

Министерство науки и высшего образования Российской Федерации
Севастопольский государственный университет
Кафедра ИС

Отчет
по лабораторной работе №2
«Корреляционный и регрессионный анализ данных»
по дисциплине
«ИНТЕЛЛЕКТУАЛЬНЫЙ АНАЛИЗ ДАННЫХ»

Выполнил студент группы ИС/б-17-2-о
Горбенко К. Н.
Проверил
Сырых О.А.

Севастополь
2020

1 ЦЕЛЬ РАБОТЫ

- исследовать возможности языка R для проведения корреляционного и регрессионного анализа данных;
- создание набора данных для проведения корреляционного и регрессионного анализа данных.

2 ЗАДАНИЕ НА РАБОТУ

1. Исследовать основные функции и команды языка R, представленные в данной лабораторной работе;
2. выполнить все примеры;
3. подобрать экспериментальные данные для анализа;
4. выполнить ввод данных с клавиатуры;
5. провести экспорт данных из текстового файла с разделителями;
6. выполнить экспорт данных из Excel.

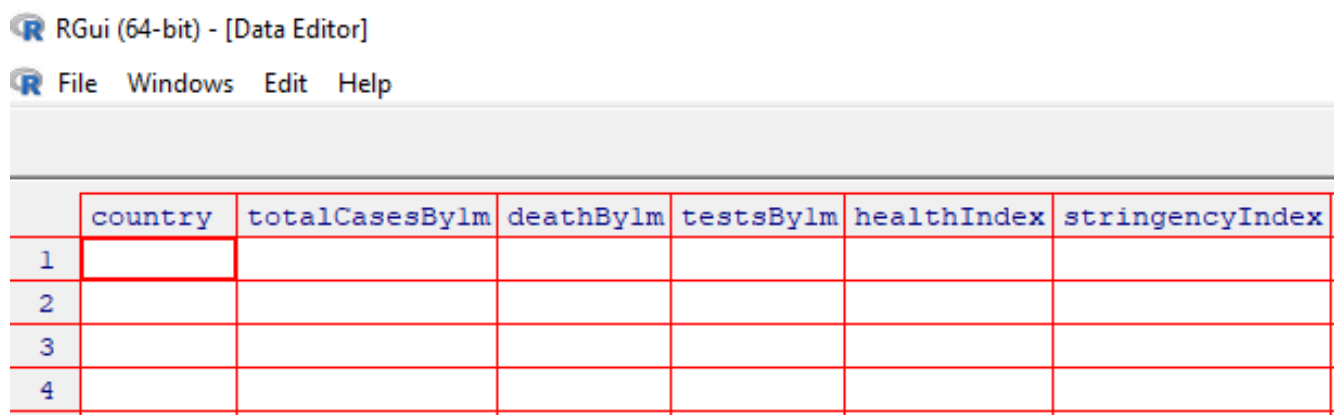
3 ХОД РАБОТЫ

3.1 Ввод данных

Выполним ввод данных с клавиатуры:

```
1 > data <- data.frame(country=character(0), totalCasesBy1m=numeric(0), deathBy1m=
  =numeric(0), testsBy1m=numeric(0), healthIndex=numeric(0), stringencyIndex=
  numeric(0))
2 > data <- edit(data)
```

Результат выполнения программы приведен на рисунке 1.



	country	totalCasesBy1m	deathBy1m	testsBy1m	healthIndex	stringencyIndex
1						
2						
3						
4						

Рисунок 1 – Ввод данных с клавиатуры

Выполним ввод данных из текстового файла:

```
1 > data <- read.table("D:\\Repositories\\Learning\\ИАД\\data.csv", header=TRUE,
  sep=",")
2 > data
```

Результат выполнения программы приведен на рисунке 2.

```
> data <- read.table("D:\\Repositories\\Learning\\ИАД\\data.csv", header=TRUE, sep=",")
> data
```

	country	total_cases_by_lm	deaths_by_lm	tests_by_lm	health_index	stringency_index
1	Spain	16003	672	272117	82	85.19
2	Luxembourg	13328	197	1302265	81	79.63
3	Moldova	12698	323	68645	69	87.04
4	Belgium	9942	861	272178	80	81.48
5	Sweden	8989	581	151525	81	46.30
6	France	8309	487	164025	81	87.96
7	Belarus	8250	87	194234	72	19.44
8	BosniaandHerzegovina	8243	253	71121	70	92.59
9	Russia	7945	140	311066	68	85.19
10	Iceland	7791	29	804135	82	53.70
11	Portugal	7265	192	250565	79	87.96
12	Ireland	7145	364	233290	80	90.74
13	Netherlands	6681	372	130687	82	79.63
14	UK	6459	618	353324	80	79.63
15	Romania	6454	247	123524	73	87.04
16	Czechia	6149	58	124686	79	74.07
17	Switzerland	6072	238	155610	84	89.81
18	Italy	5152	593	184286	81	87.96
19	Austria	4816	88	174076	82	85.19
20	Ukraine	4693	93	50987	64	88.89
21	Denmark	4670	112	648565	82	72.22
22	Albania	4654	132	28494	73	84.26
23	Croatia	3963	66	72213	75	96.30
24	Serbia	3828	86	128356	72	100.00
25	Germany	3442	114	186555	82	76.85
26	Bulgaria	2923	116	74923	74	71.30
27	Hungary	2648	78	72974	75	76.85
28	Slovenia	2592	72	105687	79	89.81
29	Norway	2538	50	190491	83	79.63
30	Estonia	2462	48	156598	75	77.78
31	Poland	2343	65	86368	77	83.33
32	Finland	1758	62	176063	79	60.19
33	Slovakia	1711	8	82036	77	87.04
34	Greece	1701	37	123406	79	84.26
35	Lithuania	1655	34	280763	70	87.04
36	Latvia	902	19	165997	71	65.74

Рисунок 2 – Ввод данных из .csv файла

Выполним ввод из xlsx файла:

```
1 library("xlsx")
2 > data <- read.xlsx("D:\\Repositories\\Learning\\ИАД\\data.xlsx", 1)
3 > data
```

Результат выполнения программы приведен на рисунке 3.

```

> data <- read.xlsx("D:\\Repositories\\Learning\\ИАД\\data.xlsx", 1)
> data

```

	country	total_cases_by_lm	deaths_by_lm	tests_by_lm	health_index	stringency_index
1	Spain	16003	672	272117	82	85.19
2	Luxembourg	13328	197	1302265	81	79.63
3	Moldova	12698	323	68645	69	87.04
4	Belgium	9942	861	272178	80	81.48
5	Sweden	8989	581	151525	81	46.30
6	France	8309	487	164025	81	87.96
7	Belarus	8250	87	194234	72	19.44
8	BosniaandHerzegovina	8243	253	71121	70	92.59
9	Russia	7945	140	311066	68	85.19
10	Iceland	7791	29	804135	82	53.70
11	Portugal	7265	192	250565	79	87.96
12	Ireland	7145	364	233290	80	90.74
13	Netherlands	6681	372	130687	82	79.63
14	UK	6459	618	353324	80	79.63
15	Romania	6454	247	123524	73	87.04
16	Czechia	6149	58	124686	79	74.07
17	Switzerland	6072	238	155610	84	89.81
18	Italy	5152	593	184286	81	87.96
19	Austria	4816	88	174076	82	85.19
20	Ukraine	4693	93	50987	64	88.89
21	Denmark	4670	112	648565	82	72.22
22	Albania	4654	132	28494	73	84.26
23	Croatia	3963	66	72213	75	96.30
24	Serbia	3828	86	128356	72	100.00
25	Germany	3442	114	186555	82	76.85
26	Bulgaria	2923	116	74923	74	71.30
27	Hungary	2648	78	72974	75	76.85
28	Slovenia	2592	72	105687	79	89.81
29	Norway	2538	50	190491	83	79.63
30	Estonia	2462	48	156598	75	77.78
31	Poland	2343	65	86368	77	83.33
32	Finland	1758	62	176063	79	60.19
33	Slovakia	1711	8	82036	77	87.04
34	Greece	1701	37	123406	79	84.26
35	Lithuania	1655	34	280763	70	87.04
36	Latvia	902	19	165997	71	65.74

```

> |

```

Рисунок 3 – Ввод данных из .xlsx файла

3.2 Построение графиков

Выведем индекс строгости для нескольких стран:

```

1 > countries <- data[[1]]
2 > indexes <- data[[6]]
3 > plot(factor(countries[1:5]), indexes[1:5], col="red", xlab="Country", ylab="
    Stringency Index", ylim=c(0, 100))

```

Результат выполнения программы приведен на рисунке 4.

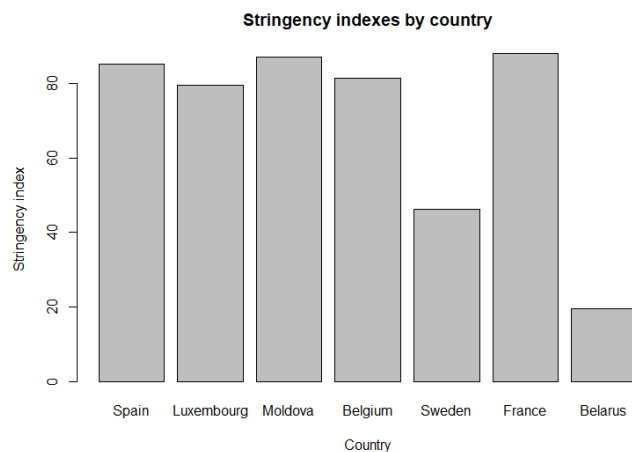


Рисунок 4 – Индексы строгости стран

Выведем коэффициенты здравоохранения для всех стран:

```

1 countries <- data[[1]]
2 healthIndexes = data[[5]]
3 library(ggplot2)
4 qplot(factor(countries),
5       healthIndexes,
6       geom="boxplot",
7       xlab="Countries",
8       ylab="Health indexes") +
9   theme(axis.text.x=element_text(angle=90, vjust=0.5, hjust=1)) +
10  geom_bar(stat="identity", position="stack") +
11  labs(x=NULL, y="Health indexes")

```

Результат выполнения программы приведен на рисунке 4.

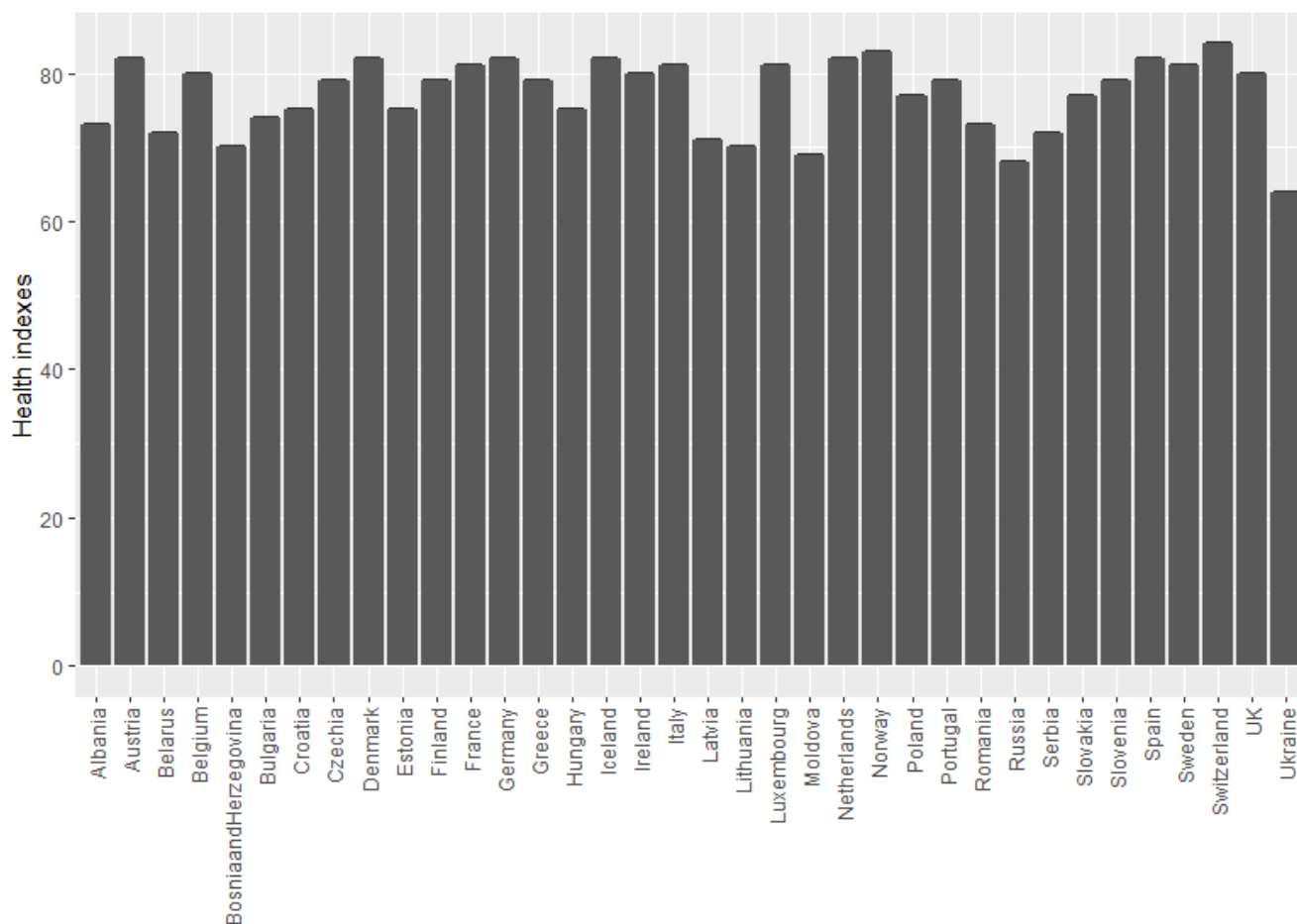


Рисунок 5 – Коэффициенты здравоохранения стран

3.3 Вычисление корреляции

Вычислим матрицу корреляции методами Пирсона и Спирмена:

```

1 cor(data[c("total_cases_by_1m", "deaths_by_1m", "tests_by_1m", "health_index",
2   "stringency_index")],
3   method="pearson",
4   use="complete")
5 cor(data[c("total_cases_by_1m", "deaths_by_1m", "tests_by_1m", "health_index",
6   "stringency_index")],
7   method="spearman",
8   use="complete")

```

Результат выполнения программы приведен на рисунке 6.

```
> cor(data[c("total_cases_by_1m", "deaths_by_1m", "tests_by_1m", "health_index", "stringency_index")],
+      method="pearson",
+      use="complete")
      total_cases_by_1m deaths_by_1m tests_by_1m health_index stringency_index
total_cases_by_1m      1.000000000  0.63626070  0.40448409  0.1292878  -0.06137103
deaths_by_1m           0.63626070  1.00000000  0.04095711  0.3147870  0.07214384
tests_by_1m            0.40448409  0.04095711  1.00000000  0.3262891  -0.18930692
health_index           0.12928779  0.31478698  0.32628909  1.0000000  -0.08291790
stringency_index       -0.06137103  0.07214384 -0.18930692 -0.0829179  1.00000000
> cor(data[c("total_cases_by_1m", "deaths_by_1m", "tests_by_1m", "health_index", "stringency_index")],
+      method="spearman",
+      use="complete")
      total_cases_by_1m deaths_by_1m tests_by_1m health_index stringency_index
total_cases_by_1m      1.000000000  0.7619048  0.2893179  0.1866414  0.06460778
deaths_by_1m           0.76190476  1.00000000  0.1480051  0.2370850  0.22090443
tests_by_1m            0.28931789  0.1480051  1.00000000  0.4639519  -0.29789214
health_index           0.18664139  0.2370850  0.4639519  1.0000000  -0.19550372
stringency_index       0.06460778  0.2209044 -0.2978921 -0.1955037  1.00000000
```

Рисунок 6 – Матрицы корреляции, составленные методами Пирсона и Спирмена

Судя по коэффициентам, сильная корреляция наблюдается только между количеством заражений и количеством смертей. Средняя - между количеством заражений и количеством тестов, между количеством тестов и смертей и коэффициентами здравоохранения. Во всех остальных случаях – корреляция слабая.

Выполним оценку уровня значимости коэффициента корреляции между коэффициентами здравоохранения и количеством смертей и между индексом строгости и количеством заражений.

```
1 with(data, cor.test(health_index, deaths_by_1m, method="pearson"))
2 with(data, cor.test(stringency_index, total_cases_by_1m, method="pearson"))
```

Результат выполнения программы приведен на рисунке 7.

```
> with(data, cor.test(health_index, deaths_by_1m, method="pearson"))

Pearson's product-moment correlation

data: health_index and deaths_by_1m
t = 1.9338, df = 34, p-value = 0.0615
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.01533465  0.58302680
sample estimates:
cor
0.314787

> with(data, cor.test(stringency_index, total_cases_by_1m, method="pearson"))

Pearson's product-moment correlation

data: stringency_index and total_cases_by_1m
t = -0.35853, df = 34, p-value = 0.7222
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.3822006  0.2726623
sample estimates:
cor
-0.06137103
```

Рисунок 7 – Оценка уровня значимости коэффициента корреляции

Для обоих случаев связь между переменными не доказана, нулевая гипотеза

не отвергается.

Построим матрицу точечных графиков:

```
1 scatterplotMatrix(~deaths_by_1m+health_index+stringency_index+tests_by_1m+
  total_cases_by_1m,
2                      regLine=FALSE, smooth=FALSE, diagonal=list(method="density"),
                      data=Dataset)
```

Результат выполнения программы приведен на рисунке 8.

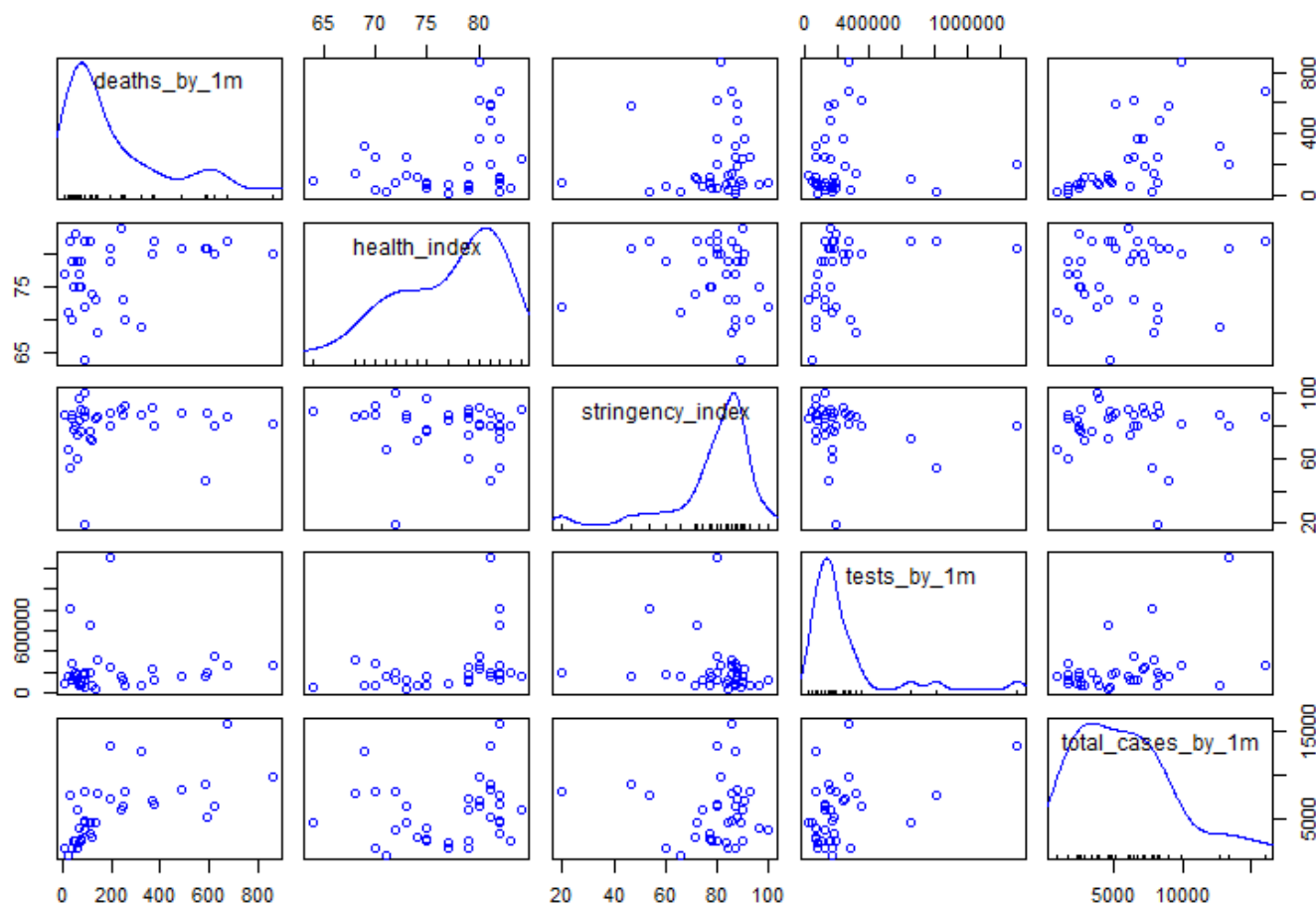


Рисунок 8 – Матрица точечных графиков

Постройте уравнение зависимости для параметров индекса здравоохранения и количества смертей. Получим функцию: $health_index = 75.492 + 1.136 * death_by_1m$. Построим график остатков.


```
Rcmdr> RegModel.1 <- lm(health_index~deaths_by_1m, data=Dataset)
Rcmdr> summary(RegModel.1)

Call:
lm(formula = health_index ~ deaths_by_1m, data = Dataset)

Residuals:
    Min       1Q   Median       3Q      Max
-12.169  -3.619   1.233   3.124   7.144

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  75.491888    1.136222   66.441  <2e-16 ***
deaths_by_1m   0.007283    0.003766    1.934   0.0615 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.875 on 34 degrees of freedom
Multiple R-squared:  0.09909, Adjusted R-squared:  0.07259
F-statistic:  3.74 on 1 and 34 DF, p-value: 0.0615
```

Рисунок 9 – Уравнение зависимости индекса здравоохранения от количества смертей

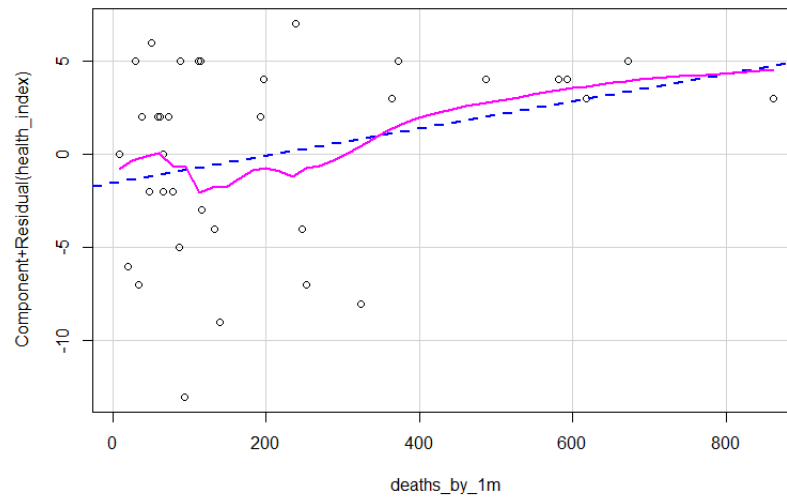


Рисунок 10 – График остатков для параметров индекс здравоохранения и количества смертей

Построим уравнение множественной регрессии:

```
Rcmdr> summary(RegModel.3)

Call:
lm(formula = health_index ~ deaths_by_1m + stringency_index +
    total_cases_by_1m, data = Dataset)

Residuals:
    Min       1Q   Median       3Q      Max
-11.727  -3.318   0.693   3.140   7.201

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  79.3962390   4.8261605   16.451  <2e-16 ***
deaths_by_1m  0.0095637   0.0050203    1.905  0.0658 .
stringency_index -0.0406307   0.0561313   -0.724  0.4744
total_cases_by_1m -0.0002017  0.0003098   -0.651  0.5197
```

Рисунок 11 – Уравнение множественной регрессии

Построим график компонента-остаток для модели множественной регрессии:

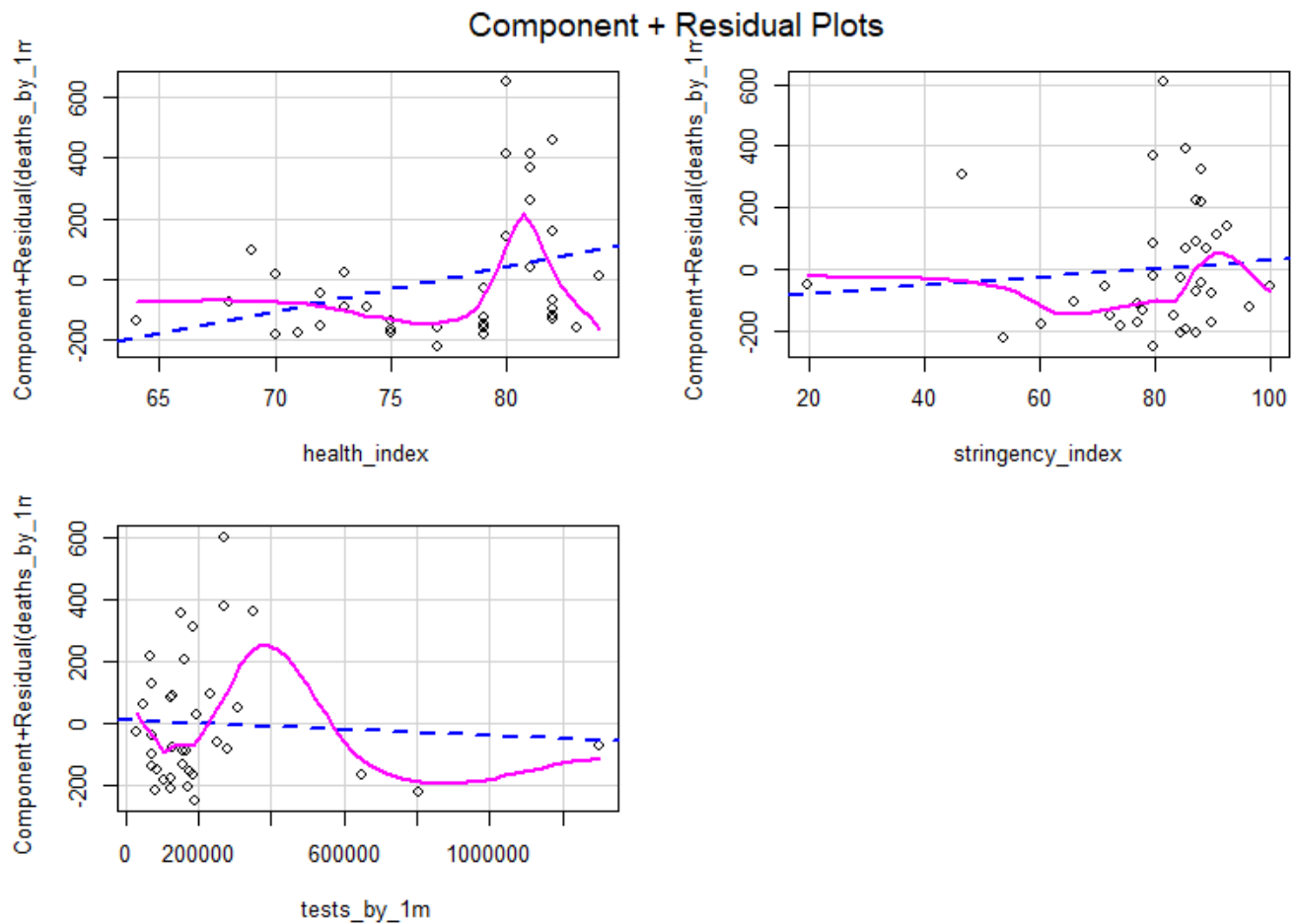


Рисунок 12 – Графики для модели множественной регрессии

Построение регрессии по направлению назад:

```
Rcmdr> stepwise(RegModel.4, direction='forward/backward', criterion='AIC')

Direction: forward/backward
Criterion: AIC

Start: AIC=388.94
deaths_by_1m ~ 1

      Df Sum of Sq    RSS   AIC
+ health_index      1  166048 1509666 387.18
<none>                        1675714 388.94
+ stringency_index  1     8722 1666992 390.75
+ tests_by_1m       1     2811 1672903 390.88

Step: AIC=387.18
deaths_by_1m ~ health_index

      Df Sum of Sq    RSS   AIC
<none>                        1509666 387.18
+ stringency_index  1     16286 1493379 388.79
- health_index      1    166048 1675714 388.94
+ tests_by_1m       1      7152 1502514 389.01

Call:
lm(formula = deaths_by_1m ~ health_index, data = Dataset)

Coefficients:
(Intercept)  health_index
   -837.14         13.61
```

Рисунок 13 – Пошаговое построение регрессии по направлению назад

Построение регрессии по направлению вперед:

```
Rcmdr> stepwise(RegModel.4, direction='backward/forward', criterion='AIC')

Direction: backward/forward
Criterion: AIC

Start: AIC=390.69
deaths_by_1m ~ health_index + stringency_index + tests_by_1m

      Df Sum of Sq    RSS    AIC
- tests_by_1m      1      4037 1493379 388.79
- stringency_index 1      13171 1502514 389.01
<none>                        1489342 390.69
- health_index      1      172466 1661808 392.64

Step: AIC=388.79
deaths_by_1m ~ health_index + stringency_index

      Df Sum of Sq    RSS    AIC
- stringency_index 1      16286 1509666 387.18
<none>                        1493379 388.79
+ tests_by_1m      1      4037 1489342 390.69
- health_index      1      173612 1666992 390.75

Step: AIC=387.18
deaths_by_1m ~ health_index

      Df Sum of Sq    RSS    AIC
<none>                        1509666 387.18
+ stringency_index 1      16286 1493379 388.79
- health_index      1      166048 1675714 388.94
+ tests_by_1m      1       7152 1502514 389.01

Call:
lm(formula = deaths_by_1m ~ health_index, data = Dataset)

Coefficients:
(Intercept) health_index
      -837.14        13.61
```

Рисунок 14 – Пошаговое построение регрессии по направлению вперед

ВЫВОДЫ

В ходе выполнения лабораторной работы были изучены такие типы данных в языке R, как список, таблица. Также были изучены методы импорта текстовых файлов с разделителями и Excel-файлов. Кроме того, были проанализированы зависимости между переменными методами корреляции и регрессии.