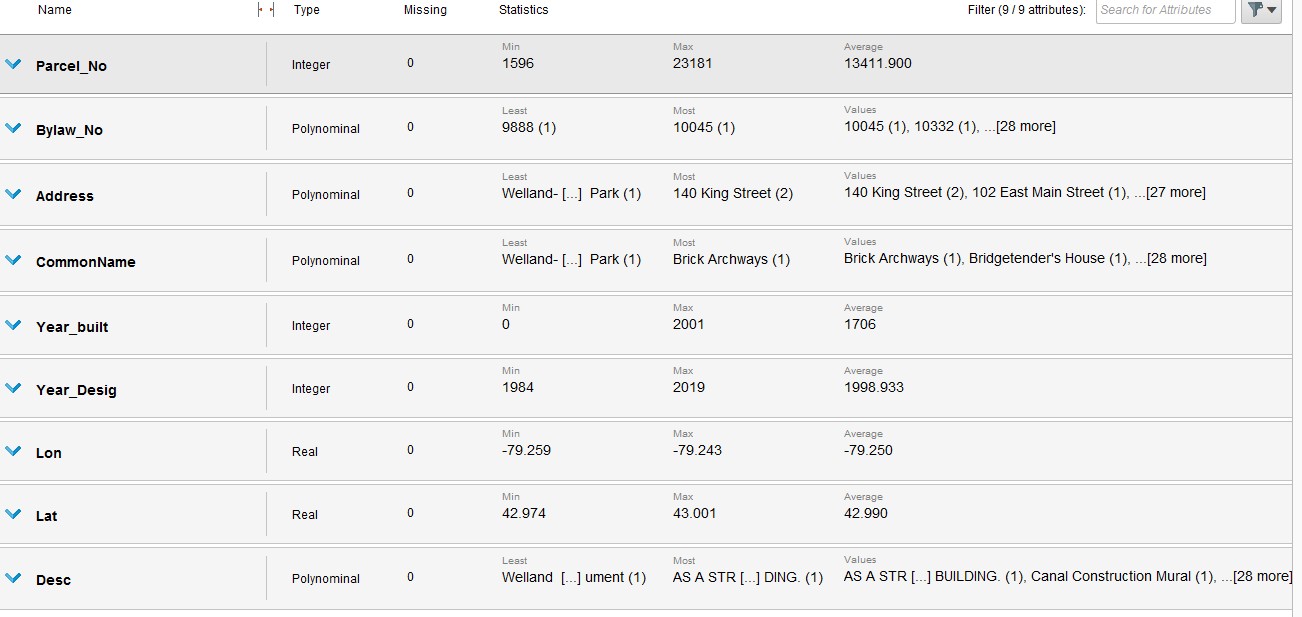


Рисунок 12 – Результаты статистической обработки (режим Statistics)

Построенный график изображен на рисунке 13. На нем отображена зависимость данных таблицы от года реконструкции конкретного объекта. Если мы также установим флаг Aggregate data, то на графике будет показаны сводные данные отображающие общее число отреставрированных в конкретный год объектов (рисунок 14).

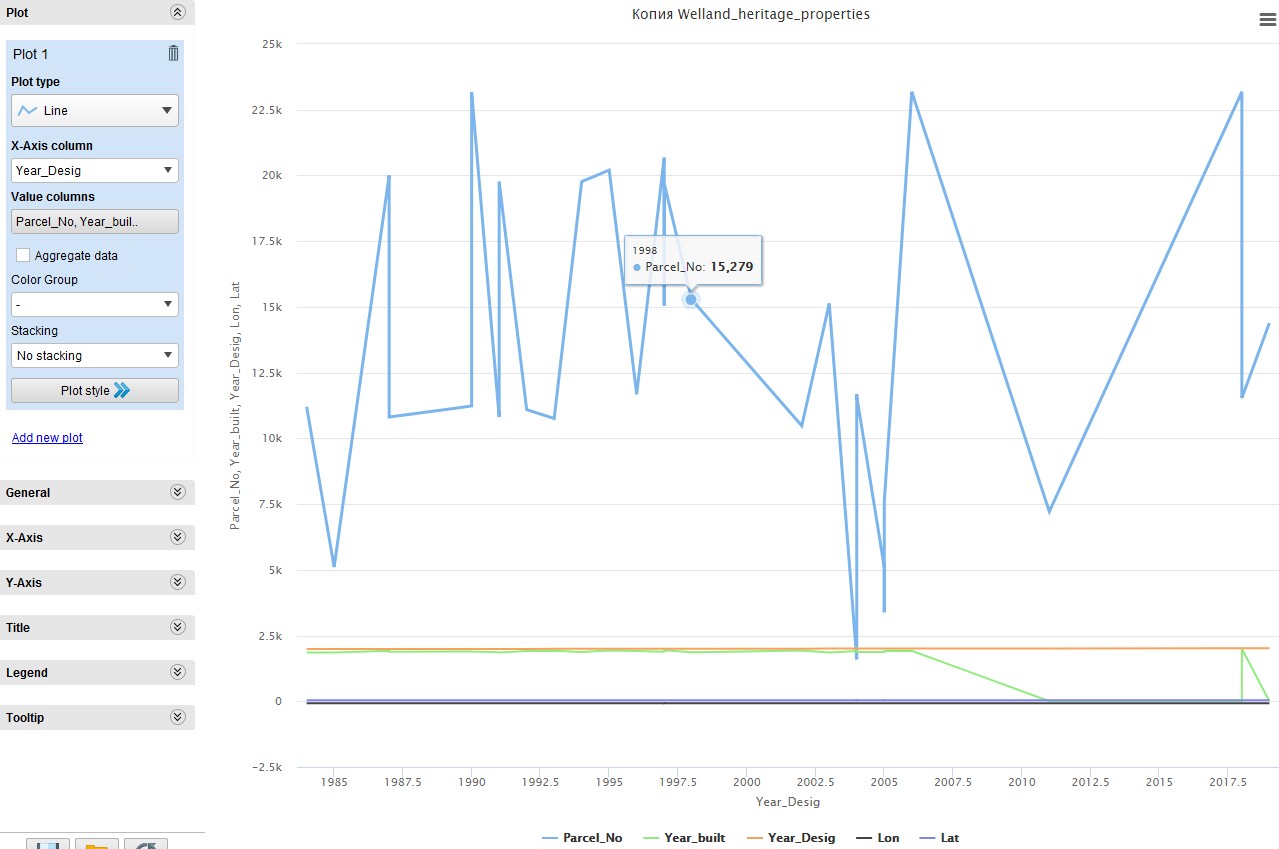


Рисунок 13 – Визуализация данных таблицы (режим Visualization)

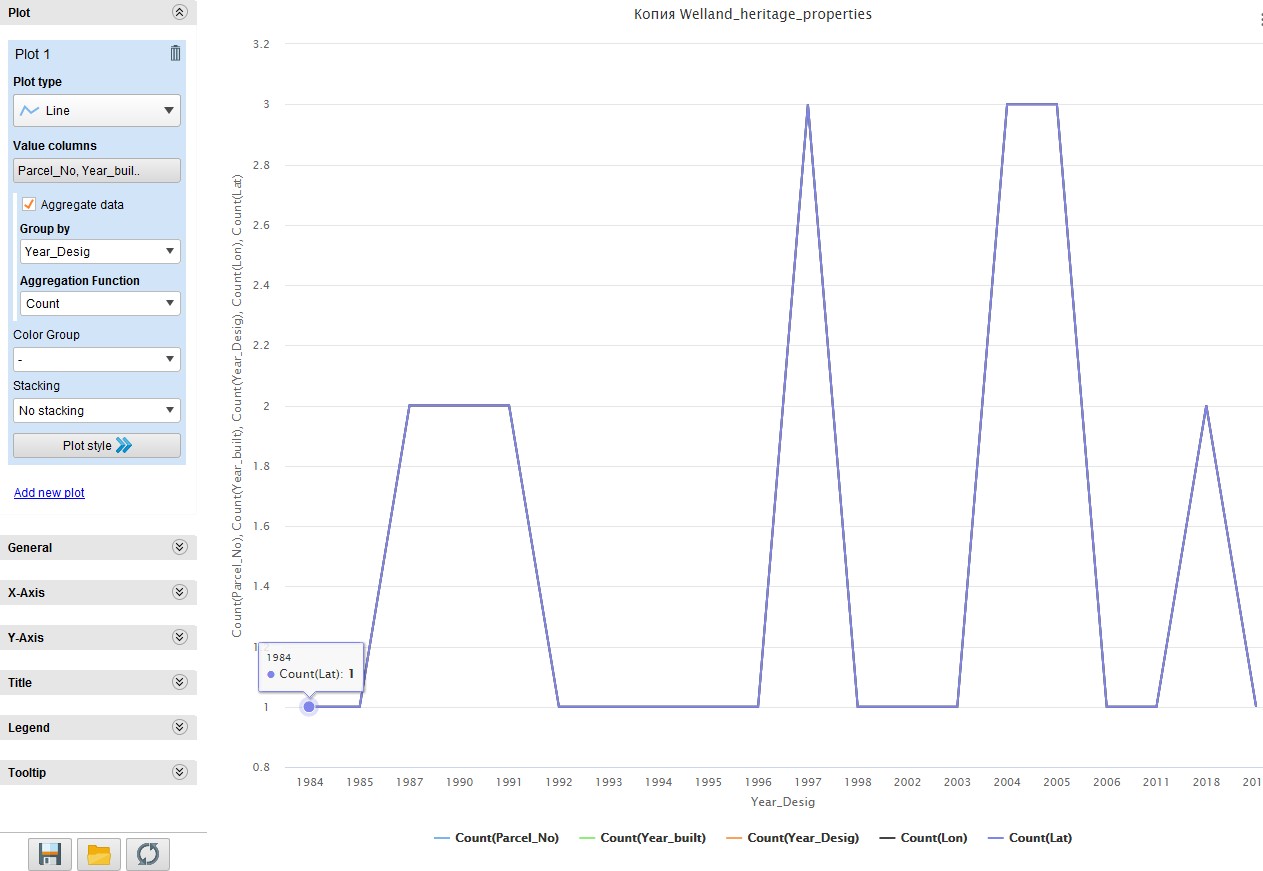


Рисунок 14 – Визуализация данных таблицы (режим Visualization, Aggregate data)

На основании полученной визуализации можно увидеть, что максимальное число одновременно отремонтированных в один год объектов равно трем. Такой результат наблюдался в 1997, 2004 и 2005 годах.

## Оформление отчета

Отчет о выполнении лабораторной работы должен содержать титульный лист, атрибуты изученных данных из папки Samples с указанием отличий, описание изученных данных, результаты их сравнения с другими близкими по смыслу наборами и результаты их графической интерпретации, структуру атрибутов индивидуальных данных студента и результаты графической интерпретации индивидуальных данных.

## Контрольные вопросы

1. Каково назначение пакета RapidMiner?
2. Какой тип лицензии на использования пакета RapidMiner, каковы права и обязанности пользователя пакета?
3. Что хранится в папках вкладки Repository?
4. Какова технология импорта данных в репозиторий пользователя?
5. Что такое атрибуты набора данных?
6. Какие возможности есть у режима Statistics?
7. Какие возможности есть у режима Visualizations?
8. Как и когда можно реализовать предлагаемые варианты начала работы с RapidMiner, а также предлагаемые шаблоны решаемых задач?
9. Какие возможности существуют при работе с главным окном?
10. Как можно воспользоваться системой обучения пакета RapidMiner и какие возможности предоставляет пользователю система помощи?

## Подготовка и преобразование данных. Лабораторная работа №2

## Методические указания

Целью выполнения работы является получение начальных навыков преобразования данных в RapidMiner Studio.

* + 1. Теоретические основы

В том или ином виде пакет RapidMiner Studio предназначен для обработки цифровых данных. Рассмотрим их возможную классификацию (рисунок 15).



Числовые

Текстовые

Многомерные

Матрицы Базы данных

Графические

Рисунок 15 – Виды цифровых данных пригодных для интеллектуального анализа

Временные

ряды

Кириллические

Латинские

Иероглифы

Векторные

Растровые

Простейшим вариантом данных для интеллектуального анализа является дискретный числовой ряд 𝑥[𝑛]. Он представляет собой конечную или бесконечную последовательность дискретных отсчетов некого процесса обычно являющегося функцией времени. Если последовательность бесконечная, то речь может идти об обработке ее фрагмента (выборки) конечной длины из 𝑁 отсчетов. Такая обработка может вестись как в обычном режиме, когда обрабатываются пакеты из последовательных выборок из 𝑁 отсчетов, так и в скользящем, когда на каждом цикле обработки из выборки исключается первый элемент 𝑥[𝑛], а в ее состав включается элемент 𝑥[𝑛 + 𝑁 + 1].

Дискретный числовой ряд может быть как детерминированным, так и случайным. Исходя из целей обучения, а также, принимая во внимание соображения по предстоящему тестированию создаваемого программного обеспечения, удобно использовать детерминированные временные ряды. В этом случае результат работы системы хорошо интерпретируется пользователем. Предполагается, что, если преобразование входных данных во входные задано очевидным математическим оператором, то и результат работы системы легко проверяется. В таблице 1 приведены варианты задания

различных дискретных последовательностей, которые могут быть использованы для исследований.

Память ЭВМ представляет собой последовательность скомпонованных из триггеров самостоятельных ячеек размером один или два байта. Как следствие, информация в памяти вычислительной установки хранится в виде двоичных чисел. Каждая ячейка памяти имеет свой номер, который обычно называют ее адресом. Максимально допустимое количество адресов в составе компьютера может быть достаточно большим, но, в силу особенностей конструкции вычислительной установки, кратным числу 2𝑛. Таблицы обрабатываемых пакетом RapidMiner данных размещаются в памяти вычислительной установки в строго определенном формате. В терминологии RapidMiner строки называются examples, а колонки attributes. Каждая колонка имеет стандартный набор собственных параметров. Иногда их также называют атрибутами.

Использовать конкретный физический адрес памяти для обращения к конкретной ячейке оказывается неудобным. В практике программирования традиционно применяется прием, использующий так называемые идентификаторы. Под идентификатором понимается символическое имя ячейки памяти. Параметром колонки может быть ее идентификатор (Name). Фактически он представляет собой имя колонки, используемое при выполнении действий с таблицами. Каждая система программирования имеет свои правила составления идентификаторов. В RapidMiner правила составления идентификаторов позволяют записывать идентификаторы как латынью, так и кириллицей строчными и прописными буквами.

Еще одним параметром являются так называемый тип данных. Под типом данных обычно понимают способ кодирования информации пользователя в памяти ЭВМ. В зависимости от решаемой задачи приходится хранить или целые и дробные числа, или коды символов. Для хранения единицы данных в зависимости от ее размера может использоваться несколько последовательных ячеек памяти. Целью выполнения лабораторной работы является изучение типов данных, используемых в RapidMiner, а также методов их преобразования от одного типа к другому.

Типы данных (Types) RapidMiner во многом совпадает с типами данных, используемыми в языках программирования высокого уровня или системах управления реляционными базами данных и представлен в таблице 2. Количество бит памяти, используемых для хранения единицы данных конкретного типа, представлено в колонке Размер и определяется способом кодирования. Типы данных в RapidMiner имеют иерархию, показанную на рисунке 16.

В среде RapidMiner присутствуют числовые (Numeric) и номинальные (Nominal) типы данных. К числовым типам данных относятся целочисленный тип (Integer), вещественный (Real), дат и времени (date, time, date\_ time).

Номинальные типы данных – это переменные категориального типа, рассматриваемые как отдельные символы или просто имена. С номинальными типами данных нельзя выполнять арифметические операции. К этим типам

относится привычный широко распространенный текстовый тип (Text), а также специальные типы данных предназначенные для хранения категориальных (тоже текстовых) нечисловых значений. К ним относятся тип данных биноминальный (Binominal), хранящий только два различных значения, и полиноминальный (Polynominal) предназначенный для хранения конечного числа значений. Например, цвет радуги является категориальным типом данных, поскольку принимает такие значения, как красный, зеленый, синий и т.д. Важно отметить, что полиномиальное/биноминальное значение всегда является номинальным, но не наоборот.

Текстовые (nominal)

Числовые (numeric)

Типы данных RapidMiner

Рисунок 16 – Типы данных языка RapidMiner Таблица 2 – Типы данных RapidMiner

integer

real

date\_time

date

time

text

polynominal

binominal

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Описание | Пример | Размер  (бит) |
| nominal | Все виды текстовых значений полиноминальные и  биноминальные) | истина / ложь, красный,  зеленый, синий, желтый | 16 |
| numeric | Все виды числовых значений (дата, время, целые и  дробные) | 23, 11.23, 23.12.2014 17:59 | 32 |
| integer | Целые числа | 23, -5 или 11 024 768 | 32 |
| real | Дробные числа | 11.23 или -0,0001 | 64 |
| text | Случайный свободный текст без структуры. Позволяет отличать текст от полиноминальных  данных. | ФИО, название | 16 |
| binominal | Частный случай текстовых данных допускающий  только два значения | истина / ложь или да / нет | 8 |
| polynominal | Частный случай биноминальных данных больше,  чем с двумя значениями | красный, зеленый, синий,  желтый | 16 |
| date\_time | И дата, и время | 23.12.2014 17:59 | 64 |
| date | Дата без времени | 23.12.2014 | 24 |
| time | Время без даты | 17:59 | 24 |

* + 1. Описание инструмента

Пакет RapidMiner создан с использованием объектно-ориентированной технологии и предусматривает графический метод программирования. С помощью графических возможностей дисплея пользователь составляет программу в виде набора операторов и связей между ними. Связи задают

последовательность выполнения алгоритма обмена данными, а сами данные представляют собой стандартные для пакета структуры, оформленные в виде объектов класса ExampleSet. Общее количество таких операторов достаточно велико и с их полным перечнем можно ознакомиться просматривая содержимое вкладки Operators рисунок 5. Учитывая тему настоящей лабораторной работы, в первую очередь имеет смысл изучить содержимое вкладок Blending / Attributes, которые содержат операторы, необходимые для работы с данными. В дальнейшем программирование будет сводиться к перетаскиванию мышкой на рабочее поле необходимых операторов и установлению соединений.

Поскольку каждый тип данных использует свой способ кодирования, при составлении программ иногда приходится делать переход от данных одного типа к другим без потери их смыслового значения. Такой переход называется преобразованием данных. При использовании RapidMiner преобразование данных реализуется за счет набора специальных функций. Их краткое описание представлено в таблице 3 в том порядке, как они размещены в папке Blending / Attributes / Types, а полное с упорядочением по алфавиту имеется в приложении А.

Таблица 3 – Операторы преобразования данных в RapidMiner

|  |  |
| --- | --- |
| Название | Краткое описание оператора |
| Numerical to Binominal | Изменяет тип выбранных числовых атрибутов на биноминальный тип и  отображает все значения этих атрибутов как биноминальные значения. |
| Numerical to Polynominal | Изменяет тип выбранных числовых атрибутов на полиноминальный тип и отображает все значения этих атрибутов как полиномиальные значения. Если вам нужен более сложный метод нормализации, используйте  операторы дискретизации. |
| Numerical to Real | Изменяет тип выбранных числовых атрибутов на тип с плавающей точкой  тип и отображает все значения этих атрибутов как значения типа Real. |
| Numerical to Date | Преобразует выбранный номинальный атрибут в выбранный тип даты и времени. Номинальные значения преобразуются в значения даты и/или времени. Это преобразование выполняется относительно указанной строки  формата даты. |
| Real to Integer | Изменяет тип выбранных атрибутов типа Real на целочисленный тип  отображает все значения этих атрибутов как целочисленные. |
| Nominal to Binominal | Изменяет тип выбранных номинальных атрибутов на биноминальный тип  и отображает все значения этих атрибутов как биноминальные. |
| Nominal to Text | Изменяет тип выбранных атрибутов на текстовые и переводит все значения  этих атрибутов в соответствующие строковые значения. |
| Nominal to Numerical | Изменяет тип выбранных нечисловых атрибутов на числовой тип и  отображает все значения этих атрибутов как числовые. |
| Nominal to Date | Преобразует выбранный номинальный атрибут в выбранный тип даты и времени. Номинальные значения преобразуются в значения даты и / или времени. Это преобразование выполняется относительно указанной строки  формата даты. |
| Text to Nominal | Изменяет тип выбранных текстовых атрибутов на номинальный и отображает все значения этих атрибутов как соответствующие  номинальные значения. |
| Date to Numerical | Изменяет тип выбранного атрибута даты на числовой тип и отображает все значения этого атрибута как числовые значения. Вы можете указать, какой именно компонент даты или времени должен быть извлечен. Вы также можете указать, относительно какой даты или времени должна извлекаться  информация о компонентах. |

|  |  |
| --- | --- |
| Название | Краткое описание оператора |
| Date to Nominal | Анализирует значения даты указанного атрибута по отношению к заданной  строке формата даты и преобразует значения в номинальные. |
| Parse Numbers | Изменяет тип выбранных номинальных атрибутов на числовой тип. Он  также отображает все значения этих атрибутов в числовые значения, анализируя числа, если это возможно. |
| Format Numbers | Переформатирует выбранные числовые атрибуты в соответствии с  указанным форматом и изменяет атрибуты на номинальные. |
| Guess Types | Повторно определяет типы значений всех атрибутов входного ExampleSet  и изменяет их соответственно. |

Возможности преобразования типов данных из одного в другой представлены в таблице 4.

Таблица 4 – Варианты преобразования данных в пакете RapidMiner

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 – преобразование бессмысленно  1 – пример преобразования имеется в пособии  + – преобразование возможно, но не реализовано, т.к. требуют последовательного применения нескольких операторов  - – преобразование невозможно | Numerical | Integer | Real | Date\_time | Date | Time | Nominal | Text | Polynominal | Binominal |
| Numerical | 0 | + | 1 | + | + | + | + | 0 | 1 | 1 |
| Integer | + | 0 | 1 | 0/- | 0/- | 0/- | + | 0 | + | + |
| Real | + | 1 | 0 | 0/- | 0/- | 0/- | + | 0 | + | + |
| Date\_time | + | 0 | 0 | 0 | + | + | + | + | + | + |
| Date | 1 | 0 | 0 | + | 0 | + | 1 | + | + | + |
| Time | + | 0 | 0 | + | + | 0 | + | + | + | + |
| Nominal | 1 | + | + | + | 1 | + | 0 | 1 | + | 1 |
| Text | 0 | 0 | 0 | + | + | + | 1 | 0 | + | + |
| Polynominal | 1 | + | + | + | + | + | + | + | 0 | + |
| Binominal | 1 | + | + | + | + | + | 1 | + | + | 0 |

Перечисленные операторы имеют один вход exa и несколько вариантов выходов. Подавляющее большинство из них имеет только два выхода exa и ori, а операторы Nominal to Binominal и Nominal to Numerical имеют еще дополнительный выход pre (рисунок 17).

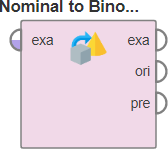
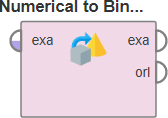


Рисунок 17 – Варианты операторов преобразования данных в RapidMiner Во всех вариантах операторов вход exa (example set) предполагает

структуру данных класса ExampleSet. В качестве входных данных также могут использоваться либо набор данных из репозитория, либо выходные данные других операторов. Во входном наборе ExampleSet должен иметь хотя бы один атрибут того типа, из которого производится преобразование данных, поскольку при отсутствии такого атрибута использование этого оператора не имеет смысла.

Выходной порт exa формирует объект ExampleSet с результатами обработки конкретного оператора. На выходной порт ori передается в полном объеме входной набор данных без изменений. Обычно он используется для

повторного применения одного и того же ExampleSet в последующих операторах или для просмотра ExampleSet в рабочей области результатов.

Если существует еще один выход pre, то он генерирует модель предварительной обработки (preprocessing model), которая содержит информацию о параметрах данного оператора в текущем процессе.

Если оператор перемещен в поле Process рабочего окна RapidMiner, то у пользователя появляется возможность настройки его параметров. В правой части рабочего окна становятся доступными имеющие непосредственное отношение к выбранному оператору две вкладки: Parameters и Help (рисунок 18). Каждая из них может быть дополнительно развернута на большую часть экрана. Первая предоставляет возможность индивидуальной настройки каждого оператора, а вторая обеспечивает пользователя необходимой информацией на английском языке.

Набор параметров индивидуален для каждого оператора, однако одноименные в рамках пакета RapidMiner параметры имеют одинаковый смысл. В приложении А приведена упорядоченная по алфавиту таблица с пояснением назначения каждого параметра. В то же время соответствующие пояснения, правда на английском языке, одновременно появляются на вкладке Help. Приложение А содержит таблицы А1 – А6 с переводом описаний некоторых используемых в пакете операторов, в частности, операторов преобразования типов данных.

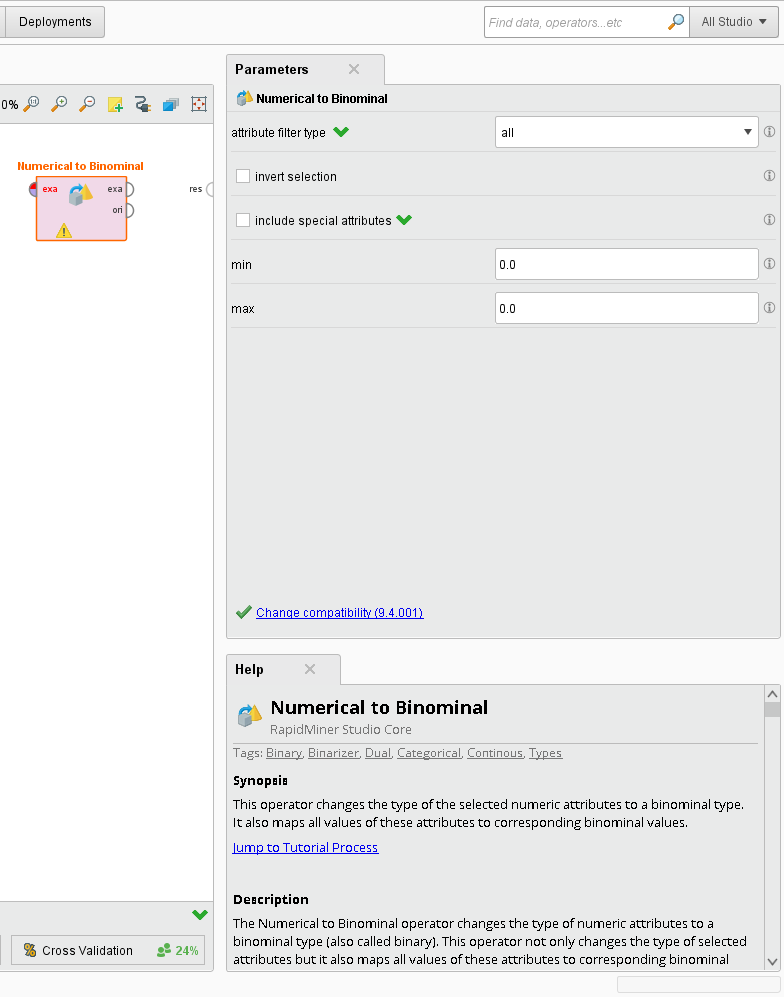


Рисунок 18 – Настройка параметров оператора

На рисунке 19 показан пример графического программирования в пакете RapidMiner. Операторы языка, оформленные в виде прямоугольников с входными и выходными портами, с помощью мышки устанавливаются в рабочем окно Process. Обычно первым оператором является оператор Retrieve получить доступ к хранимой информации в репозитории и загрузить ее в процесс. Выход этого оператора, обозначенный как out, также мышкой соединяется с входом следующего оператора, обозначаемого exa. Тогда на этом входе образуется объект данных формата RapidMiner, который в дальнейшем может быть подвергнут обработке в соответствии со свойствами конкретного оператора описанными во вкладке Help в правой нижней части рабочего окна. Как правило, операторы имеют набор собственных параметров, значения которых устанавливаются во вкладка Parameters в правой верхней части рабочего окна. Конкретный вариант настройки таких параметров показан в таблице 5.

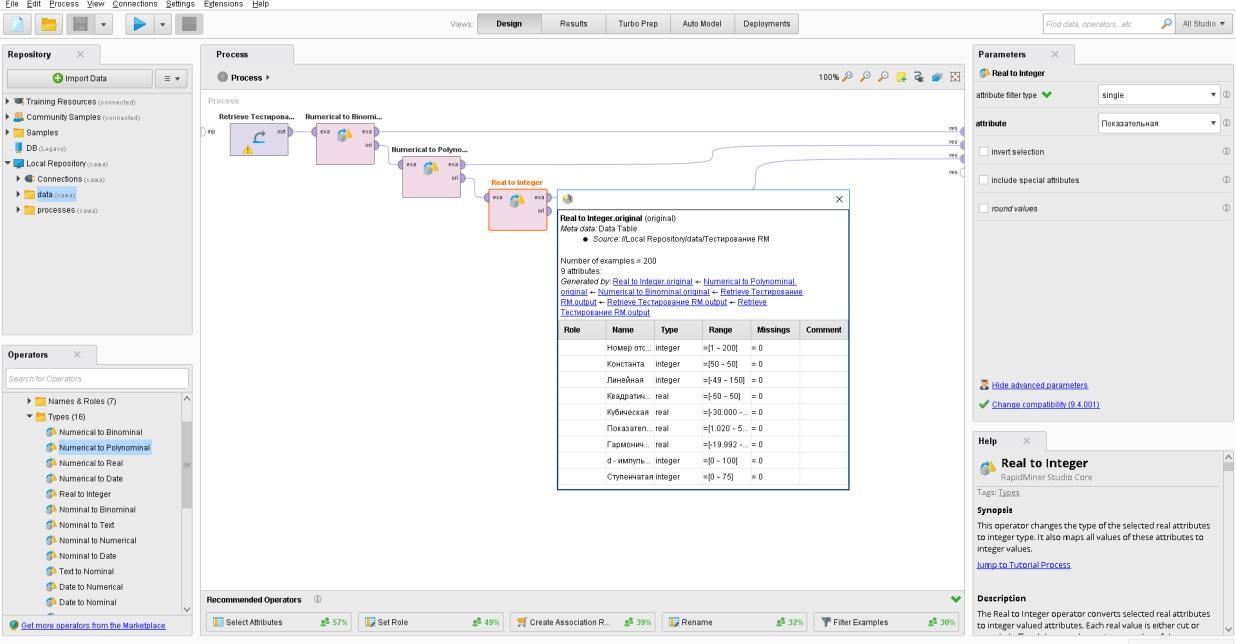


Рисунок 19 – Примеры программирования преобразования данных Таблица 5 – Настройки параметров процессов изменения ролей атрибутов

|  |  |  |
| --- | --- | --- |
| Оператор | Настройка | |
| Retrieve | Тестирование RM | |
| Numerical to Binominal | attribute filter type | single |
| attribute | Гармоническая |
| invert selection | false |
| include special attributes | false |
| min | -10.0 |
| max | 10.0 |
| Numerical to Polynominal | attribute filter type | single |
| attribute | Квадратичная |
| invert selection | false |
| include special attributes | false |
| Real to Integer | attribute filter type | single |
| attribute | Квадратичная |
| invert selection | false |
| include special attributes | false |
| round values | false |

Результат работы оператора появляется на выходе exa. Соединяя его с входом следующего оператора, можно строить систему последовательной обработки данных. Кроме этого, если существует выходной порт ori, то подключение к нему позволяет получить входной набор данных без обработки оператором.

Потребность в преобразовании типов данных возникает при подготовке реальных данных к обработке. Необходимо иметь в виду различие способов хранения числовых и текстовых данных и, как следствие, отличие алгоритмов программирования последующей обработки.

Еще одним параметром каждого атрибута является его диапазон изменения (Range). Обычно он вычисляется автоматически для конкретной таблицы и носит информативный характер. В первую очередь он нужен для отыскания случайных выбросов в реальных данных.

Параметр атрибута количество отсутствующих значений (Missing) тоже вычисляется автоматически. Параметр фиксирует факт наличия или отсутствие в составе колонки пропущенных данных. Если они присутствуют в таблице, то при составлении алгоритма решения задачи могут дополнительно потребоваться специальные действия.

Использование параметра комментарий (Comment) никак не регламентировано.

Отдельное значение имеет параметр роль (Role). Он может быть по умолчанию пустым (regular) или специальным (special) – label, id, prediction, cluster, weight, batch (рисунок 20). Существует два типа специальных значений

– те, которые описывают все таблицы и те, которые распространяются на конкретную таблицу (экземпляр).

Роль (**Role**)

# общая

обычная

(**regular**)

устанавливаемая

идентификатор(*id*)

**Generate ID**

добавляющая

атрибут

определенная пользователем (*user defined*)

**Generate Attributes**

кластер (*cluster*)

**Set Role, k-Means**

для

# конкретных

специальная (**special**)

таблиц

метка (*label*) **Set Role**

пустая (*regular*) **Set Role**

вес (*weight*)

**Set Role**

партия (*batch*) **Set Role**

Рисунок 20 – Варианты значений параметра (Role)

прогноз (*prediction*) **Set Role, Apply Model**

Так, например, задание значения параметра роль (Role) в виде label указывает, что соответствующий атрибут используется для обучения. Такой атрибут может называться «целевой переменной» или «классом».

Для указания записи таблицы может использоваться роль id. Она предназначена для создания нового атрибута таблицы, содержащего номера строк, и используемого для идентификации конкретной строки таблицы данных.

Значение роли prediction означает, что этот атрибут является результатом прогнозирования. Такие колонки появляются в таблицах после

выполнения оператора Apply Model, например, в результате решения задачи регрессии.

Аналогичная ситуация и со значением роли cluster. Атрибут с ролью кластера указывает на принадлежность таблицы с данными к определенному кластеру. Например, оператор k-means добавляет такое значение.

Weight – значение роли, которое используется в сочетании со значением роли label. Атрибут с ролью weight указывает вес примеров по отношению к label. Веса используются в процессах обучения для определения важности примеров.

Специальная роль batch указывает на принадлежность атрибута определенному пакету (группе).

Значение роли user defined (определяемая пользователем) может быть назначена атрибуту вводом в текстовое поле. Конкретную роль нельзя назначать нескольким атрибутам. Атрибуты с пользовательскими ролями игнорируются в процессе обучения, однако соответствующая колонка остается в пользовательской таблице.

В качестве примера был создан процесс ЛР2\_Установить роль (рисунок 21). Его настройки приведены в таблице 6, а результаты работы отображены на рисунке 22.

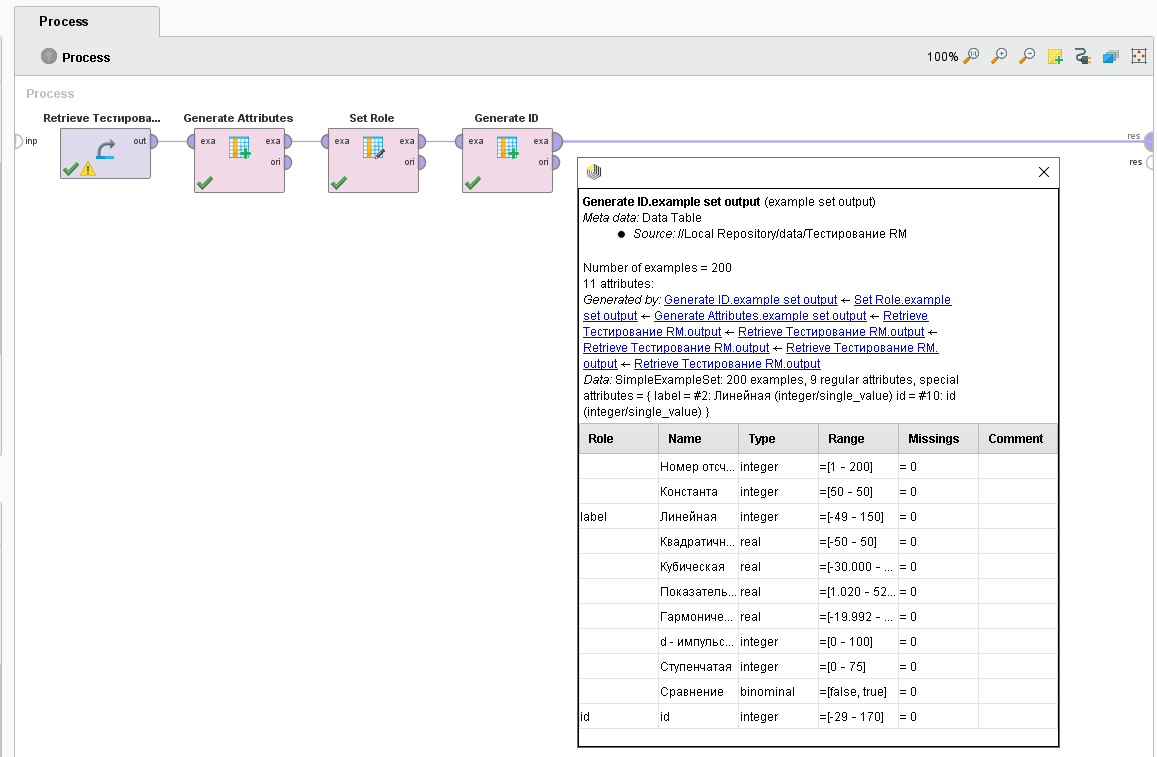


Рисунок 21 – Пример выполнения операций с ролями Таблица 6 – Настройка параметров процесса изменения ролей атрибутов

|  |  |  |
| --- | --- | --- |
| Оператор | Настройка | |
| Retrieve | Тестирование RM | |
| Generate Attributes | attribute name | Сравнение |
| function expression | [Линейная]<= -38 |

|  |  |  |
| --- | --- | --- |
| Оператор | Настройка | |
|  | keep all | true |
| Set Role | attribute name | Линейная |
| target role | label |
| set additional roles |  |
| Generate ID | create nominal ids | false |
| offset | false |

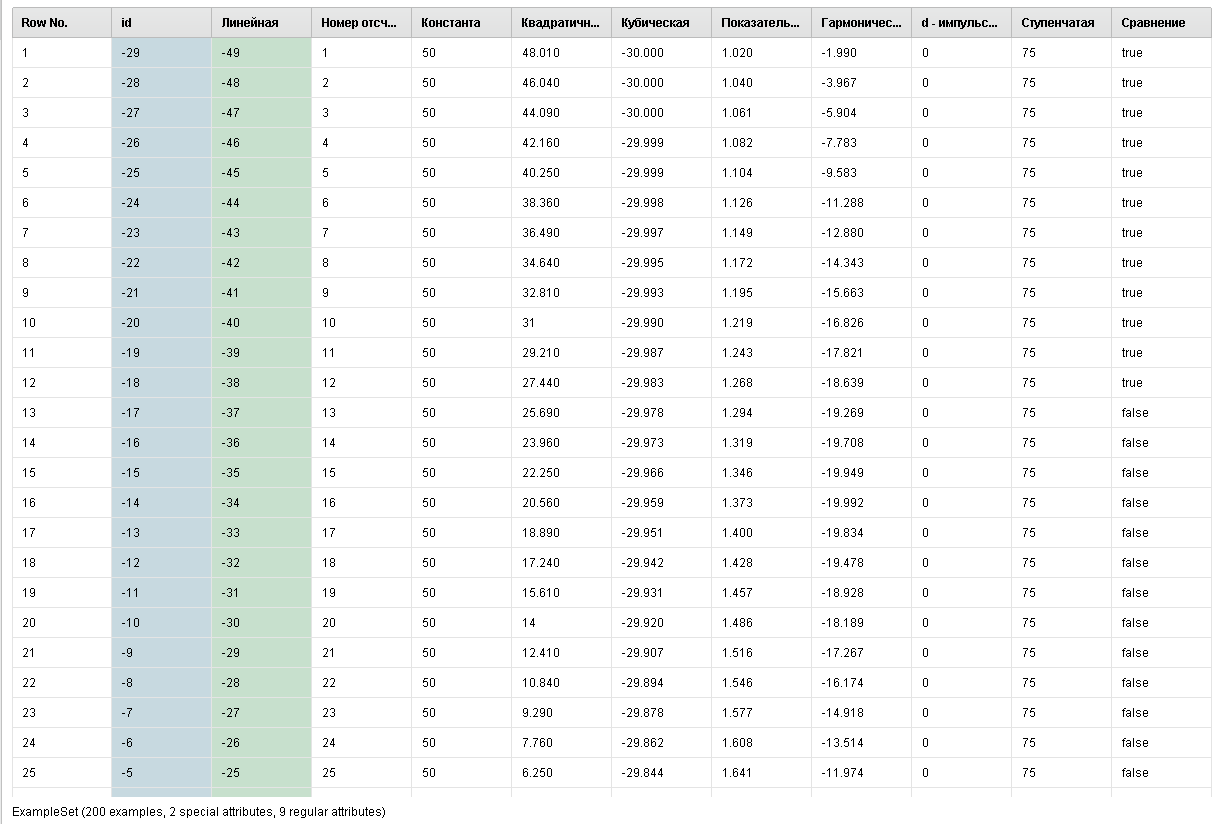


Рисунок 22 – Результаты работы процесса изменения ролей атрибутов

## Исходные данные

Исходными данными являются предлагаемые разработчиками пакета RapidMiner Studio образцы данных (Samples), а также индивидуальные данные, подготовленные студентом в процессе выполнения лабораторной работы №1.

## Порядок выполнения

1. Создайте процессы, преобразующие данные типа Числовые (numeric) в данные типа Текстовые (nominal) с избранными вами данными из папки Repository. Выполните обратные преобразования и убедитесь в корректности результатов.
2. Используя индивидуальные наборы данных, созданные в процессе выполнения лабораторной работы №1, выполните над ними различные варианты преобразований типов данных.

## Пример создания и подготовки индивидуального набора данных

В систему были импортированы данные об объектах наследия взятые с сайта мэрии города Уэлленд, Канада [5] (рисунок 23).

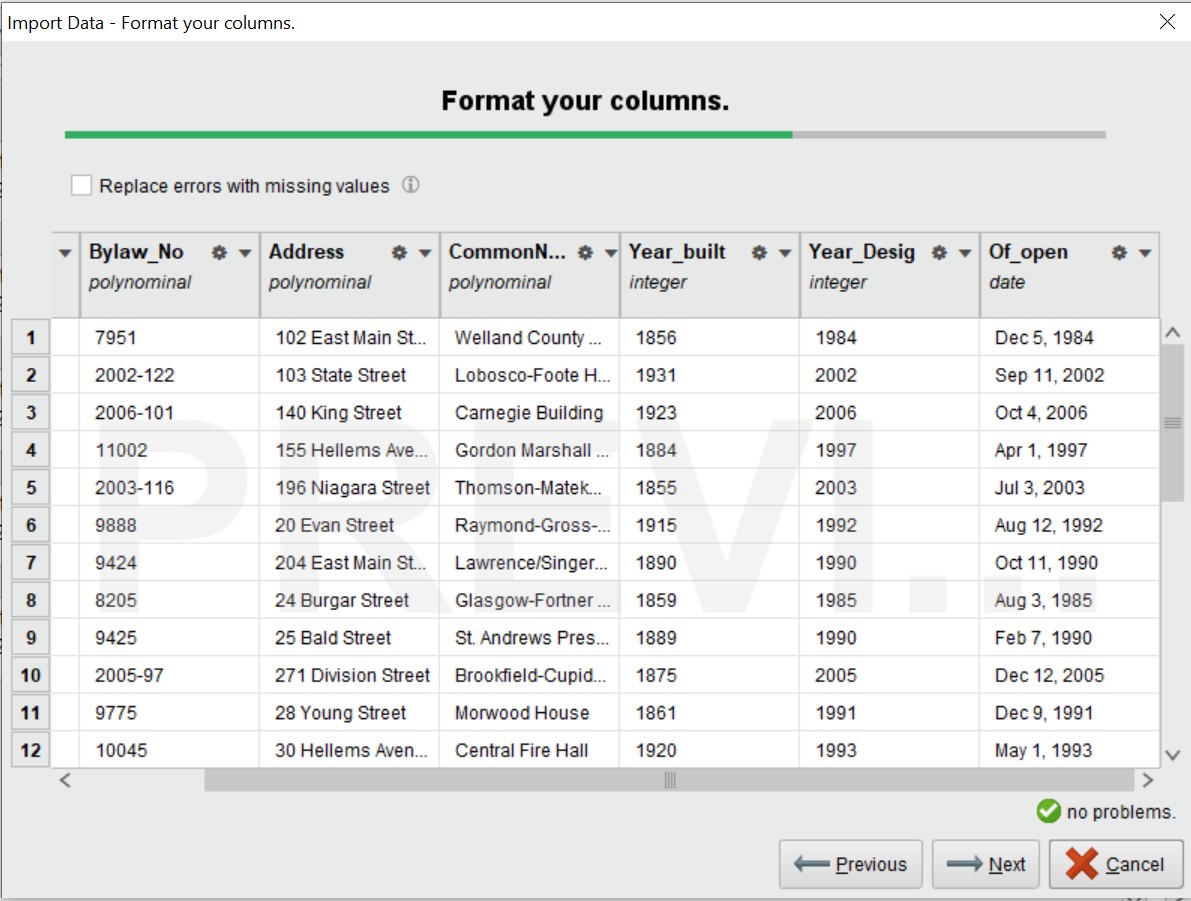


Рисунок 23 – Результаты импорта данных с сайта мэрии города Уэлленд, Канада

В столбцах таблицы содержится информация о почтовом индексе объекта, номере постановления, на основании которого он признан объектом наследия, адресе расположения, названии, годе постройки, годе реставрации и дате открытия объекта после реставрации. Для дальнейшего преобразования данных будем использовать атрибут с датой открытия после реставрации Of\_open с типом date.

*Преобразование Date to nominal*. Для преобразования типа данных в поле процесса был добавлен соответствующий оператор, установлены связи между входами и выходами, выбран атрибут, для которого будет изменен тип данных, а также формат даты, после чего произведен запуск процесса (рисунок 24). Результат преобразования данных показан на рисунке 25.

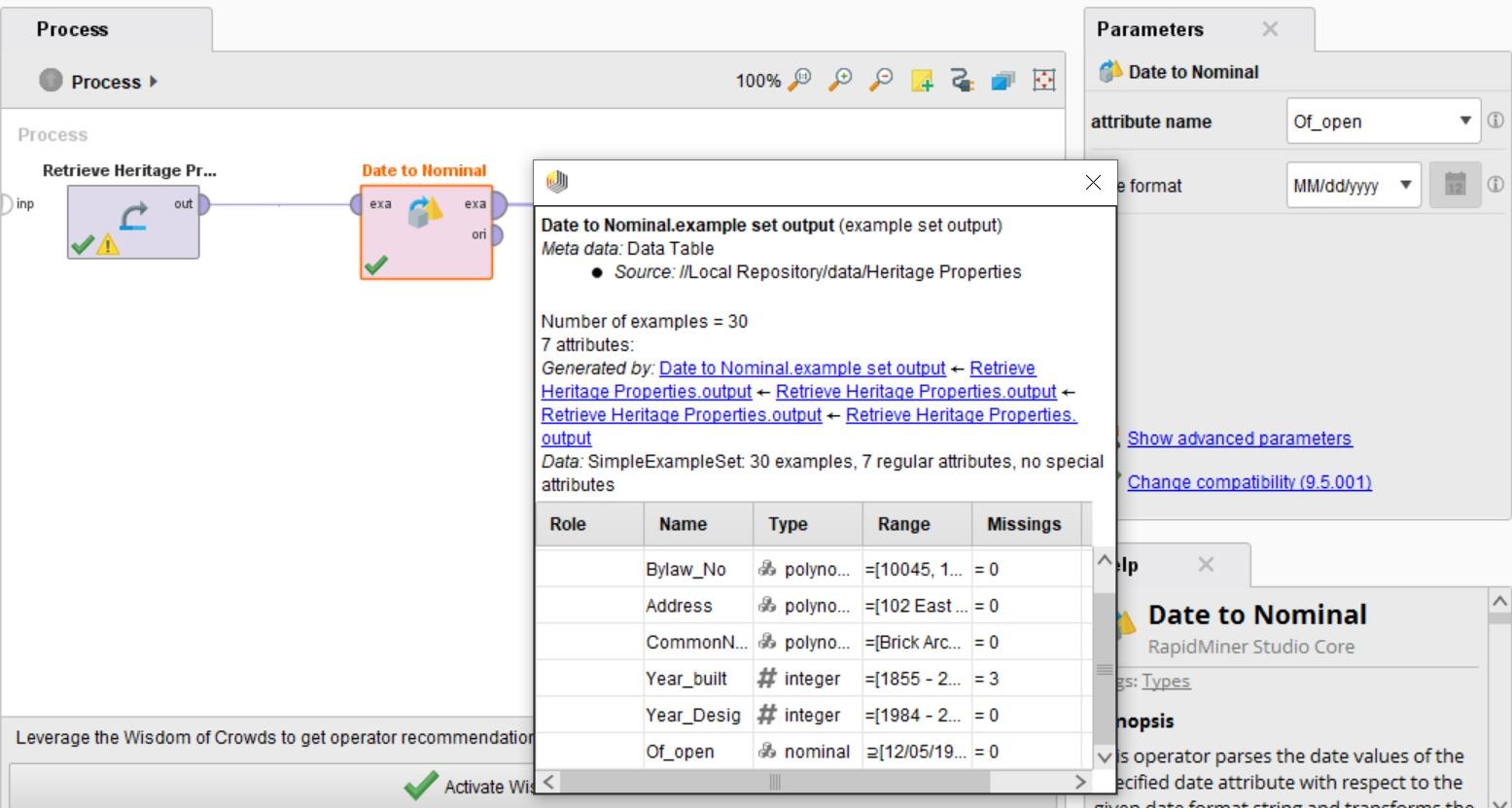


Рисунок 24 – Процесс преобразования Date to nominal

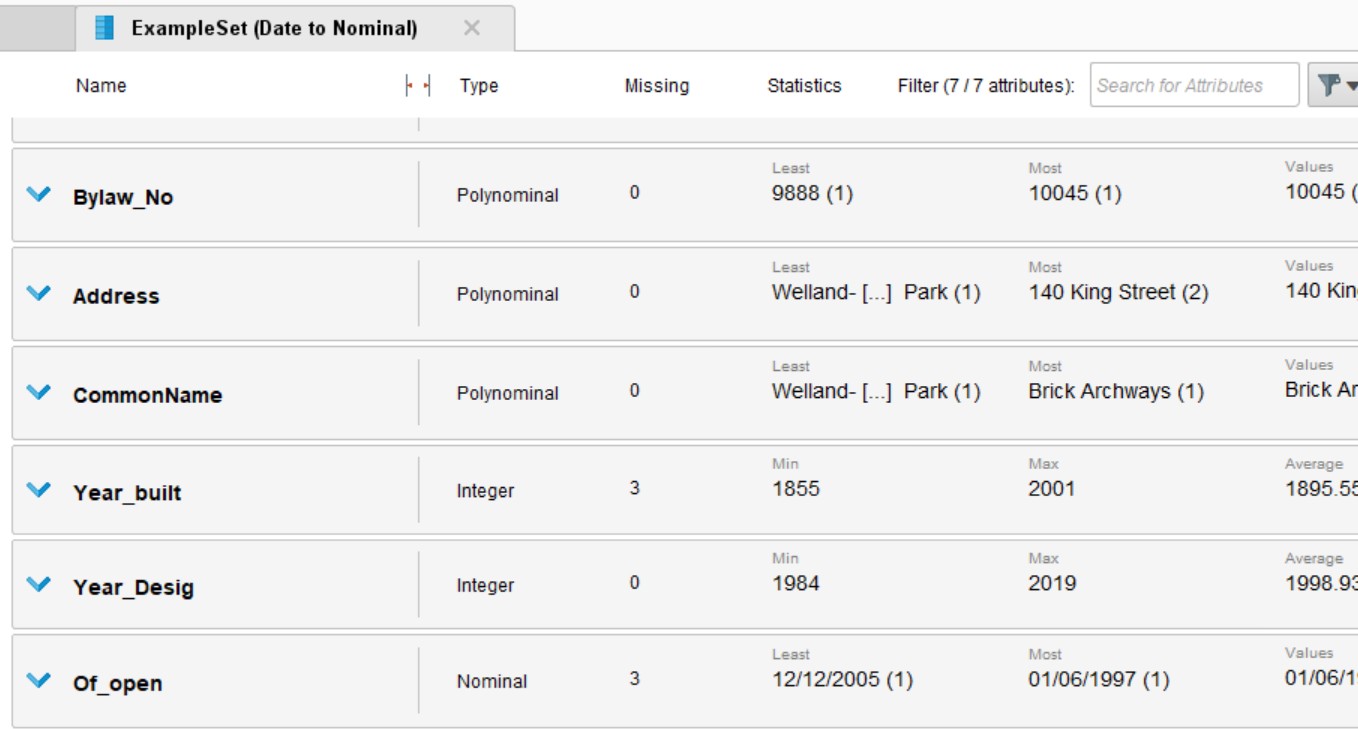
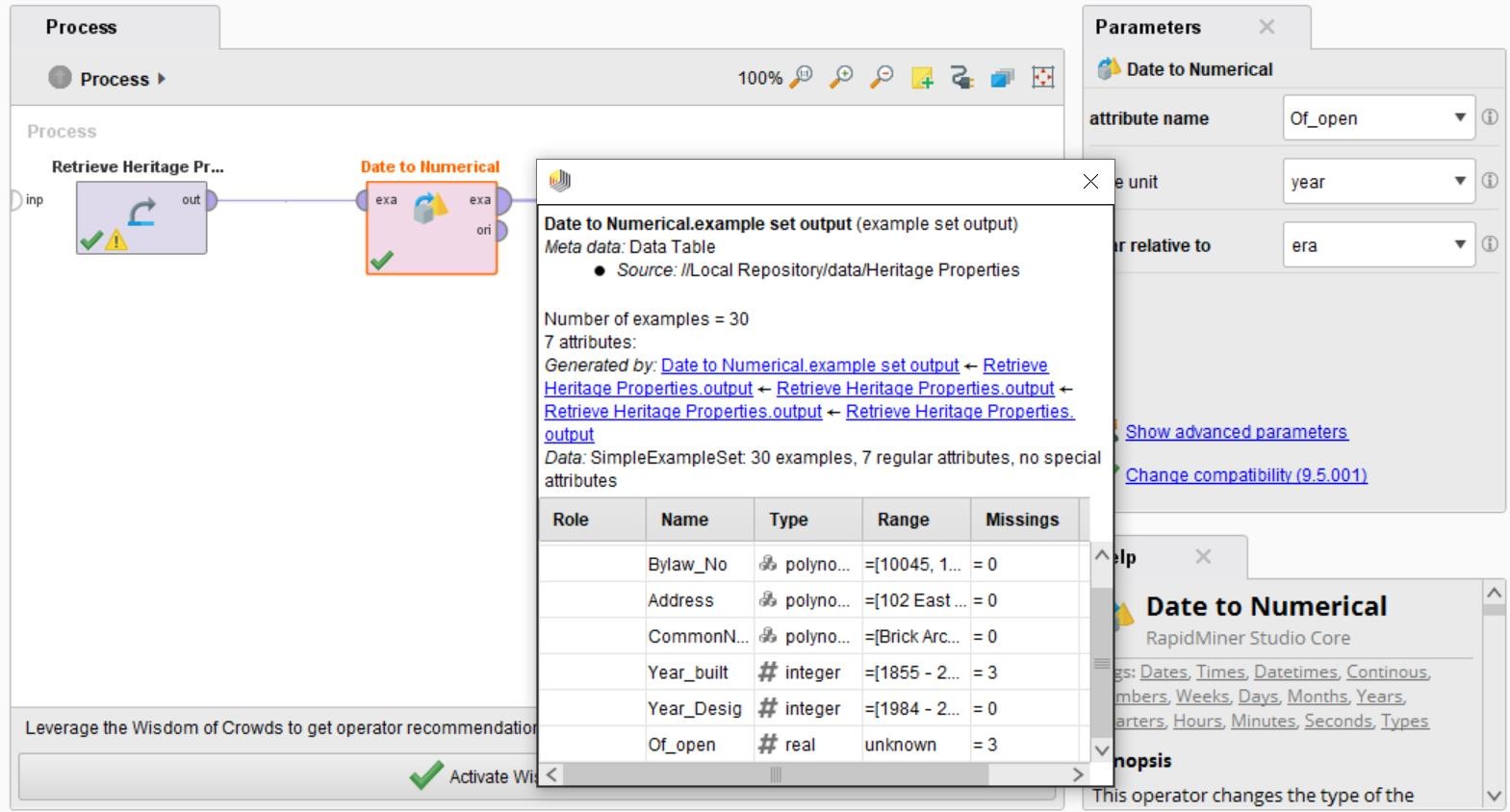


Рисунок 25 - Результат преобразования Date to nominal

*Преобразование Date to numerical*. Аналогичным образом было произведено преобразование данных к числовому формату. Процесс преобразования отображен на рисунке 26[Рисунок 26](#_bookmark18), а результат на рисунке 27.



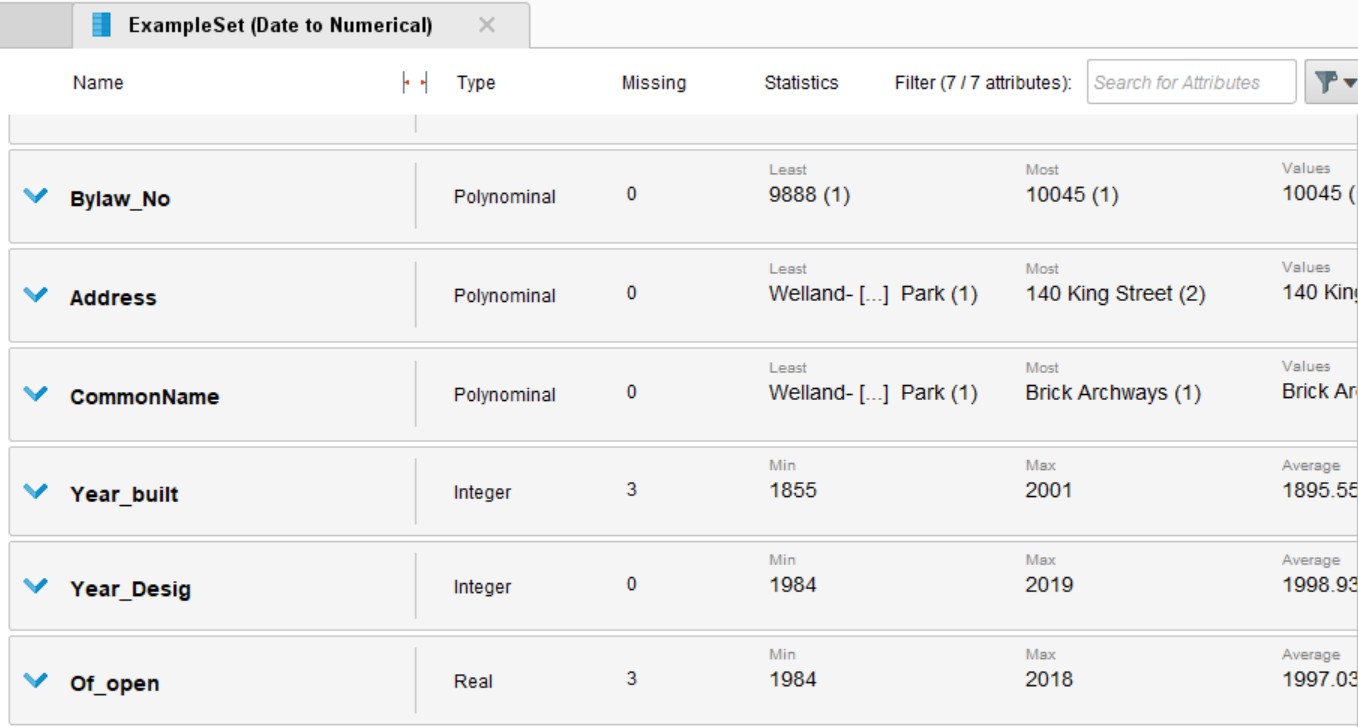
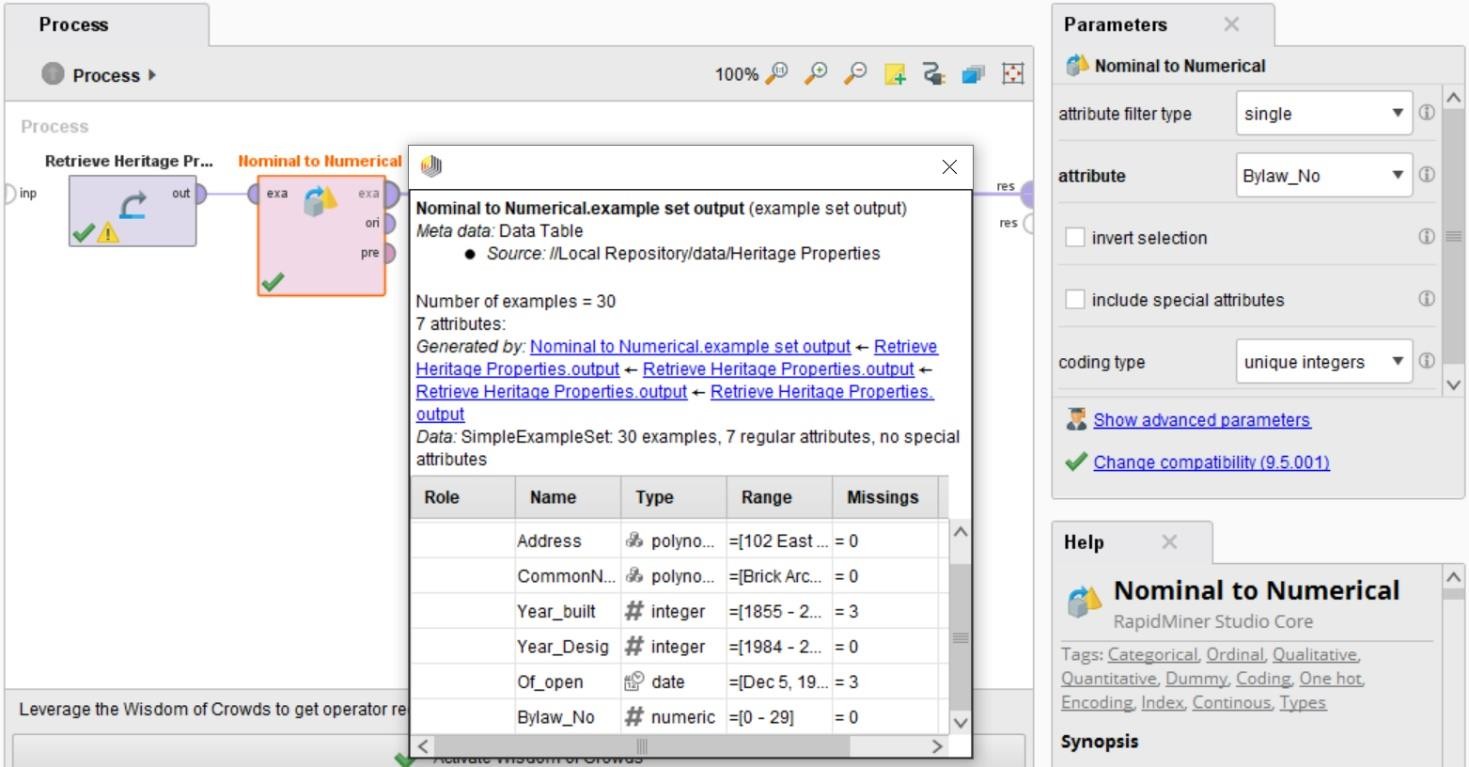
Рисунок 26 - Процесс преобразования Date to numerical

Рисунок 27 – Результат преобразования Date to numerical

*Преобразование Nominal to numerical*. Для преобразования текстовых данных в числовые в поле процесса был добавлен соответствующий оператор, установлены связи между входами и выходами, выбран атрибут (Bylaw\_No), для которого будет производиться преобразование, выбран тип кодирования unique integers, при котором каждому текстовому атрибуту будет присваиваться уникальное числовое значение. Процесс преобразования показан на рисунке 28, а полученный результат на рисунке 29.



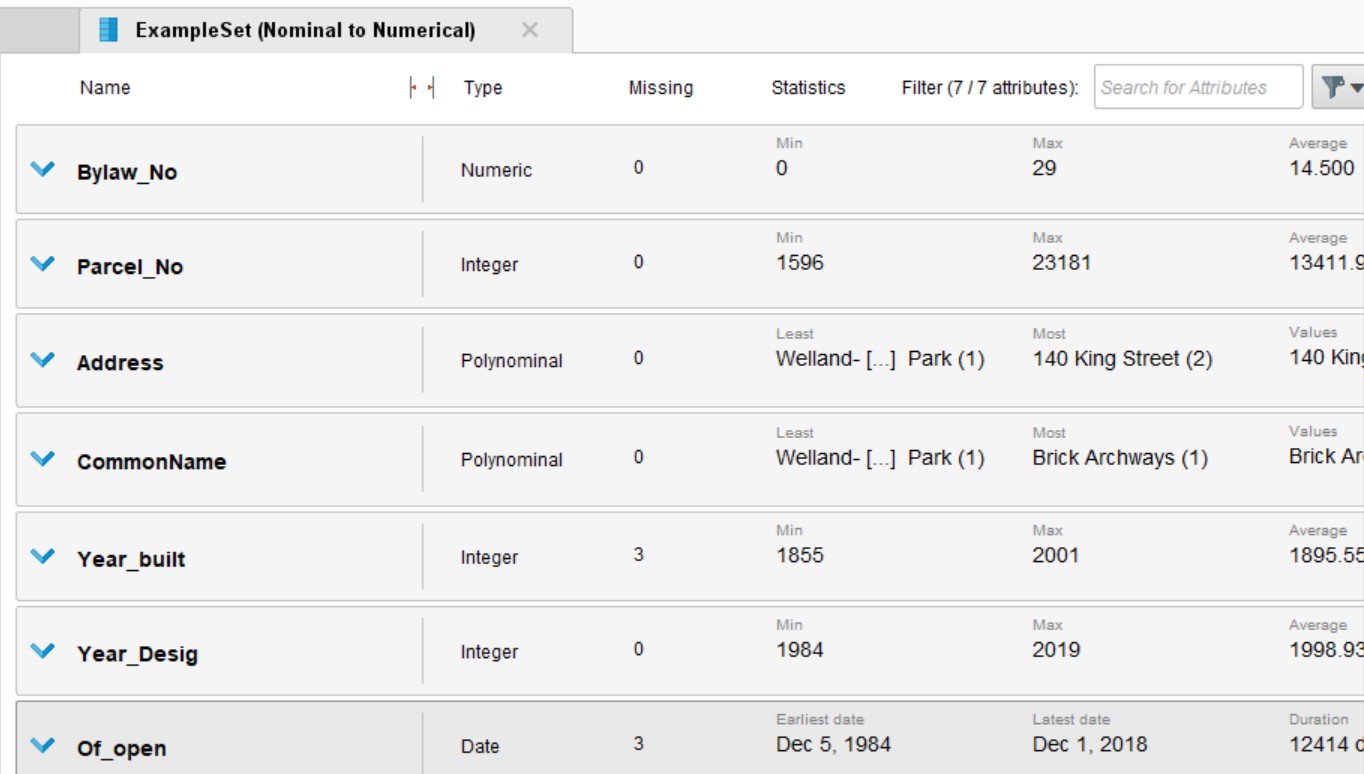
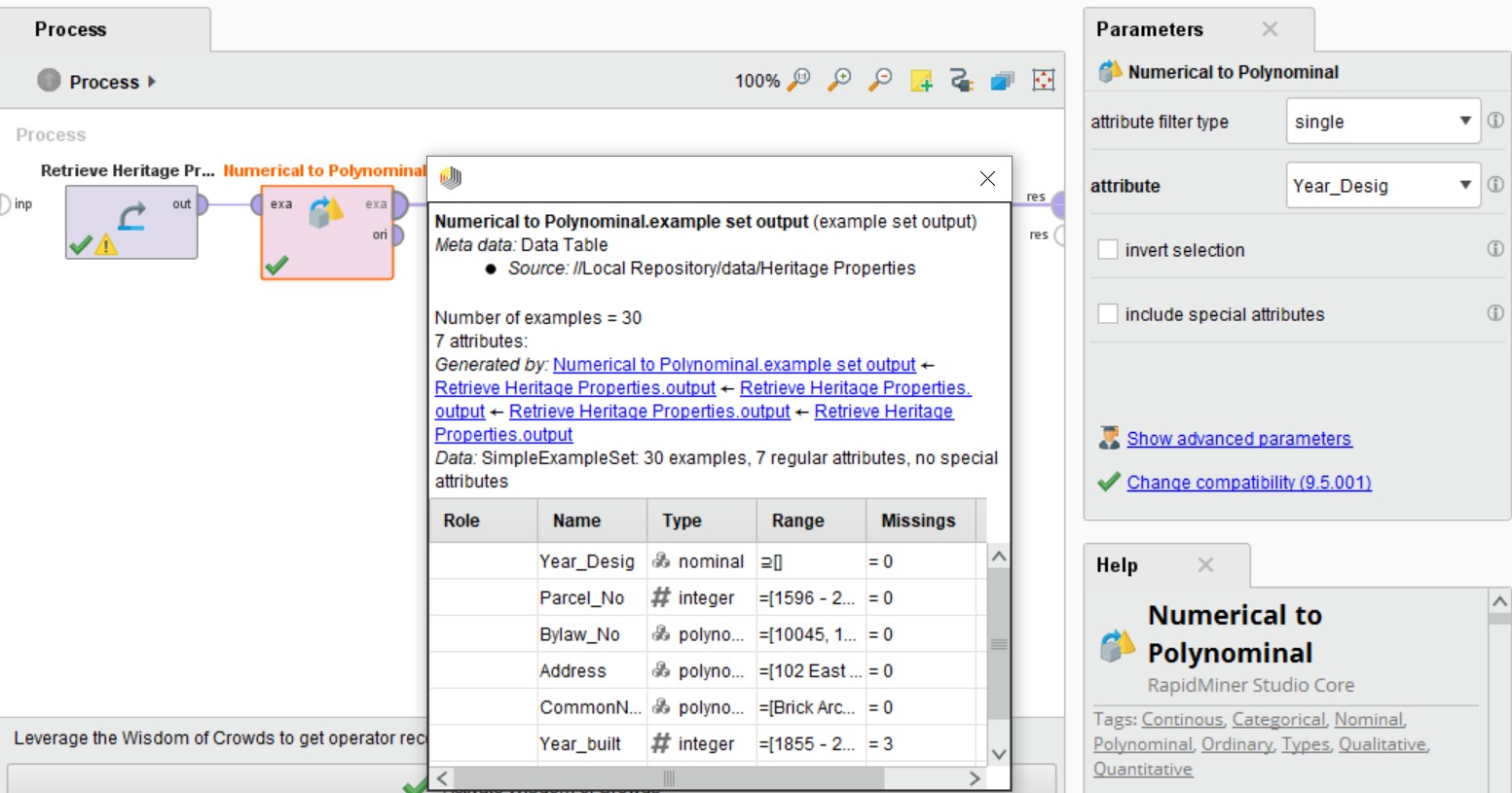
Рисунок 28 – Процесс преобразования Nominal to numerical

Рисунок 29 – Результат преобразования Nominal to numerical

*Преобразование Numerical to polynominal*. Для преобразования числовых данных в текстовые в поле процесса был добавлен соответствующий оператор, установлены связи между входами и выходами, выбран атрибут (Year\_Design), для которого будет производиться преобразование. Процесс преобразования показан на рисунке 30, [Рисунок 30](#_bookmark19)а полученный результат на рисунке 31. [Рисунок 31](#_bookmark20)



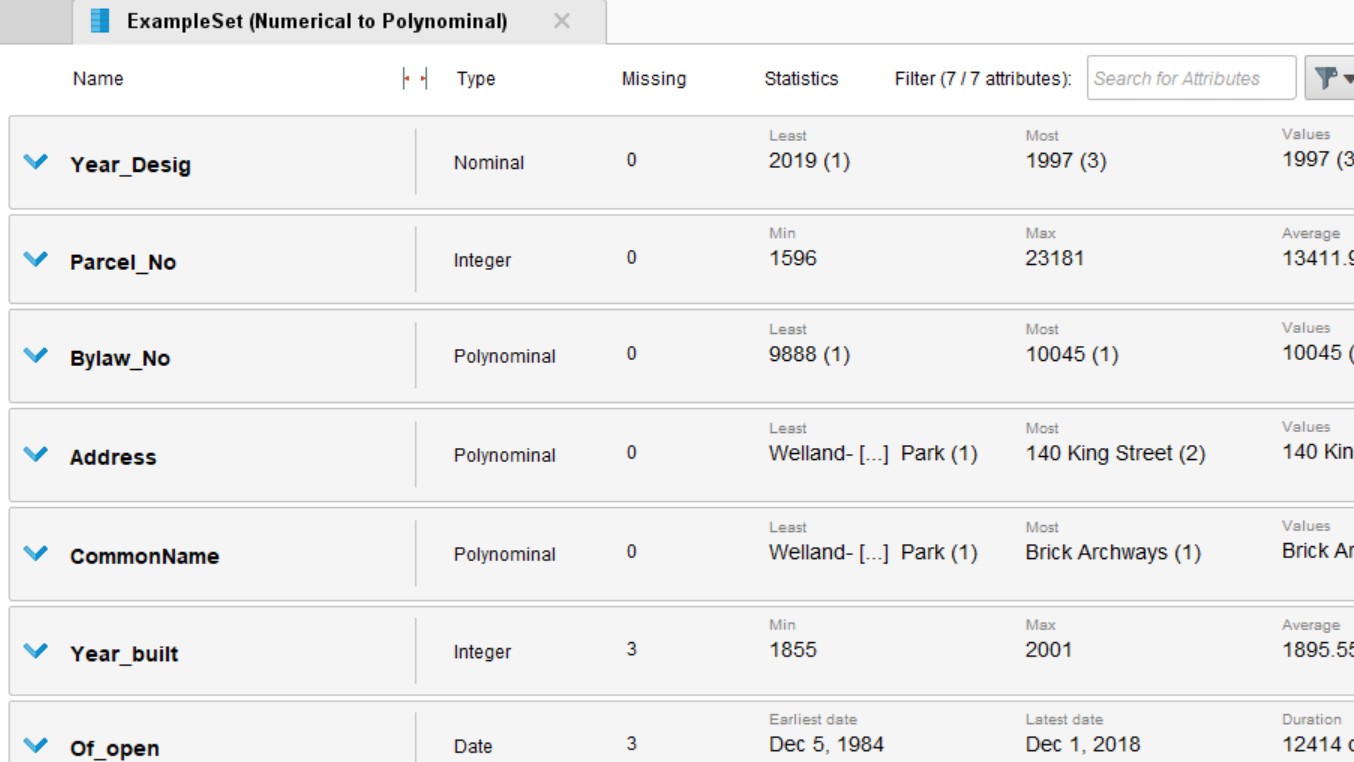
Рисунок 30 – Процесс преобразования Numerical to polynominal

Рисунок 31 – Результат преобразования Numerical to polynominal

## Оформление отчета

Отчет о выполнении лабораторной работы должен содержать титульный лист, перечень созданных за время выполнения лабораторной работы процессов и свои комментарии.

## Контрольные вопросы

1. Какие типы данных могут быть использованы в RapidMiner?
2. Какой смысл параметра Name?
3. Какой смысл параметра Type?
4. Какой смысл параметра Range?
5. Какой смысл параметра Missing?
6. Какой смысл параметра Role?
7. Какой смысл роли label?
8. Какой смысл роли id?
9. Какой смысл ролей prediction, cluster?
10. Какой смысл ролей weight, batch?