

Лабораторная работа
Разработка системы идентификации с помощью пакета Orange

Выполнила: Короткова Инга Сергеевна

2020 год

Цель работы – закрепить навыки проектирования интеллектуальных моделей с помощью платформы Orange Data Mining для задач построения идентифицирующих систем и оценки эффективности их работы.

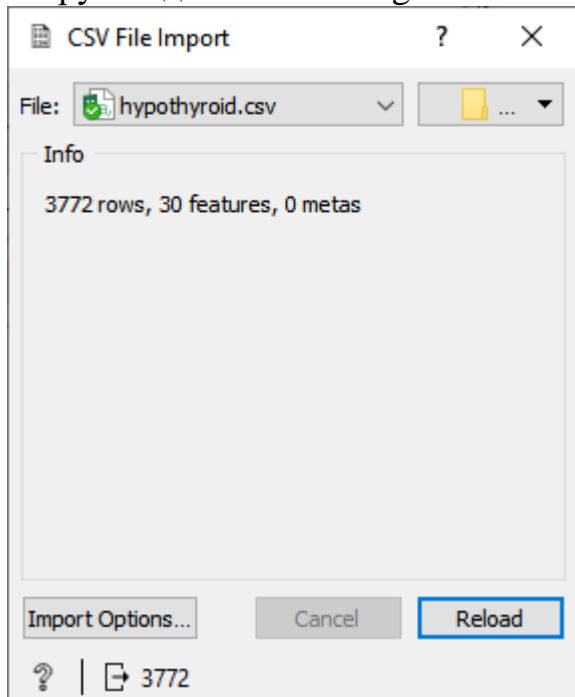
Задачи:

1. Выбрать набор данных, содержащий не менее 500 объектов и не менее 10 атрибутов, отмеченный как набор, для задачи классификации или регрессии с репозитория <https://archive.ics.uci.edu/ml/datasets.php>
2. Загрузить выбранный набор данных.
3. Произвести исследовательский анализ данных:
 - получить объём исследуемых данных;
 - получить число атрибутов и их типы данных;
 - посмотреть распределение числа примеров классов
 - если необходимо, выполнить преобразование категориальных атрибутов;
 - если необходимо, заполнить пропущенные значения в выборке.
4. Разделить набор данных на обучающую и тестовую выборку и объяснить это разбиение.
5. Обучить несколько моделей с помощью трех любых алгоритмов построения классификаторов или регрессоров (по выбору слушателя курса), отметить почему были выбраны эти алгоритмы
6. Оценить эффективность моделей на тестовой выборке с помощью матрицы неточностей, критериев полноты *Recall* и точности *Precision*, в случае создания классификатора, или критериев средняя квадратичная ошибка *MSE*, средняя абсолютная ошибка *MAE* и коэффициент детерминации *R2*.

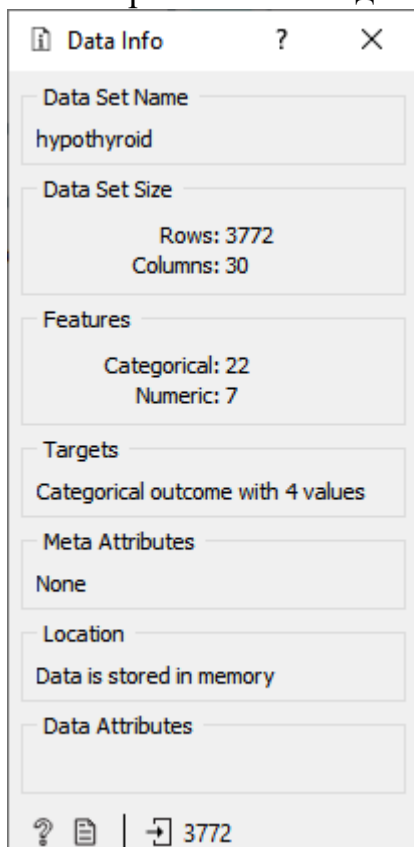
В этой работе будем использовать датасет по заболеваниям щитовидной железы.

Датасет взят с сайта: <http://archive.ics.uci.edu/ml/datasets/thyroid+disease>
Данные собраны Garvan Institute в Австралии

Загрузим данные в Orange.



Посмотрим ближе на данные:



По итогу есть 3772 записи и 30 колонок.

Признаки распределены следующим образом: 7 числовых и 23 категориальных.

Общий вид на данные, при помощи инструмента Data Table.

	Class	age	sex	on_thyroxine	query_on_thyroxine	antithyroid_medication	sick	pregnant	thyroid_surgery	l131_treatment
1	negative	41	F	f	f	f	f	f	f	f
2	negative	23	F	f	f	f	f	f	f	f
3	negative	46	M	f	f	f	f	f	f	f
4	negative	70	F	t	f	f	f	f	f	f
5	negative	70	F	f	f	f	f	f	f	f
6	negative	18	F	t	f	f	f	f	f	f
7	negative	59	F	f	f	f	f	f	f	f
8	negative	80	F	f	f	f	f	f	f	f
9	negative	66	F	f	f	f	f	f	f	f
10	negative	68	M	f	f	f	f	f	f	f
11	negative	84	F	f	f	f	f	f	f	f
12	negative	67	F	t	f	f	f	f	f	f
13	negative	71	F	f	f	f	t	f	f	f
14	negative	59	F	f	f	f	f	f	f	f
15	negative	28	M	f	f	f	f	f	f	f
16	compensated...	65	F	f	f	f	f	f	f	f
17	negative	42	?	f	f	f	f	f	f	f
18	negative	63	F	f	f	f	f	f	f	f

Отбросим часть данных, которые либо же являются по сути дубликатами уже существующих значений, либо же полностью отсутствуют.

Select Columns

Available Variables

Filter

- TBG_measured
- TBG
- T4U_measured
- TT4_measured
- T3_measured
- FTI_measured

Features

Filter

- age
- sex
- on_thyroxine
- query_on_thyroxine
- on_antithyroid_medication
- sick
- pregnant
- thyroid_surgery
- l131_treatment
- query_hypothyroid

Target Variable

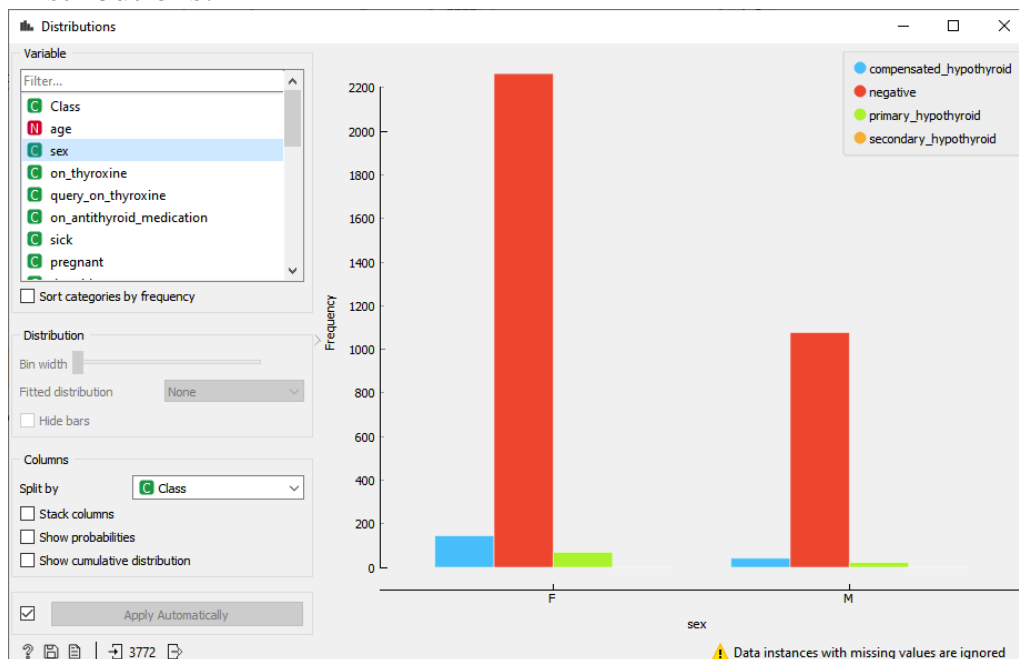
Class

Meta Attributes

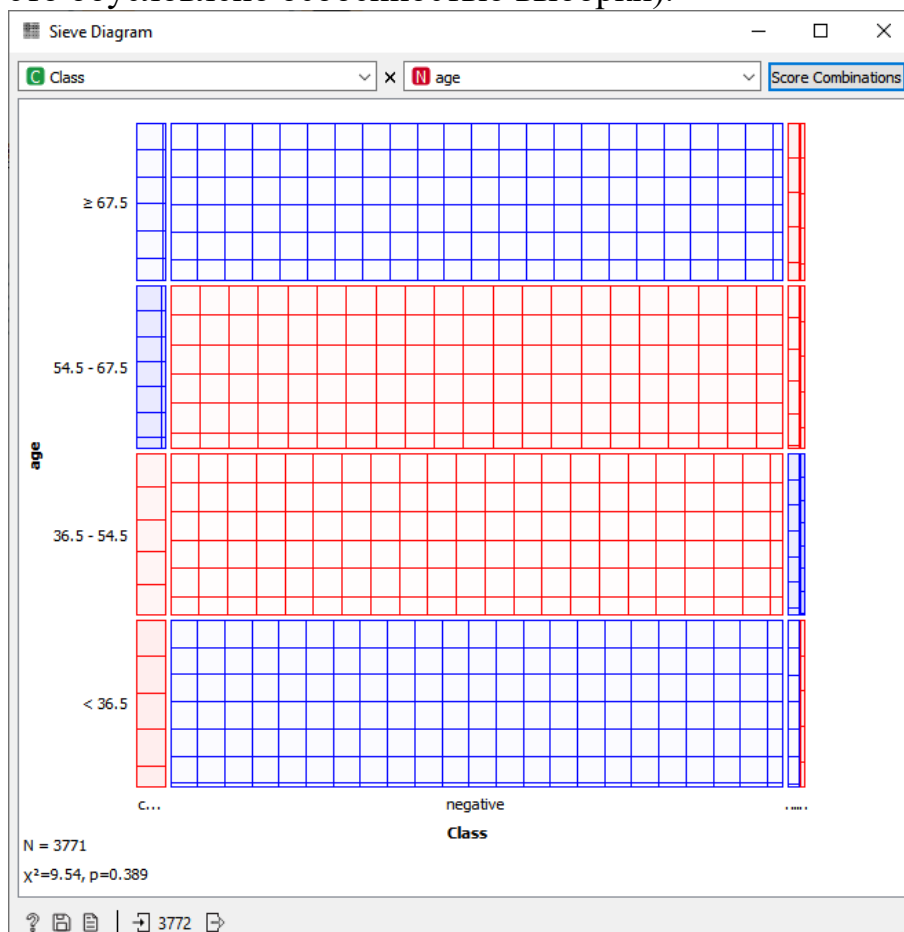
Reset

Send Selection

Продолжим обзорное исследование данных при помощи инструмента Distributions:

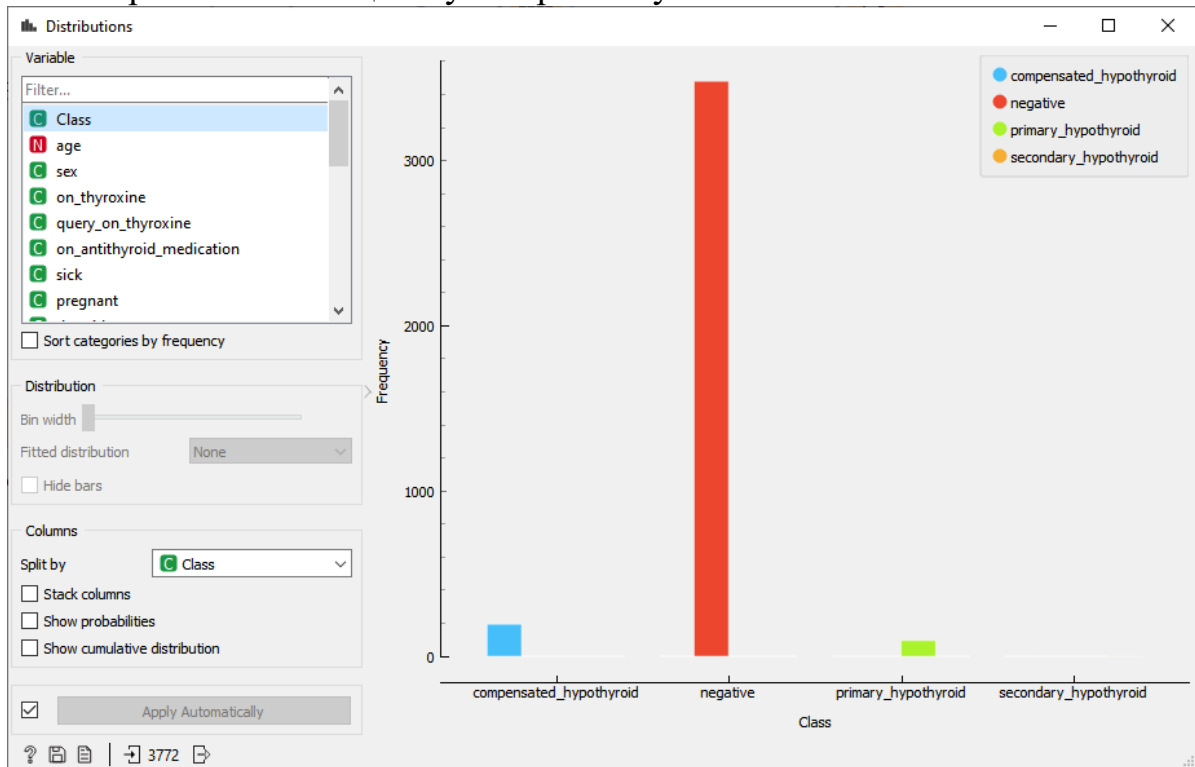


Как видно, в выборке больше женщин и у они чаще болеют (скорее всего это обусловлено особенностью выборки).



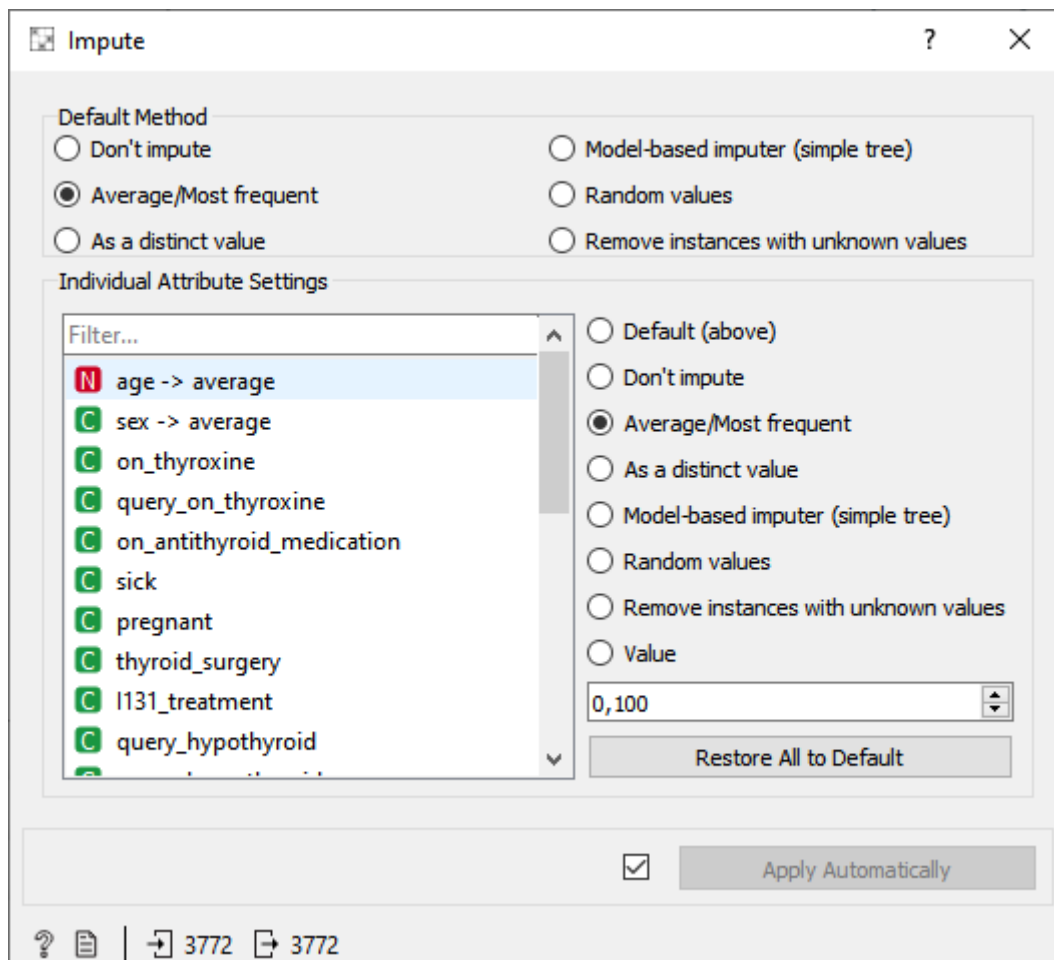
Как видно, большую часть выборки составляют люди старшего возраста и заболевание имеет более “возрастной” характер.

Посмотрим ближе на целевую переменную:

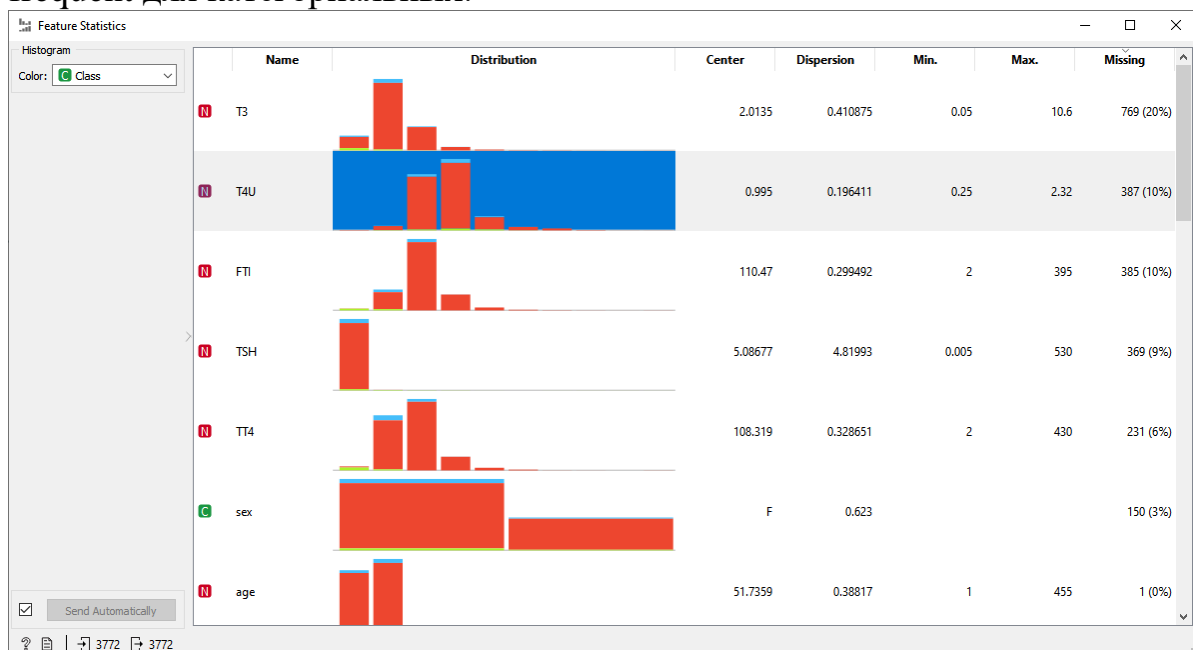


Целевая переменная Class состоит из следующих значений: ('negative', 'compensated_hypothyroid', 'primary_hypothyroid', 'secondary_hypothyroid'). Причем, количество объектов класса 'secondary_hypothyroid' очень мало.

Заполним пропуски при помощи блока Impute.

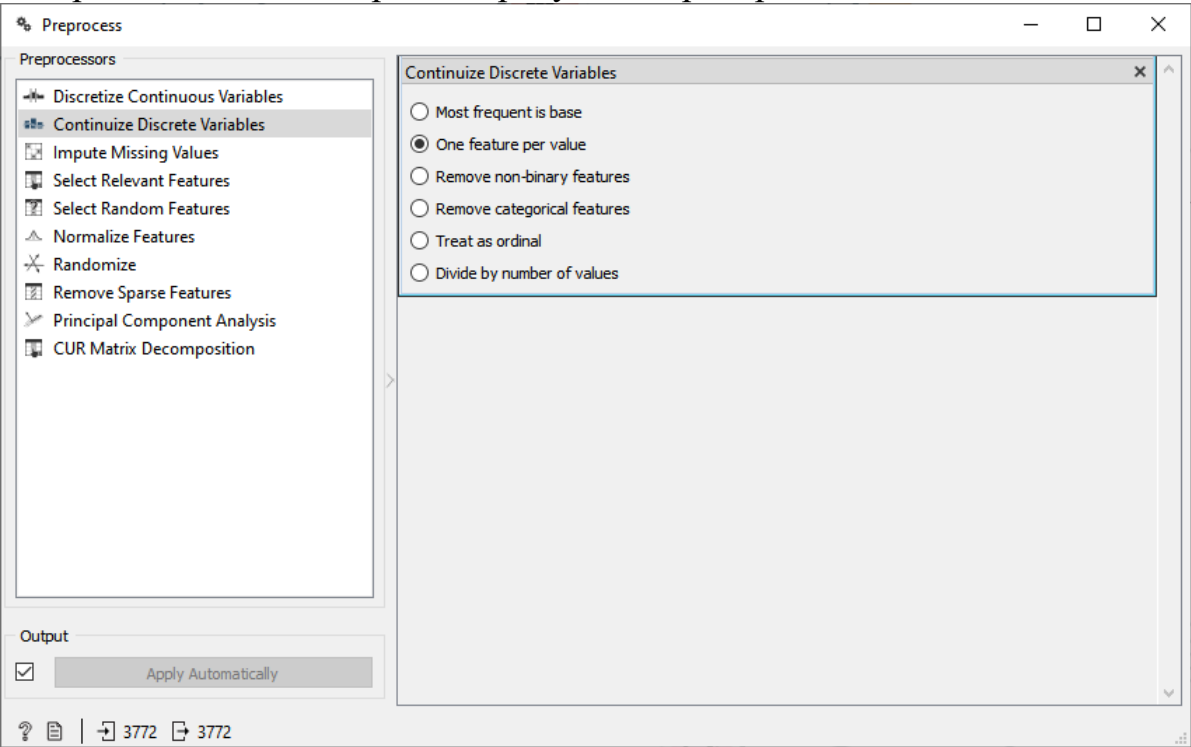


Заполнять будем средним (для численных признаков), либо же most frequent для категориальных.



Всего потребовалось заполнять пропуски в 7 признаках: T3, T4U, FTI, TSH, TT4, Age, Sex.

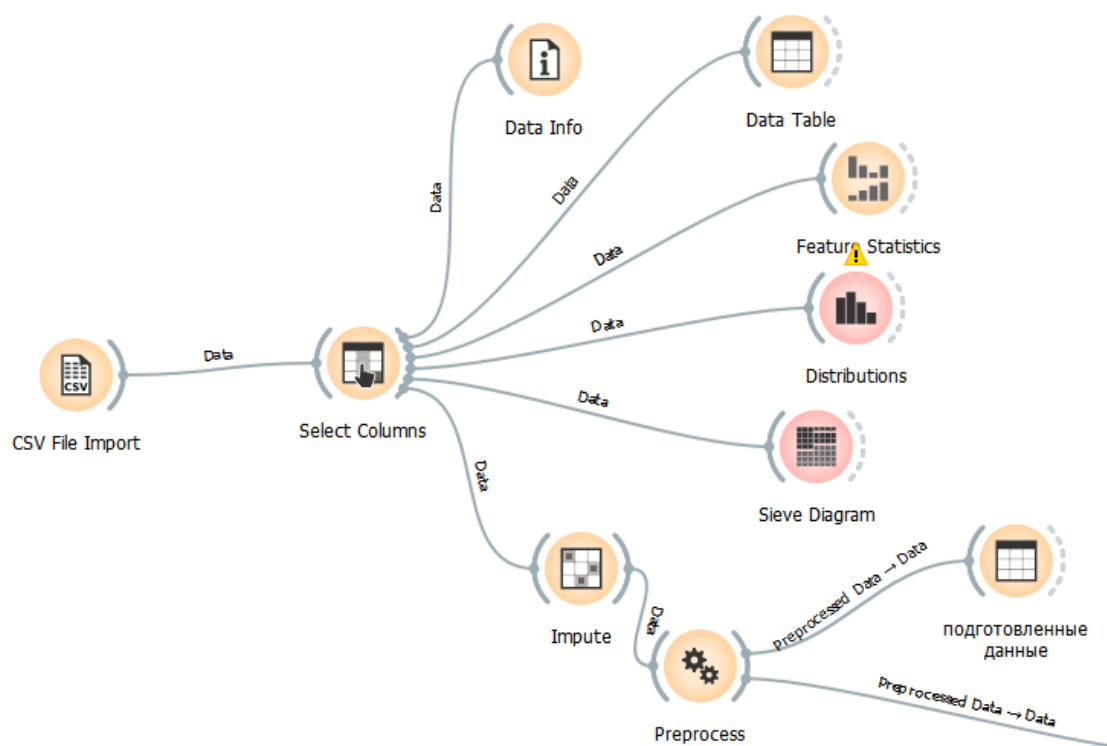
Теперь остается кодировать категориальные переменные.
Для этого воспользуемся блоком Preprocess -> One Feature per value,
которая позволяет совершить требуемые преобразования.



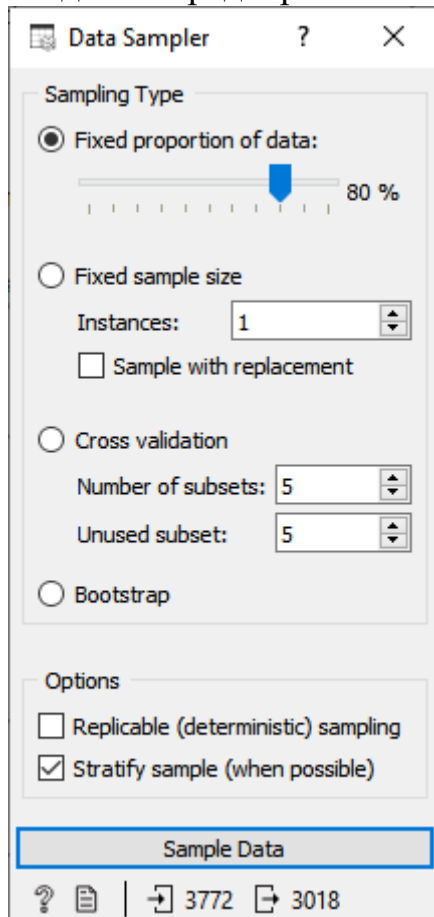
Получившийся результат, после One Hot Encoding, видно, что колонка Sex, теперь разделилась на 2, где 1 указан нужный пол.

подготовленные данные												
Info												
3772 instances (no missing data)												
43 features												
Target with 4 values												
No meta attributes												
Variables												
<input checked="" type="checkbox"/> Show variable labels (if present)												
<input type="checkbox"/> Visualize numeric values												
<input checked="" type="checkbox"/> Color by instance classes												
Selection												
<input checked="" type="checkbox"/> Select full rows												
Restore Original Order												
<input checked="" type="checkbox"/> Send Automatically												
Class	age	sex=F	sex=M	on_thyroxine=f	on_thyroxine=t	ery_on_thyroxine	ery_on_thyroxine	tithyroid_medica	tithyroid_medica	tic		
1 negative	41	1	0	1	0	1	0	1	0	0		
2 negative	23	1	0	1	0	1	0	1	0	0		
3 negative	46	0	1	1	0	1	0	1	0	0		
4 negative	70	1	0	0	1	1	0	1	0	0		
5 negative	70	1	0	1	0	1	0	1	0	0		
6 negative	18	1	0	0	1	1	0	1	0	0		
7 negative	59	1	0	1	0	1	0	1	0	0		
8 negative	80	1	0	1	0	1	0	1	0	0		
9 negative	66	1	0	1	0	1	0	1	0	0		
10 negative	68	0	1	1	0	1	0	1	0	0		
11 negative	84	1	0	1	0	1	0	1	0	0		
12 negative	67	1	0	0	1	1	0	1	0	0		
13 negative	71	1	0	1	0	1	0	1	0	0		
14 negative	59	1	0	1	0	1	0	1	0	0		
15 negative	28	0	1	1	0	1	0	1	0	0		
16 compensated...	65	1	0	1	0	1	0	1	0	0		
17 negative	42	1	0	1	0	1	0	1	0	0		
18 negative	63	1	0	1	0	1	0	1	0	0		

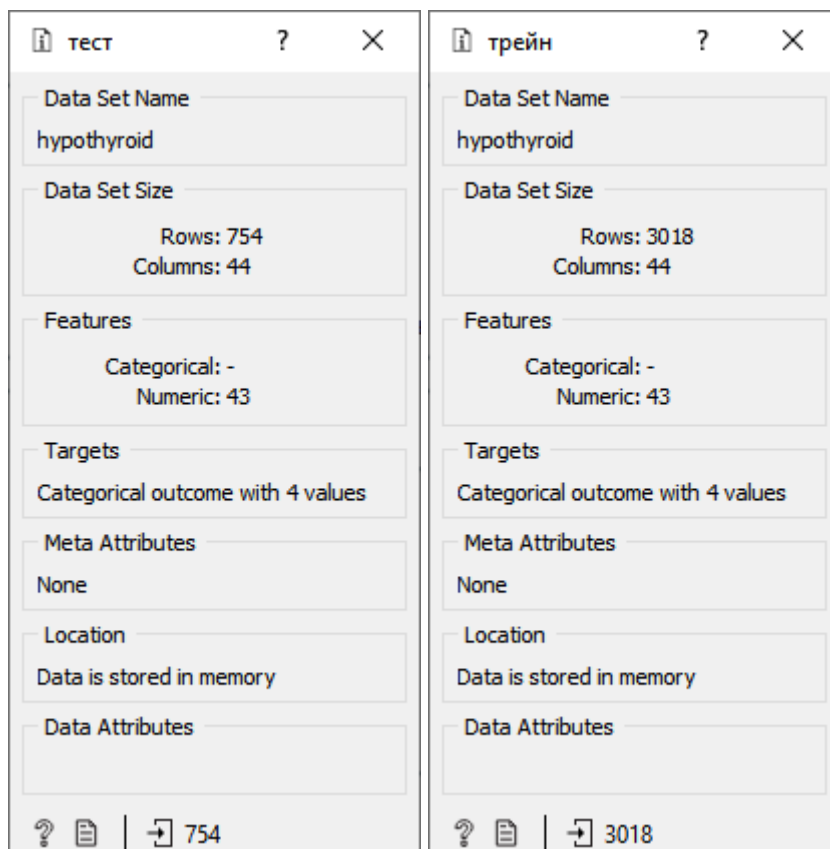
Общий вид dashboard'a Orange перед разделением данных.



Разделим предобработанные данные на обучающую и тестовую выборки.



Данные делим в соотношении 80% / 20 %, где на 80% придется обучающий набор данных, а проверять будем на оставшихся 20%. Такое соотношение является достаточно распространённым и позволит обучить и проверить работу алгоритмов машинного обучения, также не стоит забывать про стратификацию, для того, чтобы пропорция целевых переменных в подвыборках была сохранена.

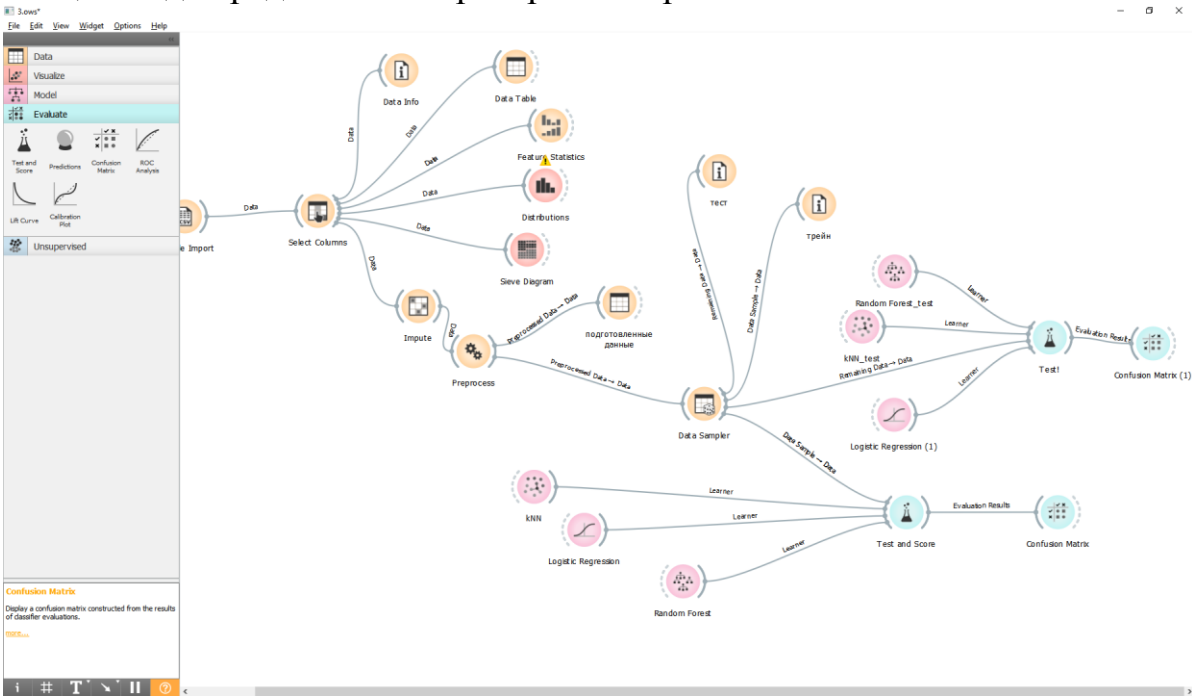


Как видно, разбиение произошло нужным образом.

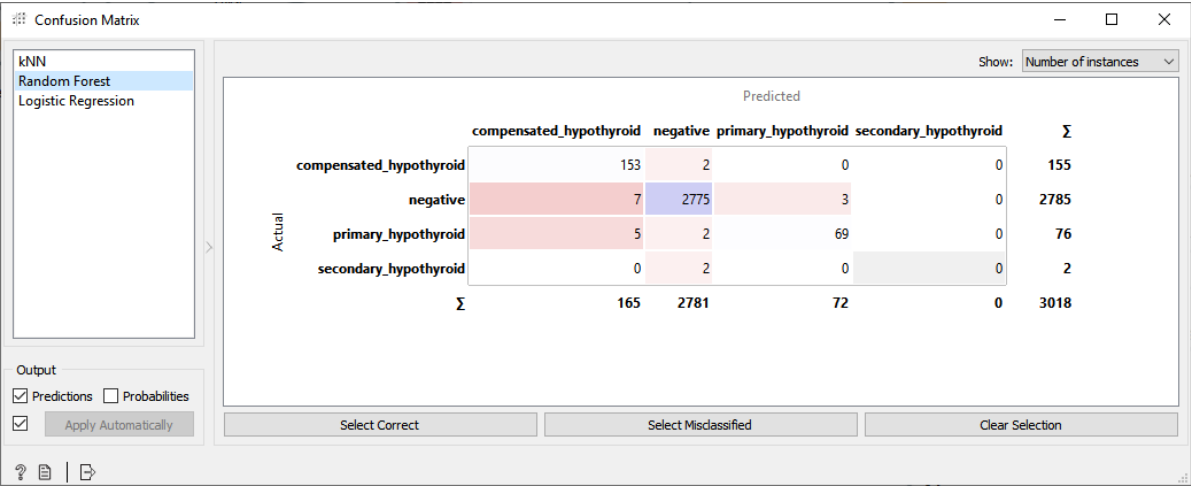
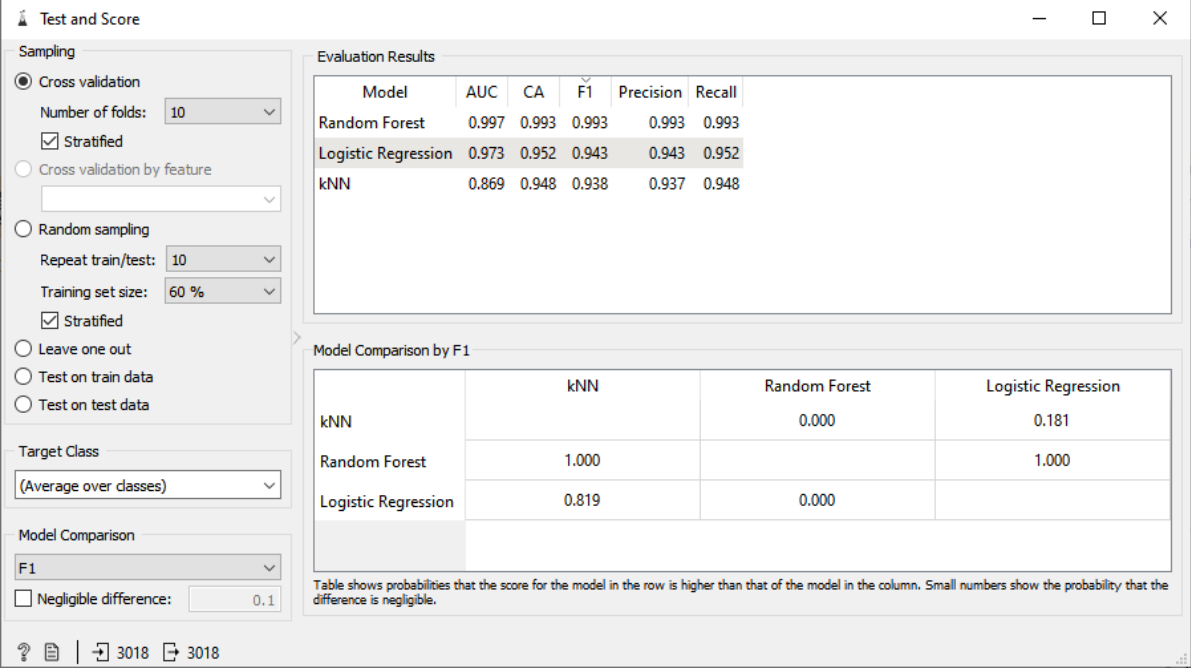
Обучим несколько моделей с помощью трех алгоритмов построения. В качестве алгоритмов были выбраны:

- Алгоритм градиентного бустинга был выбран в силу того, что он сочетает в себе скорость работы и содержит простую, но эффективную идею того, что ансамбль слабых моделей в совокупности дает хороший результат.
- Алгоритм k-ближайших соседей.
Он был выбран потому, что в основе его лежит простая гипотеза о том, что можно судить о классе объекта (исходя из выбранной метрики) по его соседям.
- Логистическая регрессия, хороший и быстрый алгоритм, который хорошо подходит для быстрого получения результатов.

Общий вид перед началом проверки алгоритмов:

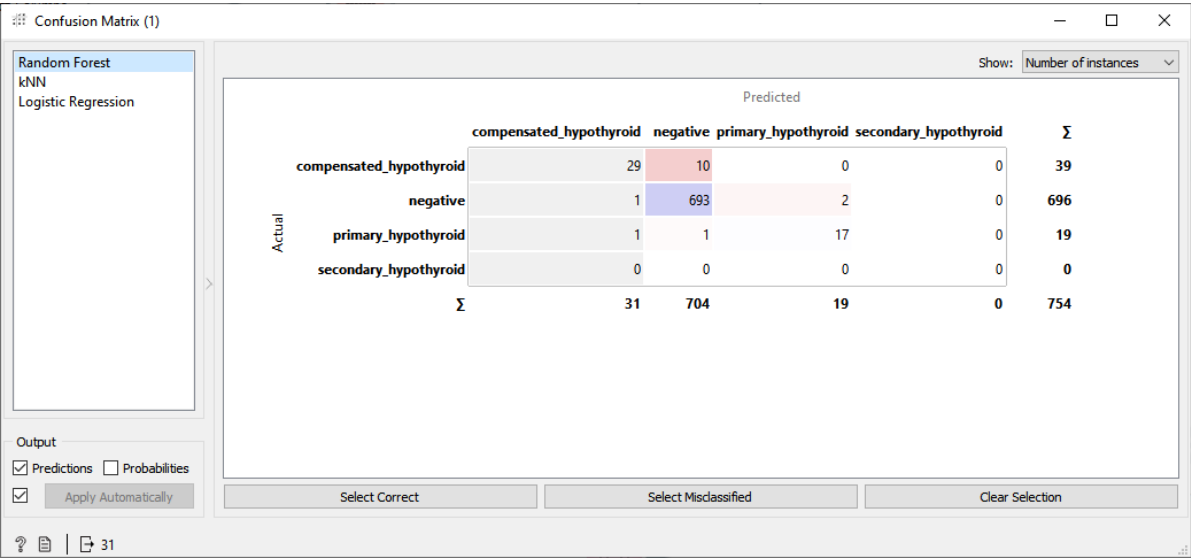
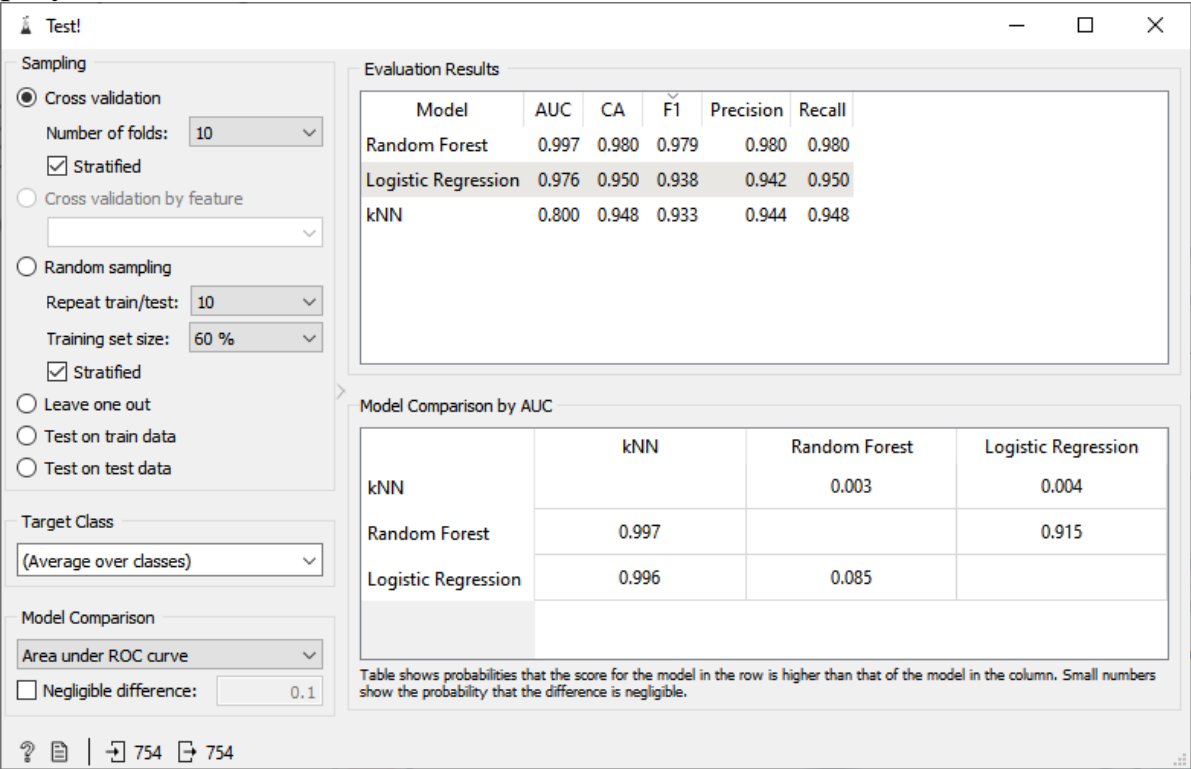


Оценим эффективность моделей на тестовой выборке с помощью матрицы неточностей, критериев полноты Recall и точности Precision. Для обучающей выборки результаты были получены следующие результаты:



Confusion Matrix для RF.

Для тестовой выборки результаты были получение следующие результаты:



Confusion Matrix для RF.

Заключение.

В данной работе мы провели: работу с датасетом, а именно:

- Загрузили данные в Orange
- Произвели исследовательский анализ данных:
 - получили объём исследуемых данных
 - получили число атрибутов и их типы данных
 - посмотреть распределение числа примеров классов
 - провели исследование других признаков
 - выполнили преобразование категориальных атрибутов
 - заполнили пропущенные значения в выборке
 - отобрали часть данных
- Разделили данные на обучающую и тестовую выборки
- Обучили 3 модели-классификатора машинного обучения
- Оценили эффективность моделей на тестовой выборке с помощью матрицы неточностей, критериев полноты *Recall* и точности *Precision* и гармоничной меры F1 Score.

Среди сравненных классификаторов и параметров наилучшими оценками обладают (в порядке убывания) по метрике F1 Score (которая по сути является гармоничной мерой, которая совмещает в себе *precision* и *recall*):

- Градиентный бустинг на деревьях принятия решений
- Логистическая регрессия
- KNN