АНАЛИЗ И ВИЗУАЛИЗАЦИЯ ДАННЫХ С ПОМОЩЬЮ R

А. В. Пашук, А. Б. Гуринович
Кафедра информатики, кафедра вычислительных методов и программирования, Белорусский государственный университет информатики и радиоэлектороники
Минск, Республика Беларусь
E-mail: aliaksandr.pashuk@gmail.com, gurinovich@bsuir.by

В данной работе рассматривается пример использования языка программирования R для анализа и визуализации больших объемов данных на примере распознавания спам-писем с использованием модели, обученной методом Random forest.

Введение

R - это бесплатный язык программирование и программная среда для статистических вычислений и визуализации. Язык широко используется статистиками и учеными, работающих с большими объемами данных для разработки приложений сбора, преобразования и анализа данных. Среди основных преимуществ стоит выделить следующие: разработан и оптимизирован для обработки больших объемов данных; имеет множество пакетов, поддерживаемых большим сообществом; бесплатный; легко интегрируется со сторонними приложениями.

Показать достоинства технологии удобнее всего на примере решения практической задачи, в данном случае, на обучении модели определять спам-письма. В качестве массива данных для обучения и проверки работы будет использоваться уже подготовленный датасет, содержащий информацию о нескольких тысячах писем [1].

I. Анализ исходных данных

Авторами данного источника уже проведена начальная обработка и чистка данных, что упрощает задачу. Данный датасет содержит следующую информацию:

- word_freq_WORD (48 столбцов) – процент слов в письме, совпадающих с словом WORD, другими словами, 100 * (количество раз, когда слово WORD употребляется в письмеl) / общее количество слов в письме.
- char_freq_CHAR] (6 столбцов) – процент символов в письме, сопадающих с символов CHAR, другими словами, 100 * (количество символов CHAR) / общее число символов в письме.
- capital_run_length_average (1 столбец) – средняя длина непрерывающейся последовательности заглавных букв в письме.
- capital_run_length_longest (1 столбец) – длина наибольшей непрерывающейся последовательности заглавных букв в письме.
- capital_run_length_total (1 столбец) – общая сумма всех последовательностей непрерывающихся заглавных букв в письме (общее количество заглавных букв в письме).
- spam (1 столбец) – определяет тип письма: спам (1) или не спам (0).

Необходимо включить следующие модули:

```
library(caret) # машинное обучение
library(doParallel) # многопоточность
```

Для того, чтобы загрузить данные из csv файлов используется функция read.csv:

```
dataset <- read.csv("data.csv",
                    header=FALSE,sep=";")
names <- read.csv("names.csv",
                    header=FALSE,sep=";")
```

Переменная dataset содержит массив данных, а переменная names – названия столбцов. Полученный массив данных по умолчанию имеет не несущие информации названия столбцов, поэтому рекомендуется преобразовать названия столбцов переменной dataset в значения, которые содержатся в первой строке csv файла:

```
names <- sapply((1:nrow(names)),
                function(i)
                    toString(names[i,1]))
names(dataset) <- names
```

Закономерности или тренды в данных часто выявляются при графическом представлении информации. Так, из следующего графика видно, что в спам-письмах (y=1) значительно чаще встречаются заглавные буквы:

```
qplot(y, capital_run_length_average,
        data=dataset, colour= y)
+ xlab("Spam|Ham")
+ ylab("Average Length of CAPITAL letters")
```

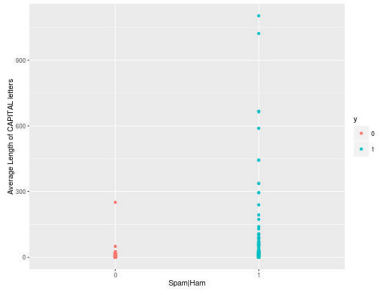

Рис. 1 – Количество заглавных букв в письмах

Получить более подробную информацию о распределении количества заглавных букв в спаме и обычных письмах можно с помощью функции summary:

```
summary(dataset[dataset$y == 1,
        'capital_run_length_average'])
summary(dataset[dataset$y == 0,
        'capital_run_length_average'])
```

Так, в данном случае среднее количество заглавных букв в спам-письмах (9.519) примерно в 4 раза превышает количество заглавных букв в обычных письмах (2.277).

II. Обучение модели методом Random forest

В рамках примера расмотрена модель, позволяющая определять спам-письма на основании ряда параметров. Для получения модели машинного обучения будем использовать алгоритм Random Forest. Перед началом обучения модели датасет необходимо разбить на тестовый (30% от всех писем) и обучающий (70%). Для обучения модели может быть использована функция train из пакета caret, которая может принимать множество различных параметров для получения более точной модели, например, параметры кросс-валидации.

```
inTrain <- createDataPartition(
                y=dataset$y,
                p=0.7,
                list=FALSE
            )
training <- dataset[inTrain,]
testing <- dataset[-inTrain,]
modFit <- train(y~., data=training,
                method="rf",
                trControl=trainControl(
                                method="cv",
                                number=5)
                            )
```

Для ускорения процедуры построения модели необходимо использовать использовать возможности многопоточности:

```
cl <- makeCluster(detectCores())
registerDoParallel(cl)
# вызов функции train
stopCluster(cl)
```

Кроме того, функция train() имеет атрибут allowParallel=TRUE, который позволяет сделать тоже самое.

III. Оценка качества модели

Получить прогнозные значения для тестового датасета (testing) можно с помощью функции predict:

```
pred <- predict(modFit, testing)
```

Для наглядности результаты прогнозирования можно преобразовать в таблицу (Таблица 1):

```
testing$predRight <- pred == testing$y
table(pred, testing$y)
```

В таблице 1 столбцы представляют собой точные значения (0 - обычное письмо, 1 - спам-письмо), в то время как строки - прогнозные значения. Правильные прогнозы находятся на главной диагонали таблицы. Из данных таблицы можно сделать вывод, что в случае ошибочного прогноза полученная модель чаще принимает обычные письма за спам (в 79 случаях), что можно считать ложным срабатыванием спам-фильтра, чем пропускает настоящие спам-письма (в 21 случае).

Таблица 1 – Результаты прогнозирования

Спам-письмо	0	1
0	757	21
1	79	522

Чтобы получить точность предсказывания, достаточно разделить количество правильных прогнозов на общее количество объектов в тестовом датасете. Полученная модель имеет точность 92,75%. Стоит отметить, что при более точной настройке параметров модели и изменении количества учитываемых переменных можно получить значительно лучшие результаты.

IV. Заключение

В данной работе была рассмотрена лишь малая часть всех возможностей языка R, который является мощным универсальным средством, подходящим как для разведочного анализа данных, так и для программирования сложных систем анализа больших объемов данных. Он позволяет в полуавтоматическом режиме генерировать отчеты с добавлением результатов анализа, графиков и диаграмм. Простота и удобство использования, большое количество пакетов (package) для всевозможных целей и поддержка ведущих университетов мира способствуют быстрому развитию и увеличению популярности данного языка как среди ученых различных областей (биология, медицина, экономика), так и среди разработчиков программного обеспечения.

1. Spambase Data Set [Electronic resource] / M. Hopkins, E. Reeber. – Hewlett-Packard Labs. – Mode of access: https://archive.ics.uci.edu/ml/datasets/Spambase. – Date of access: 29.09.2016.
2. The caret Package [Electronic resource]. – Mode of access: http://topepo.github.io/caret/index.html. – Date of access: 28.09.2016.
3. Package doParallel [Electronic resource]. – Mode of access: https://cran.r-project.org/web/packages/doParallel/doParallel.pdf. – Date of access: 28.09.2016.