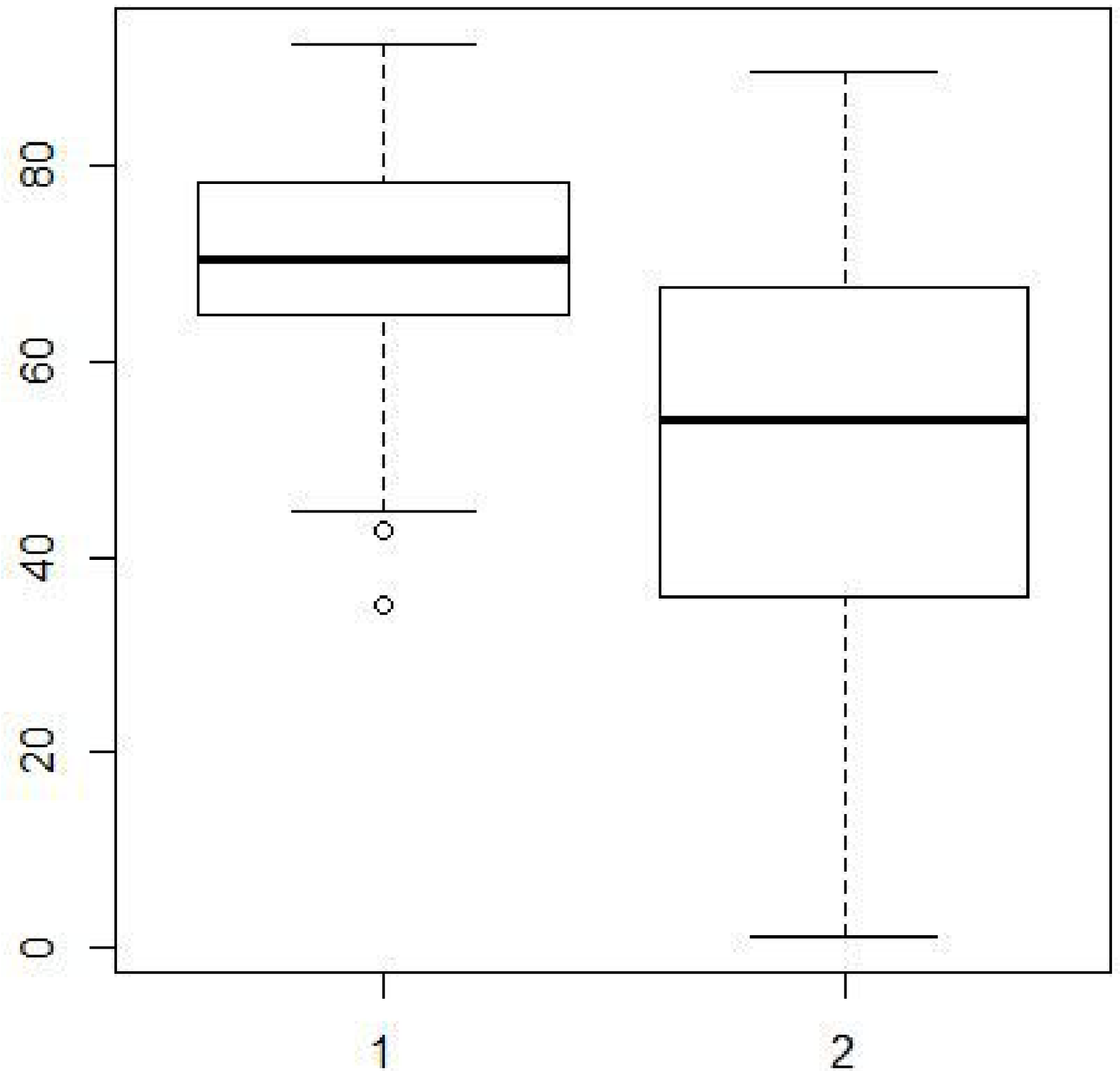
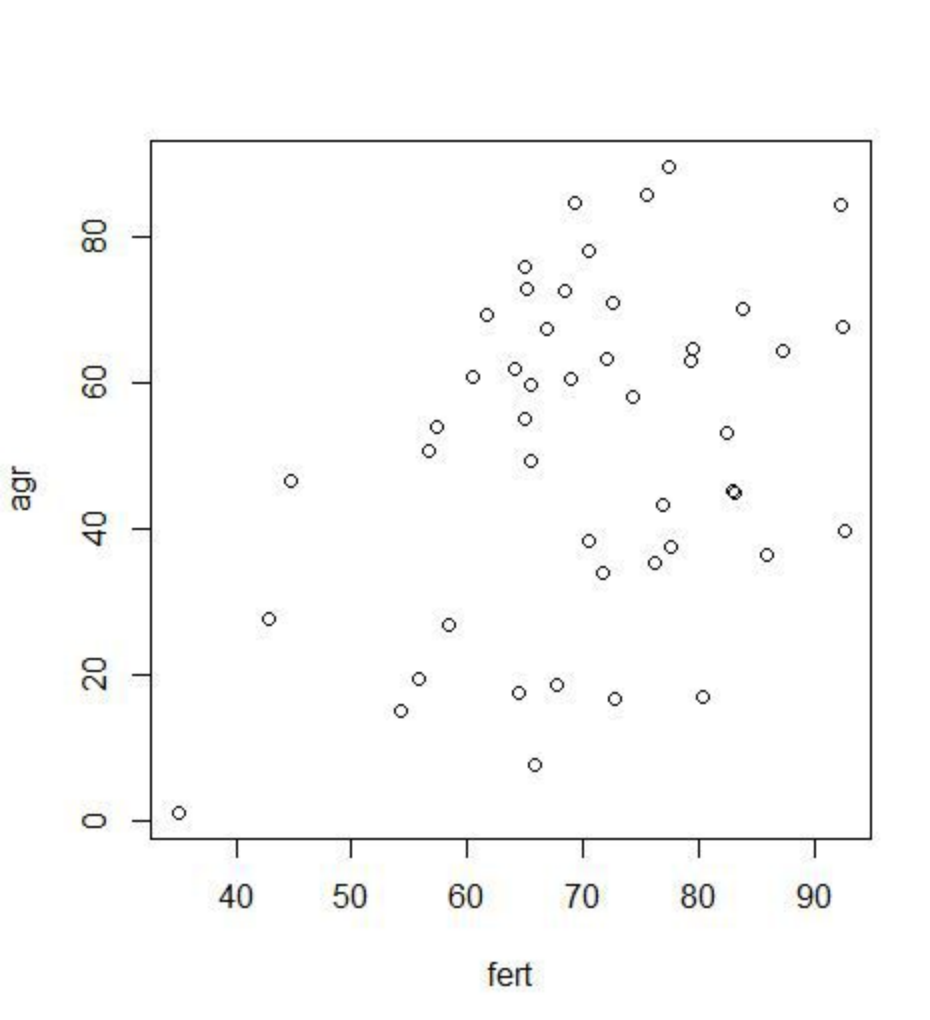
```r
fert <- swiss[,1]
agr <- swiss[,2]
exam <- swiss[,3]
educ <- swiss[,4]
relig <- swiss[,5]
mort <- swiss[,6]

features <- cbind(agr, exam, educ, relig, mort)
features_names <- c("Agriculure:      ", "Examination:     ", "Education:       ", "Religion:

#(i)

res <- rep(0,5)
for (i in (1:5)){
  spear = cor.test(features[,i], fert, method = "spearman")
  kend = cor.test(features[,i], fert, method = "kendall")
  res[i] = paste(features_names[i],
        "correlation (spearman)", toString(round(spear$estimate, 2)),
        " p-value:", toString(round(spear$p.value, 2)),
        "correlation (kendall)", toString(round(kend$estimate, 2)),
        "p-value:", toString(round(kend$p.value, 2)))
  }

res # Коэффициенты корреляции всех признаков с fertility
```

```
>
> res # Коэффициенты корреляции
[1] "Agriculure:       correlation (spearman) 0.24  p-value: 0.1 correlation (kendall) 0.18 p-value: 0.08"
[2] "Examination:      correlation (spearman) -0.66  p-value: 0 correlation (kendall) -0.48 p-value: 0"
[3] "Education:        correlation (spearman) -0.44  p-value: 0 correlation (kendall) -0.33 p-value: 0"
[4] "Religion:         correlation (spearman) 0.41  p-value: 0 correlation (kendall) 0.25 p-value: 0.02"
[5] "Infant mortality: correlation (spearman) 0.44  p-value: 0 correlation (kendall) 0.32 p-value: 0"
> W = 88, p-value = 0.1175
```

```r
#(ii)

boxplot(fert, agr) # диаграмма с "усами"
# Как можно видеть, есть несколько выбросов по fertility, по признаку agriculture выбросов нет.

M <- cbind(fert, features)

q <- boxplot.stats(fert)$out
M <- M[-which(fert %in% q),] # избавляемся от выбросов

spear = cor.test(M[,2], M[,1], method = "spearman")
kend = cor.test(M[,2], M[,1], method = "kendall")
paste("Spearman correlation:", toString(round(spear$estimate, 2)),
      " p-value:", toString(round(spear$p.value, 2)),
      "Kendall correlation:", toString(round(kend$estimate, 2)),
      "p-value:",toString(round(kend$p.value, 2)))

plot(fert, agr)
# Хоть корреляция между признаками и есть, она невелика, к тому же
# в обоих тестах p-value больше 0.1. Поэтому полученный результат незначим.

# Как мы видим, корреляция после удаления выбросов ощутимо уменьшилась.
# Коэффициент корреляции - мера линейной зависимости двух переменных, и мы строим прямую регрессии,
# которая использует сумму квадратов расстояний от наблюдаемых точек до этой прямой.
# Так как выбросы довольно далеки от средних значений, квадраты расстояний от них до прямой
# сильно влияют на результат, поэтому когда мы их убрали, корреляция пересчиталась. Могла и вырасти.
```


0
20
40
60
80
1
2


agr
0
20
40
60
80
40
50
60
70
80
90
fert

```
+       "p-value:",toString(round(kend$p.value, 2)))
[1] "Spearman correlation: 0.15  p-value: 0.32 Kendall correlation: 0.12 p-value: 0.24"
>
> plot(fert, agr)
> # Хоть корреляция между признаками и есть, она невелика, к тому же
> # в обоих тестах p-value больше 0.1. Поэтому полученный результат незначим.
>
> # Как мы видим, корреляция после удаления выбросов ощутимо уменьшилась.
> # Коэффициент корреляции - мера линейной зависимости двух переменных, и мы строим прямую регрессии,
> # которая использует сумму квадратов расстояний от наблюдаемых точек до этой прямой.
> # Так как выбросы довольно далеки от средних значений, квадраты расстояний от них до прямой
> # сильно влияют на результат, поэтому когда мы их убрали, корреляция пересчиталась. Могла и вырасти.
> W = 88, p-value = 0.1175
```

```r
#(iii)

scaled_feat <- scale(features)
dst <- rep(0, 37)
res <- rep(0,10)
for (i in 1:10){
  x = scaled_feat[i,]
  for(j in 1:37){
    dst[j] = dist(rbind(x, scaled_feat[j+10,]))
  }
  res[i] = fert[which.min(dst)]
}

plot(res, fert[1:10])
wilcox.test(res, fert[1:10], alternative = "two.sided", paired = TRUE)
# p-value больше чем 0.1, то есть даже на 10%-ном уровне значимости нулевая гипотеза о том,
# что значения fertility в 10 парах похожих провинций одинаковы, не отвергается.
```

```
> wilcox.test(res, fert[1:10], alternative = "two.sided", paired = TRUE)

        Wilcoxon signed rank test with continuity correction

data:  res and fert[1:10]
V = 11.5, p-value = 0.1139
alternative hypothesis: true location shift is not equal to 0

Warning message:
In wilcox.test.default(res, fert[1:10], alternative = "two.sided",  :
  не могу подсчитать точное p-значение при наличии повторяющихся наблюдений
> # p-value больше чем 0.1, то есть даже на 10%-ном уровне значимости нулевая гипотеза о том,
> # что значения fertility в 10 парах похожих провинций одинаковы, не отвергается.
> W = 88, p-value = 0.1175
```

fert[1:10]
90
85
80
74
76
78
80
82
84
res

```r
#(iv)

C <- swiss[which(relig > 80),]
P <- swiss[which(relig < 20),]
M <- swiss[-which(relig > 80),]
M <- M[-which(M[,5] < 20),]

kruskal.test(list(C[,1], P[,1], M[,1])) # для нескольких (>2) выборок

# Тесты Уилкоксона
wilcox.test(C[,1], P[,1]) # H0 отвергаем
wilcox.test(C[,1], M[,1]) # H0 отвергаем на уровне значимости 5%
wilcox.test(P[,1], M[,1]) # очень большое p-значение!!
# Посмотрим, можно ли проверить альтернативу попроще:
wilcox.test(P[,1], M[,1], alternative = 'greater') # p-value велико
wilcox.test(P[,1], M[,1], alternative = 'less') # p-value порядка 0.9
# Для этой пары нулевая гипотеза не отвергается даже для прочих альтернатив
```

```
> kruskal.test(list(C[,1], P[,1], M[,1])) # для нескольких (>2) выборок

        Kruskal-Wallis rank sum test

data:  list(C[, 1], P[, 1], M[, 1])
Kruskal-Wallis chi-squared = 18.948, df = 2, p-value = 7.681e-05

>
> # Тесты Уилкоксона
> wilcox.test(C[,1], P[,1]) # H0 отвергаем

        Wilcoxon rank sum test with continuity correction

data:  C[, 1] and P[, 1]
W = 372.5, p-value = 2.158e-05
alternative hypothesis: true location shift is not equal to 0

Warning message:
In wilcox.test.default(C[, 1], P[, 1]) :
  не могу подсчитать точное p-значение при наличии повторяющихся наблюдений
> wilcox.test(C[,1], M[,1]) # H0 отвергаем на уровне значимости 5%

        Wilcoxon rank sum test

data:  C[, 1] and M[, 1]
W = 67, p-value = 0.02496
alternative hypothesis: true location shift is not equal to 0
```

```
> wilcox.test(P[,1], M[,1]) # очень большое p-значение!!

        Wilcoxon rank sum test

data:  P[, 1] and M[, 1]
W = 88, p-value = 0.235
alternative hypothesis: true location shift is not equal to 0

> # Посмотрим, можно ли проверить альтернативу попроще:
> wilcox.test(P[,1], M[,1], alternative = 'greater') # p-value велико

        Wilcoxon rank sum test

data:  P[, 1] and M[, 1]
W = 88, p-value = 0.1175
alternative hypothesis: true location shift is greater than 0

> wilcox.test(P[,1], M[,1], alternative = 'less') # p-value порядка 0.9

        Wilcoxon rank sum test

data:  P[, 1] and M[, 1]
W = 88, p-value = 0.8929
alternative hypothesis: true location shift is less than 0

> # Для этой пары нулевая гипотеза не отвергается даже для прочих альтернатив
> W = 88, p-value = 0.1175
```

```r
#(v)

mort_1st_quart_C <- quantile(swiss[which(relig > 80), 6], 0.25) # смотрим на 0.25-квантиль
med_agr_C <- quantile(swiss[which(relig > 80), 2], 0.5)
mort_1st_quart_P <- quantile(swiss[which(relig < 20), 6], 0.25) # смотрим на 0.25-квантиль
med_agr_P <- quantile(swiss[which(relig < 20), 2], 0.5)
mort_1st_quart_M <- quantile(swiss[which((relig>=20)&(relig<=80)), 6], 0.25)
med_agr_M <- quantile(swiss[which((relig>=20)&(relig<=80)), 2], 0.5)

# Католики
C_1 <- swiss[which((relig > 80) & (agr > med_agr_C) & (mort < mort_1st_quart_C)), 1]
C_2 <- swiss[which((relig > 80) & (agr < med_agr_C) & (mort < mort_1st_quart_C)), 1]
C_3 <- swiss[which((relig > 80) & (agr > med_agr_C) & (mort > mort_1st_quart_C)), 1]
C_4 <- swiss[which((relig > 80) & (agr < med_agr_C) & (mort > mort_1st_quart_C)), 1]

# Протестанты
P_1 <- swiss[which((relig < 20) & (agr > med_agr_P) & (mort < mort_1st_quart_P)), 1]
P_2 <- swiss[which((relig < 20) & (agr < med_agr_P) & (mort < mort_1st_quart_P)), 1]
P_3 <- swiss[which((relig < 20) & (agr > med_agr_P) & (mort > mort_1st_quart_P)), 1]
P_4 <- swiss[which((relig < 20) & (agr < med_agr_P) & (mort > mort_1st_quart_P)), 1]

# Нормальные люди
M_1 <- swiss[which((relig>=20)&(relig<=80) & (agr > med_agr_M) & (mort < mort_1st_quart_M)), 1]
M_2 <- swiss[which((relig>=20)&(relig<=80) & (agr < med_agr_M) & (mort < mort_1st_quart_M)), 1]
M_3 <- swiss[which((relig>=20)&(relig<=80) & (agr > med_agr_M) & (mort > mort_1st_quart_M)), 1]
M_4 <- swiss[which((relig>=20)&(relig<=80) & (agr < med_agr_M) & (mort > mort_1st_quart_M)), 1]
```

```r
avg_C <- c(mean(C_1), mean(C_2), mean(C_3), mean(C_4))
avg_P <- c(mean(P_1), mean(P_2), mean(P_3), mean(P_4))
avg_M <- c(mean(M_1), mean(M_2), mean(M_3), mean(M_4))

avg_C
avg_P
avg_M

friedman.test(cbind(avg_C, avg_P, avg_M))
# Это значит, что нулевую гипотезу мы не отклоняем, и медианы действительно можно считать равными
```

```
>
> avg_C <- c(mean(C_1), mean(C_2), mean(C_3), mean(C_4))
> avg_P <- c(mean(P_1), mean(P_2), mean(P_3), mean(P_4))
> avg_M <- c(mean(M_1), mean(M_2), mean(M_3), mean(M_4))
>
> avg_C
[1] 77.56667 79.30000 78.66000 83.35714
> avg_P
[1] 62.68333 54.30000 67.75714 68.88333
> avg_M
[1]  NaN 35.0 68.3 42.8
>
> friedman.test(cbind(avg_C, avg_P, avg_M))

        Friedman rank sum test

data:  cbind(avg_C, avg_P, avg_M)
Friedman chi-squared = 4.6667, df = 2, p-value = 0.09697

> # Это значит, что нулевую гипотезу мы не отклоняем, и медианы действительно можно считать равн
ыми
> W = 88, p-value = 0.1175
```