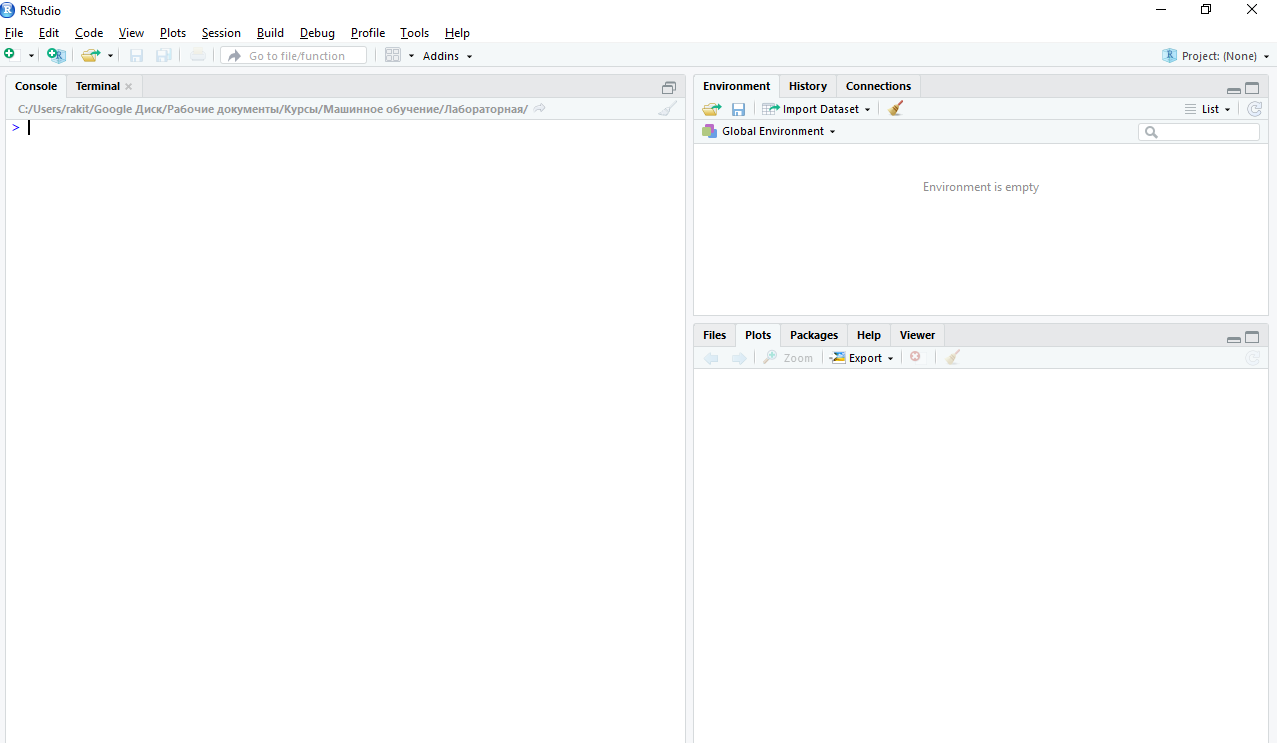
**Лабораторная работа: Основы работы с пакетом R**

R – это язык программирования для статистической обработки данных и работы с графикой, а также это свободная программная среда с открытым исходным кодом, развиваемая в рамках проекта GNU. Средства разработки на этом языке являются бесплатными и свободно распространяемыми, а кроме того представляют собой невероятно удобный способ обработки статистических данных.

В рамках лабораторной работы предлагается изучить основы языка R и базовые подходы, такие как ввод исследуемых данных, их преобразование и анализ при помощи линейной регрессии, а также графическое представление анализируемых данных.

По ссылке <https://www.r-project.org/> можно скачать базовые средства языка программирования, необходимые для работы, а также найти всю необходимую документацию. Кроме того, нам потребуется удобная визуальная среда разработки, которых существует несколько, однако в рамках лабораторной предлагается ознакомиться с одной из самых удобных (по мнению автора курса) RStudio. Необходимые для установки дистрибутивы, как и подробное описание находятся по ссылке <https://www.rstudio.com/>

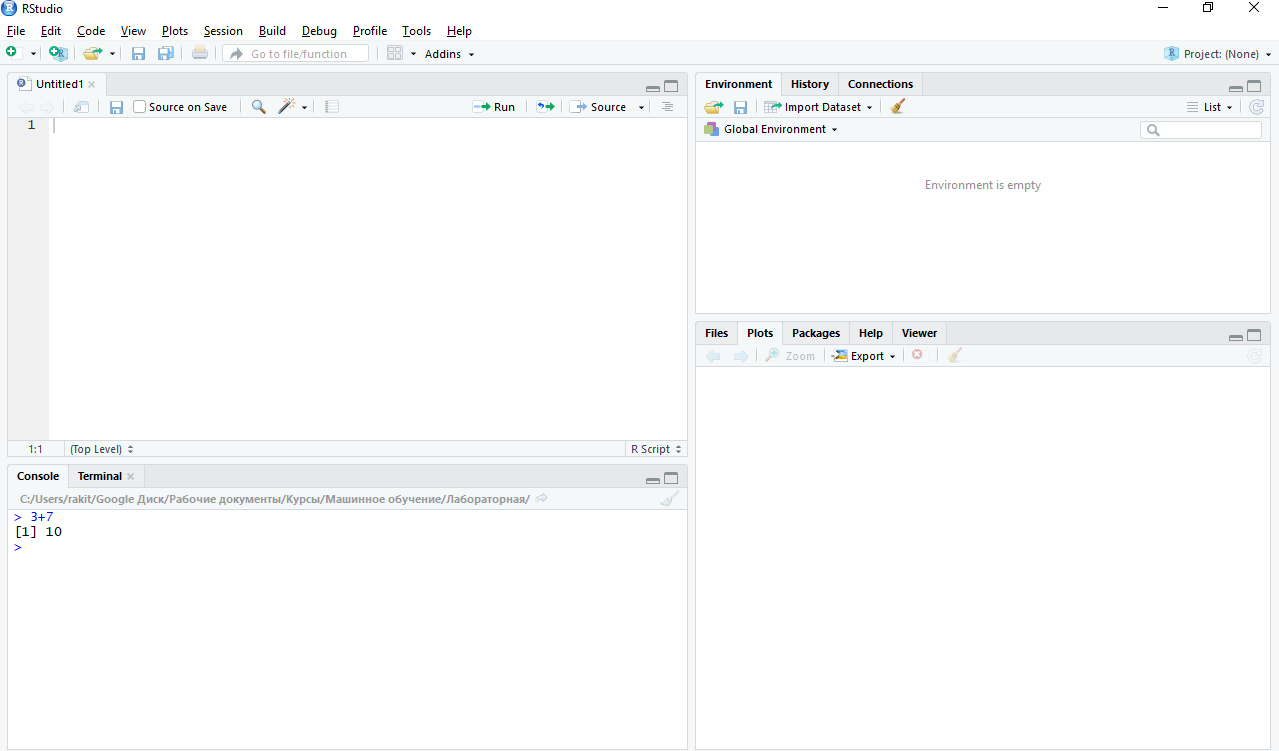
Скачав и установив, сначала базис языка, а потом и RStudio, можно приступить к изучению. Для начала запустим RStudio. Откроется следующее окно:



Так как язык является интерпретируемым, то мы можем начать набирать команды языка прямо в консоли, например, используя его как калькулятор. Если ввести в консоли 3+7 и нажать Enter, то на экран выведется ответ на это выражение:

>3+7  
>10

Тем не менее, нас интересует возможность создания скриптов, то есть более сложных программ, которые можно будет после ввода запустить на выполнение, не вводя команды по шагам. Для этого во вкладке File нужно выбрать New File -> R Script. Появится следующее окно:



Скрипт необходимо сохранить, введя его имя. Во избежание конфликтов в системе рекомендуется имена файлов задавать без пробелов и латинскими буквами (хотя это и не обязательно).

После этого можно приступить к работе в системе.

**Базовые конструкции языка R**

Подробное описание всех типов данных, управляющих конструкций и базовых функций с примерами можно найти по ссылке <https://ru.wikibooks.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F_R/%D0%92%D0%B2%D0%B5%D0%B4%D0%B5%D0%BD%D0%B8%D0%B5>

В первую очередь необходимо скачать и подключить ряд дополнительных пакетов, что в RStudio делается довольно просто. Необходимо нажать на вкладку Tools и выбрать опцию Install packages. Сначала будет предложено выбрать зеркало для скачивания (в принципе, можно выбрать любое, однако рекомендуется выбирать наиболее близкое территориально), после чего откроется окно с полным списком доступных пакетов. Для выполнения данной лабораторной работы нам понадобятся пакеты dplyr (необходим для эффективной и интуитивно-понятной работы с данными), ggplot2 (содержит функции для отрисовки графиков), GGally (дополнение к предыдущему пакету для более удобного вызова функций отрисовки) и foreign (считывание данных из файлов). При необходимости можно аналогично устанавливать и любые другие пакеты, которые могут потребоваться. В частности, если в связке с вышеуказанными не был установлен пакет RCpp, то необходимо установить и его.

Для подключения пакетов к скрипту используется функция library(“название\_пакета”).

Таким образом, наш скрипт будет начинаться со следующих строк:

library(foreign)  
library("dplyr")  
library("ggplot2")  
library("GGally")

**Работа с файлами**

Возможности языка R по вводу и редактированию данных умышленно ограничены создателями, которые предполагают, что для этого будут использованы более подходящие существующие программы (такие, как MS Excel, например, или любые другие подобные приложения). Поэтому, как правило, данные, подлежащие анализу, подготавливаются в сторонних программах и только потом загружаются в среду R. Предпочтительнее всего при этом использовать текстовые файлы, но при помощи библиотеки foreign возможно считывать данные, представленные и во многих других форматах (excel, spss, sas,stata,access, matlab, sql, oracle и т.д., подробнее можно прочитать в <http://cran.r-project.org/doc/manuals/R-data.pdf> ).

Отметим ряд правил, которым должны соответствовать все таблицы в загружаемых файлах:

* В таблице не должно быть пустых ячеек. Если в ячейке отсутствует значение, необходимо записать туда NA.
* Рекомендуется любую импортируемую таблицу преобразовать в текстовый файл, в котором в качестве разделителей будут использоваться символы табуляции. Как правило, чтобы это сделать, необходимо в приложении выбрать пункт Save as (сохранить как) и там выбрать нужный формат.
* Файл должен лежать в рабочей папке среды R. Определить рабочую папку можно при помощи getwd(). При этом можно пойти и другим путём, задать в качестве рабочей папки путь к той, где лежит вводимый файл, сделать это можно при помощи команды setwd(“путь”), где в качестве пути необходимо указать полный путь до папки. Не стоит при этом забывать, что в строковых последовательностях символ \ является служебным, поэтому для корректной обработки пути необходимо использовать \\ (например, “C:\\temp\\my\_prog”).
* В качестве первой строки файла по умолчанию располагается строка с заголовками столбцов, если заголовки отсутствуют, то это необходимо указать в соответствующем параметре функции read.table().

Основной функцией для импортирования данных в рабочую среду R является **read.table()**. Эта функция позволяет достаточно тонко настроить процесс загрузки внешних файлов, в связи с чем она имеет большое количество управляющих аргументов. Наиболее важные из этих аргументов перечислены ниже в таблице (подробнее см. файл помощи, доступный по команде ?read.table).

|  |  |
| --- | --- |
| **Аргумент** | **Назначение** |
| file | Служит для указания пути к импортируемому файлу. Путь приводят либо в абсолютном виде (например, file = "C:/Temp/MyData.dat"), либо указывают только имя импортируемого файла (например, file = "MyData.txt"), но при условии, что последний хранится в рабочей папке программы (см. выше). В качестве имени можно также указывать полную URL-ссылку на файл, который предполагается загрузить из Internet (например: file = "http://somesite.net/YourData.csv"). |
| header | Служит для сообщения программе о наличии в загружаемом файле строки с заголовками столбцов. По умолчанию принимает значение FALSE. Если строка с заголовками столбцов имеется, этому аргументу следует присвоить значение TRUE. |
| row.names | Служит для указания номера столбца, в котором содержатся имена строк (например, в рассмотренном выше примере это был первый столбец, поэтомуrow.names = 1). Важно помнить, что все имена строк должны быть уникальными, т.е. одинаковые имена для двух или более строк не допускаются. Если у строк нет названий, то этот параметр не указывается. |
| sep | Служит для указания используемого в файле разделителя значений переменных (*separator* – разделитель). По умолчанию предполагается, что значения переменных разделены "пустым пространством", например, в виде пробела или знака табуляции (sep = ""). В файлах формата csv значения переменных разделены запятыми, и поэтому для них sep = ",". |
| dec | Служит для указания знака, используемого в файле для отделения целой части числа от дроби. По умолчанию sep = ".". Однако во многих странах в качестве десятичного знака применяют запятую, о чем важно вспомнить перед загрузкой файла и, при необходимости, использовать dec = ",". |
| nrows | Выражается целым числом, указывающим количество строк, которое должно быть считано из загружаемой таблицы. Отрицательные и иные значения игнорируются. Пример: nrows = 100. Если нужно считать весь файл до конца, то этот параметр указывать не нужно |
| skip | Выражается целым числом, указывающим количество строк в файле, которое должно быть пропущено перед началом импортирования. Пример: skip = 5 |

Пример:

d <- read.table(“stud.txt”, header=FALSE)

**Создание данных при помощи средств языка R**

Рассмотрим простейший вариант, допустим, требуется проанализировать некоторую функцию, значения которой нужно задать при помощи формулы. Например, . При этом X меняется в диапазоне от 0 до 10 с шагом 0.5. Тогда напишем следующий код:

X <- (0:20)/2  
Y <- sin(X)+sqrt(X)

Выполнив этот код, получим два вектора, которые можно, например, объединить в некоторый фрейм.

F <- data.frame(Y,X,X^2,X^3,X^4)

Обратим внимание, что здесь были добавлены столбцы со значениями . Эти столбцы понадобятся, чтобы построить регрессию для функции на основе полинома 3й степени. Такой полином даст более качественное приближение к искомой функции.

Теперь у нас создан фрейм с тремя столбцами, однако у них нет имён, чтобы обращаться по ним в программе. Исправить это можно при помощи функции colnames():

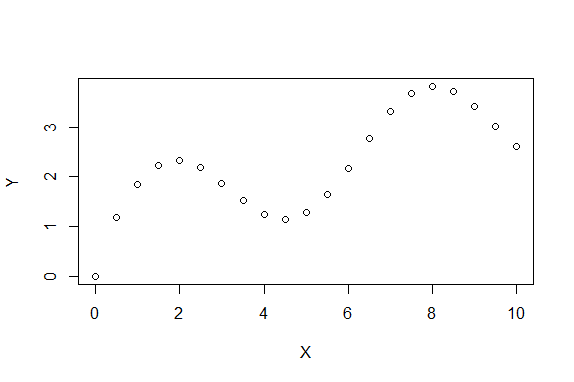
colnames(F) <- c(‘Y’,’X’,’X2’,’X3’,’X4’)

Так как имена столбцов – это вектор строк, то мы и присваиваем ему вектор, образованный из строк с названиями столбцов, в том порядке, в котором они были добавлены во фрейм.

**Линейная регрессия при помощи МНК и построение прогнозного значения**

Допустим мы уже сформировали тем или иным образом некоторый фрейм с данными, в нём есть зависимая переменная и независимая (или независимые, если их несколько) переменная. В качестве основного будем рассматривать пример из предыдущего пункта. То есть, имеется некоторые пары (x,y), где х – независимая переменная, а y – значение некоторой функции от этой переменной. Требуется построить модель, которая позволит получить значение y для произвольного x, не зная самой исходной функции. Кроме того, необходимо минимизировать среднеквадратичное отклонение полученного значения от исходного (задача аппроксимации).

Будем считать, что наша модель описывается некоторым полиномом . Прийти к подобному выводу можно, если посмотреть на графическое отображение исходных точек, мысленно соединив их линиями. Для построения графика используем функцию plot(X,Y)



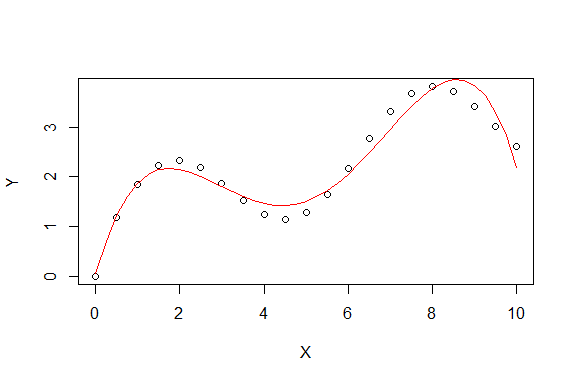
Для построения такой модели воспользуемся функцией построения линейной регрессии при помощи метода наименьших квадратов – lm(формула, данные):

model <- lm(Y~X+X2+X3+X4,F)  
В данном случае + не служит для арифметического сложения, а позволяет перечислить все независимые переменные, от которых зависит Y.

Если набрать в консоли model, то будет выведено значение коэффициентов A, соответствующих построенной модели. Таким образом мы теперь можем сделать предсказание. Создадим сначала массив аргументов , создадим на его основе фрейм с теми же столбцами и названиями, как были в F и посчитаем для них прогнозное значение .

X0 <- (0:40)/4  
Xpred <- data.frame(X0,X0^2,X0^3,X0^4)  
colnames(Xpred)=c(‘X’,’X2’,’X3’,’X4’)  
Y0 <- predict(model,Xpred)

Для отображения полученного прогноза и сопоставления его с исходными данными на графике напишем функцию lines(X0,Y0,col=’red’)



На графике можно увидеть, что полученное прогнозное значение оказалось довольно близко к искомому. Дополнительную информацию по всем описанным выше функциям и способам их применения можно найти по ссылкам, указанным в конце данного документа. Примеры анализа данных при помощи линейной регрессии с вариантами ввода с клавиатуры, и заполнения функцией находятся в папке с заданием к лабораторной работе по ссылке:

<https://drive.google.com/open?id=1IsBizt6hqQiw1FP-4VWvA1HSUPAaTXRG>

Для того, чтобы выполнить скрипт построчно, либо выполнить одну конкретную строку из него – необходимо установить курсор на эту строку и нажать ctrl+enter. Для выполнения всего скрипта целиком, необходимо выделить весь его код нажать ctrl+enter.

**Задание на лабораторную работу**

Общее задание является единым для всех вариантов, вариант влияет только на тип входных данных и способ ввода. Способов ввода будет 2:

1. Создать таблицу с данными и сохранить её в текстовом файле, данные считывать из этого файла
2. Данные задаются при помощи функции

При формировании данных необходимо создать не менее 30 пар значений. Необходимо написать скрипт, который будет считывать/формировать данные, по этим данным стоится наиболее подходящая модель линейной регрессии. При помощи полученной модели стоится прогнозное значение для заданного значения независимой переменной. Исходные данные и прогнозное значение отображаются на графике, который необходимо приложить к отчёту.

Отчёт по лабораторной работе включает в себя текст задания в соответствии с вариантом, исходный код полученного скрипта с поясняющими комментариями, вывод исходных данных в табличном виде, полученное значение прогноза для заданного в варианте значения независимой переменной, полный вывод работы скрипта и график. Степень полинома для построения модели студенту необходимо определить самостоятельно.

Для защиты лабораторной работы будут предложены контрольные вопросы, соответствующие теме и дополнительное контрольное значение, для которого нужно будет вычислить прогноз при помощи полученного скрипта.

Варианты задания:

Способ ввода данных 1):

1. Данные о машинах. Таблица состоит из 4х столбцов: год выпуска машины (диапазон 1930 – 2010), максимальная скорость машины (диапазон 60-200 км/ч), скорость разгона (за какое время скорость машины увеличивается на 1 км/ч, диапазон 0.01-1 с), мощность двигателя в лошадиных силах (диапазон 25-300). Генерировать данные необходимо приближенными к реальности. Зависимая переменная – время разгона машины до максимальной, независимая переменная – мощность двигателя. Контрольное значение для прогноза 145 л.с..

2. Данные об абитуриентах. Будем считать, что все абитуриенты уже много лет сдают 3 выпускных экзамена (математика, русский, физика) по 100 балльной системе. 4 столбца – год поступления (диапазон 1970-2010), средний балл по области по математике (диапазон 30-100), по русскому, по физике. Зависимая переменная – средняя сумма баллов, независимая переменная – средний балл по математике. Контрольное значение для прогноза 72.25.

3. Курс рубля. 3 столбца – день наблюдения (диапазон 1-1000), курс рубля к доллару (диапазон 1-100, числа дробные), курс рубля к евро (диапазон 1-100, числа дробные), дни наблюдения должны быть упорядочены. Зависимая переменная – стоимость бивалютной корзины, независимая переменная – день наблюдения. Контрольное значение для прогноза 1001.

4. Цены акций нефтяных компаний. 3 столбца – день наблюдения (диапазон 1-1000), цена за баррель нефти (1-200$, число дробное), стоимость акции (0.1-1000$). Зависимой переменной является стоимость акции, независимой – цена за баррель. Контрольное значение для прогноза 100.

5. Зависимость роста детей от роста родителей. 3 столбца – рост отца (150-220 см), рост матери (150-220 см), средний рост всех взрослых детей пары (150-220 см), все числа дробные до 2х знаков после запятой. Зависимая переменная – средний рост детей, независимая переменная – средний рост родителей (рост отца+рост матери / 2). Контрольное значение для прогноза 185 см.

Способ ввода данных 2):

6. . Диапазон X: 0.1-10. Контрольное значение для прогноза X=9.35.

7. . Диапазон Х: 0.1-10. Контрольное значение для прогноза X=6.75.

8. . Диапазон Х: 2-10. Контрольное значение для прогноза Х=7.5.

9. . Диапазон Х: 1-10. Контрольное значение для прогноза Х=11.

10. . Диапазон X: 1-10. Контрольное значение для прогноза Х=5.5.

Ссылки на дополнительные материалы:

<http://www.inp.nsk.su/~baldin/DataAnalysis/R/R-01-intro.pdf> - Подробная инструкция по установке и настройке языка R, а также основы разработки на нём

<http://r-analytics.blogspot.ru/2011/07/r_24.html#.WoH9PlRuZdh> – Инструкция по работе с файлами.

<http://kpfu.ru/docs/F407025247/metodichka_R_2.pdf> - Использование языка R для статистической обработки данных

<http://www.algorithmist.ru/2011/11/regularization-with-examples-in-r.html> - Линейная регрессия с примерами на R

<http://finzi.psych.upenn.edu/nmz.html> — поиск в материалах по R

<http://www.statmethods.net/index.html> — хороший справочный ресурс

<http://zoonek2.free.fr/UNIX/48_R/all.html> — ещё один справочный ресурс

<http://pj.freefaculty.org/R/Rtips.html> — советы по использованию R