Instrukcja do Projektu

Słowem wstępu do całkowitego projektu zaliczamy dwie składowe: dokument typu .R (projekt.R) + skrypt ze szczegółową instrukcją do niego. Głównym celem projektu było przeprowadzenie podstawowej analizy statystycznej dla danych medycznych, gdzie grupy poddane badaniu były niezależne. Grupy, które zostały poddane badaniu i na których działałam w programie to: CHOR1, CHOR2 oraz KONTROLA.

Aby sprawdzić działanie skryptu, należy na początek pobrać odpowiednie programy. Najbardziej zalecanym środowiskiem do używania języka R jest RStudio.

Na samym początku znajdują się poszczególne biblioteki, które należy załadować, aby wszystkie funkcje użyte w programie poprawnie działały. Następnie możemy przejść do dalszej części projektu jakim jest wczytanie dokumentu typu csv do naszego programu.

I. Pierwsze zadanie dotyczyło usunięcia braków danych, które został wczytane z resztą danych do wektora **dane_med.** Zostały one zastąpione poprzez zaimputowanie/dodanie w miejsca 'NA' wartości średniej. W taki sposób pozbyliśmy się luk w naszej tabeli danych, bez zbędnego i całkowitego usuwania komórek z brakami. Poniżej przykład jak dokładnie wyglądało to przed i po, aby lepiej zobrazować jak konkretnie to działa.

Tabelka przed usunięciem braków:

	grupa	plec	wiek	hsCRP	ERY	PLT	HGB	HCT	MCHC	MON	LEU
1	CHOR1	k	36	2.711000	4.19	201	13.21020	0.3920	34.71490	0.48	11.86
2	CHOR1	m	39	4.699380	4.48	222	13.04910	0.3800	35.37930	0.76	10.32
3	CHOR1	k	35	2.353540	3.59	278	10.14930	0.3210	32.55560	1.08	13.60
4	CHOR1	m	29	2.271610	3.66	200	11.27700	0.3360	34.54880	0.63	10.11
5	CHOR1	m	29	4.465190	4.41	128	12.40470	0.3630	35.21320	NA	10.55
6	CHOR1	m	43	6.162690	3.68	176	11.43810	0.3400	34.71490	0.83	9.28
7	CHOR1	k	29	4.988360	4.12	288	12.24360	0.3570	35.37930	0.90	10.07
8	CHOR1	k	26	1.849380	4.44	231	13.21020	0.3980	34.21660	0.74	9.56
9	CHOR1	m	23	20.154800	4.13	153	12.56580	0.3840	35.59523	1.07	14.48
10	CHOR1	m	23	3.204050	4.02	249	11.92140	0.3530	34.88100	1.07	10.51
11	CHOR1	m	24	0.487607	4.07	177	11.92140	0.3500	35.04710	0.61	6.79
12	CHOR1	k	30	2.322680	4.11	295	12.24360	0.3600	35.04710	0.72	14.97
13	CHOR1	m	26	16.406900	4.18	174	NA	0.3340	36.37590	1.50	16.00
14	CHOR1	k	27	3.044270	4.59	207	13.85460	0.3940	36.20980	0.59	9.23
15	CHOR1	m	30	42.649900	4.20	170	12.40470	0.3640	35.21320	1.52	16.81

przykładowe, widoczne miejsca w których występuje NA: w kolumnie o nazwie HGB oraz MON, zostały zaznaczone w czerwonej pętelce

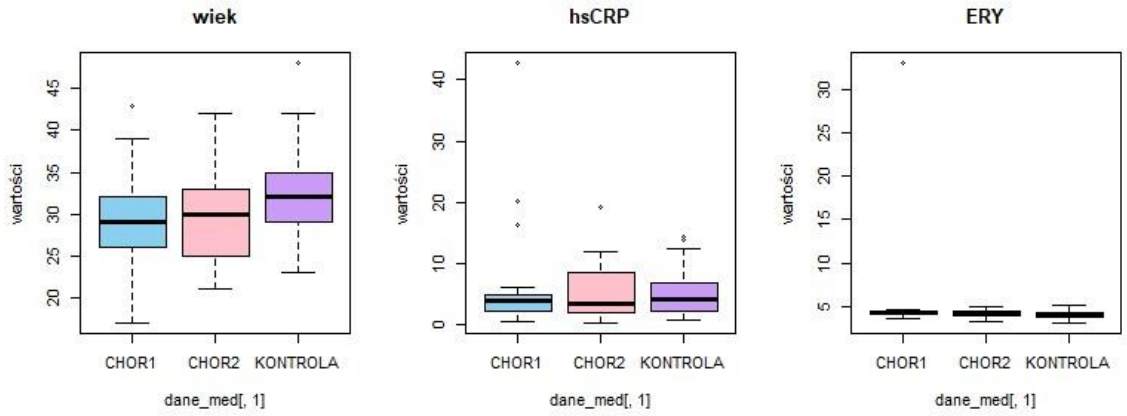
Tabelka po usunięciu braków:

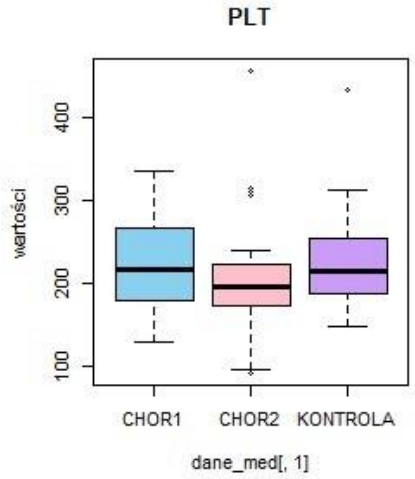
	grupa	plec	wiek	hsCRP	ERY	PLT	HGB	HCT	MCHC	MON	LEU
1	CHOR1	k	36	2.711000	4.19	201	13.21020	0.3920	34.71490	0.480000	11.86
2	CHOR1	m	39	4.699380	4.48	222	13.04910	0.3800	35.37930	0.760000	10.32
3	CHOR1	k	35	2.353540	3.59	278	10.14930	0.3210	32.55560	1.080000	13.60
4	CHOR1	m	29	2.271610	3.66	200	11.27700	0.3360	34.54880	0.630000	10.11
5	CHOR1	m	29	4.465190	4.41	128	12.40470	0.3630	35.2132[illegible]	0.857027	10.55
6	CHOR1	m	43	6.162690	3.68	176	11.43810	0.3400	34.71490	0.830000	9.28
7	CHOR1	k	29	4.988360	4.12	288	12.24360	0.3570	35.37930	0.900000	10.07
8	CHOR1	k	26	1.849380	4.44	231	13.21020	0.3980	34.21660	0.740000	9.56
9	CHOR1	m	23	20.154800	4.13	153	12.56580	0.3840	35.59523	1.070000	14.48
10	CHOR1	m	23	3.204050	4.02	249	11.92140	0.3530	34.88100	1.070000	10.51
11	CHOR1	m	24	0.487607	4.07	177	11.92140	0.3500	35.04710	0.610000	6.79
12	CHOR1	k	30	2.322680	4.11	295	12.24360	0.3600	35.04710	0.720000	14.97
13	CHOR1	m	26	16.406900	4.18	174	12.16923	0.3340	36.37590	1.500000	16.00
14	CHOR1	k	27	3.044270	4.59	207	13.85460	0.3940	36.20980	0.590000	9.23
15	CHOR1	m	30	42.649900	4.20	170	12.40470	0.3640	35.21320	1.520000	16.81

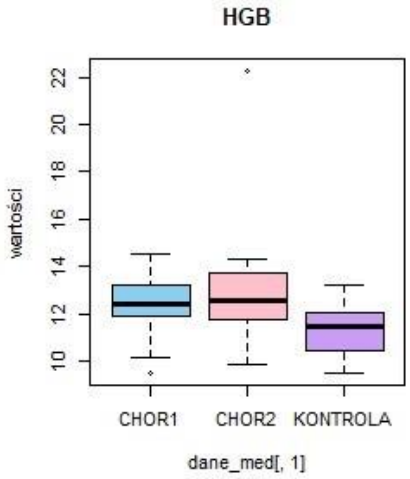
tabelka po uzupełnieniu pustych komórek wartością średniej

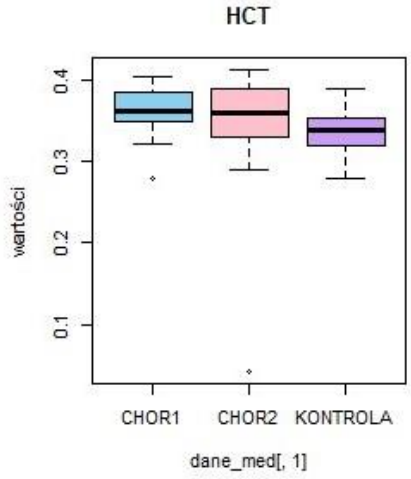
Na przykładzie zostały oznaczone dokładnie te same miejsca, gdzie wcześniej pojawiała się wartość NA. A teraz po krótce wyjaśniona zostanie pętla, dzięki uzyskaliśmy taki wynik. Dokładnie w pętli for zaczynamy przechodzenie po naszych kolumnach od 1 do ostatniej w celu znalezienia potencjalnych braków. Następnie w instrukcji warunkowej sprawdzamy funkcją sum() oraz is.na() ile dokładnie pozycji (dzięki sumowaniu) w tabeli, ma wartość NA, czyli wartość pustą. Instrukcja if wykona się kompletnie, w sytuacji gdy suma NA w kolumnie będzie większa niż 0. Jeśli suma będzie wynosić 0 oznacza to, że kolumna jest kompletna i w żadnej komórce nie brakuje żadnej z wartości. Natomiast w przypadku kiedy suma będzie > 0, zostaną automatycznie zastąpione brakujące miejsca/wartości w komórkach - wartością średniej.

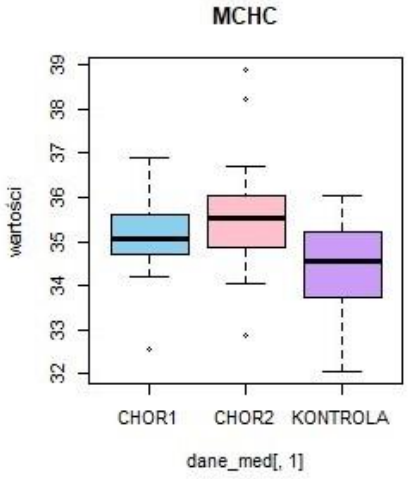
Kolejnym podpunktem w tym zadaniu było graficzne przedstawienie wartości odstających dla wybranych parametrów i dla każdej z grup. W programie zostało to zareprezentowanie za pomocą boxplota, który w programie będzie wyglądał w następujący sposób:

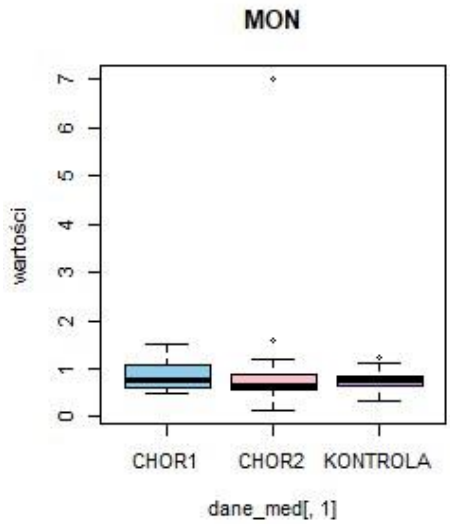


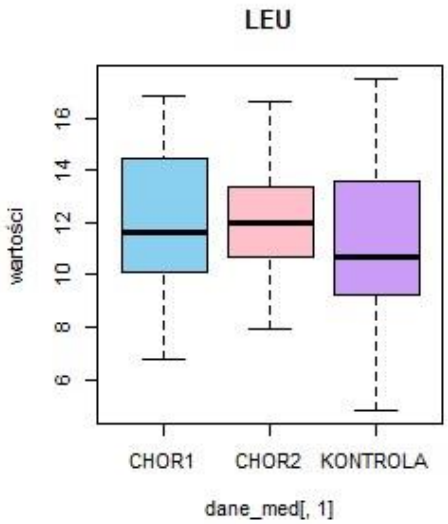










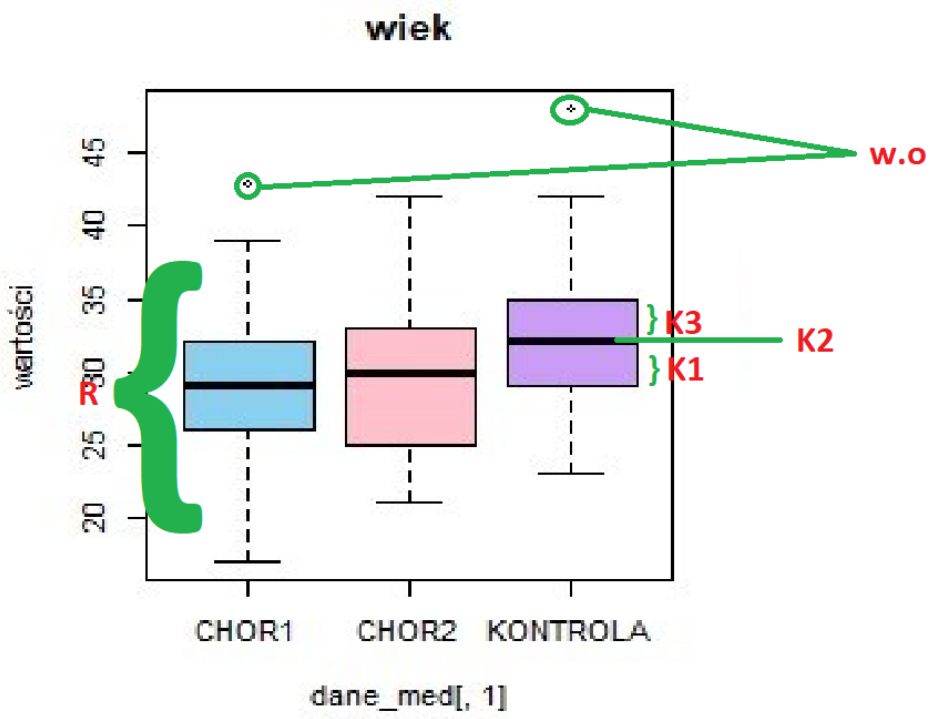


Na wykresach dokładnie widzimy wartości jak się mają wartości odstające w poszczególnych parametrach dla poszczególnych grup w stosunku do wartości. Na każdym z wykresów są wszystkie trzy badane w projekcie grupy. A teraz po krótce jak należy czytać i interpretować

taki wykres.Po lewej stronie na osi y mamy wartości, natomiast na osi x znajdują się nazwy poszczególnych grup, w naszym przypadku jest to CHOR1, CHOR2 oraz KONTROLA. Czarną widoczną linią na każdym z 'boxów' dla poszczególnej grupy jest mediana. Wszystko to co znajduję się pod czarną linią - medianą jest pierwszym kwartylem. Media jest drugim kwartylem natomiast trzecim kwartylem jest sama góra naszych boxów.

*dodatkowa informacja: za pomocą funkcji IQR, moglibyśmy poznać różnicę między I a III kwartylem.

Na podstawie jednego z wykresu, poniżej zostało rozpisane jak konkretnie należy interpretować taki wykres i gdzie są poszczególne kwartyle jak i wartości odstające.



Gdzie poszczególne oznaczenia interpretujemy:
R -> rozstępy międzykwartylowe
w.o -> wartości odstające
K1 -> kwartyl I
K2 -> kwartyl II (mediana)
K3 -> kwartyl III

Jedną z ciekawszych rzeczy, które możemy gołym okiem zauważyć jest to, że na 9 wykresów 8 z nich posiada wartości odstające, natomiast jest jeden dokładnie dla parametru LEU, który nie posiada wartości odstających.

II. Drugim punktem, który należało wykonać była charakterystyka poszczególnych grup w formie/strukturze tabelarycznej. Do tego celu podzieliliśmy naszą tabele z uzupełnionymi już danymi (bez braków), według grup. Następnie nasze dane z rozszczepionej tabelki według grup, przekazujemy za pomocą potoku (%>%) wartości do funkcji map(summary). Jeśli funkcja będzie zwracała wektor, to funkcja map zwróci nam listę i w tym przypadku tak się dzieje.

Funkcja summary() pozwala nam na podsumowanie wektora. W rezultacie otrzymamy każdą z grup osobno i każdy parametr dla każdej grupy, będzie w osobnej tabelce. Wygląda to w następujący sposób:

```
$CHOR1
    grupa              plec               wiek           hsCRP              ERY
 Length:25          Length:25          Min.   :17.00   Min.   : 0.4876   Min.   : 3.530
 Class :character   Class :character   1st Qu.:26.00   1st Qu.: 2.3227   1st Qu.: 4.070
 Mode  :character   Mode  :character   Median :29.00   Median : 3.9665   Median : 4.200
                                       Mean   :29.56   Mean   : 6.1030   Mean   : 5.363
                                       3rd Qu.:32.00   3rd Qu.: 4.9935   3rd Qu.: 4.510
                                       Max.   :43.00   Max.   :42.6499   Max.   :33.000
      PLT              HGB             HCT              MCHC             MON
 Min.   :128.0   Min.   : 9.505   Min.   :0.2800   Min.   :32.56   Min.   :0.4800
 1st Qu.:179.0   1st Qu.:11.921   1st Qu.:0.3500   1st Qu.:34.71   1st Qu.:0.6100
 Median :217.0   Median :12.405   Median :0.3630   Median :35.05   Median :0.7600
 Mean   :225.3   Mean   :12.402   Mean   :0.3636   Mean   :35.13   Mean   :0.8579
 3rd Qu.:266.0   3rd Qu.:13.210   3rd Qu.:0.3860   3rd Qu.:35.60   3rd Qu.:1.0700
 Max.   :336.0   Max.   :14.499   Max.   :0.4050   Max.   :36.87   Max.   :1.5200
      LEU
 Min.   : 6.79
 1st Qu.:10.11
 Median :11.66
 Mean   :12.02
 3rd Qu.:14.48
 Max.   :16.81
```

```
$CHOR2
    grupa              plec               wiek           hsCRP              ERY
 Length:25          Length:25          Min.   :21.00   Min.   : 0.3351   Min.   :3.250
 Class :character   Class :character   1st Qu.:25.00   1st Qu.: 2.0781   1st Qu.:3.850
 Mode  :character   Mode  :character   Median :30.00   Median : 3.4455   Median :4.270
                                       Mean   :30.04   Mean   : 5.5360   Mean   :4.198
                                       3rd Qu.:33.00   3rd Qu.: 8.6093   3rd Qu.:4.430
                                       Max.   :42.00   Max.   :19.2124   Max.   :5.040
      PLT              HGB             HCT              MCHC             MON
 Min.   : 91.0   Min.   : 9.827   Min.   :0.0423   Min.   :32.89   Min.   :0.1400
 1st Qu.:172.0   1st Qu.:11.760   1st Qu.:0.3300   1st Qu.:34.88   1st Qu.:0.5500
 Median :195.0   Median :12.566   Median :0.3600   Median :35.55   Median :0.6600
 Mean   :209.1   Mean   :12.806   Mean   :0.3460   Mean   :35.55   Mean   :0.9528
 3rd Qu.:223.0   3rd Qu.:13.694   3rd Qu.:0.3900   3rd Qu.:36.04   3rd Qu.:0.8800
 Max.   :456.0   Max.   :22.232   Max.   :0.4120   Max.   :38.87   Max.   :7.0000
      LEU
 Min.   : 7.95
 1st Qu.:10.70
 Median :12.00
 Mean   :12.04
 3rd Qu.:13.34
 Max.   :16.59
```

```
$KONTROLA
    grupa              plec               wiek           hsCRP              ERY
 Length:25          Length:25          Min.   :23.00   Min.   : 0.7584   Min.   :3.090
 Class :character   Class :character   1st Qu.:29.00   1st Qu.: 2.3022   1st Qu.:3.820
 Mode  :character   Mode  :character   Median :32.00   Median : 4.2204   Median :3.980
                                       Mean   :32.32   Mean   : 5.2951   Mean   :4.013
                                       3rd Qu.:35.00   3rd Qu.: 6.8521   3rd Qu.:4.330
                                       Max.   :48.00   Max.   :14.3951   Max.   :5.050
      PLT              HGB             HCT              MCHC             MON
 Min.   :147.0   Min.   : 9.505   Min.   :0.2790   Min.   :32.06   Min.   :0.3500
 1st Qu.:188.0   1st Qu.:10.472   1st Qu.:0.3200   1st Qu.:33.72   1st Qu.:0.6500
 Median :214.0   Median :11.438   Median :0.3390   Median :34.55   Median :0.7600
 Mean   :225.9   Mean   :11.300   Mean   :0.3376   Mean   :34.40   Mean   :0.7604
 3rd Qu.:254.0   3rd Qu.:12.082   3rd Qu.:0.3530   3rd Qu.:35.21   3rd Qu.:0.8600
 Max.   :434.0   Max.   :13.210   Max.   :0.3890   Max.   :36.04   Max.   :1.2500
      LEU
 Min.   : 4.83
 1st Qu.: 9.22
 Median :10.68
 Mean   :11.36
 3rd Qu.:13.59
 Max.   :17.46
```

III. W trzecim punkcie projektu, należało wykonać analizę porównawczą między grupami oraz określenie czy istnieją istotne różnice statystyczne. Na początku w jakiś sposób, trzeba było

“powyciągać” poszczególne nazwy kolumn w taki sposób, aby zrobić z nich zmienne. Czyli np.. “grupa” --> grupa
Zatem na samym początku jest stworzony wektor, do którego dodajemy nazwy kolumn z tabelki z danymi. Następnie, dzięki funkcji parse_expr() wyciągamy pojedynczo poszczególne nazwy do innego wektora, w tym przypadku jest to y. Na samym początku musimy skupić się na zgodności danych z rozkładem normalnym. Do tego posłuży nam test Shapiro-Wilka. Na samym początku została utworzona lista, która po przejściu pętli będzie przechowywała następujące dane: grupe, statystykę dla poszczególnej grupy oraz wartości p.value dla grup. W pętli w ifelsie, sprawdzona zostanie wartość p.value dla każdej grupy, aby sprawdzić czy rozkład danych różni się od rozkładu normalnego. Jeśli p.value > 0.05 to oznacza to, że rozkład naszych danych nie różni się jakoś bardzo znacząco od rozkładu normalnego, czyli możemy założyć normalność naszych danych.

Kolejnym krokiem jest przejście do oceny homogeniczności wariancji oraz testy statystyczne dla grup. Tutaj również tworzymy nową listę, która będzie przechowywała dane. W tym przypadku lista zawiera dane dotyczące tylko kolumn, w których występują numeryczne dane. Czyli np.. kolumny grupa i plec nie są brane pod uwagę. Jest to oczywiście warunek w pętli, który sprawdza czy dane w tabeli są numeryczne, jeśli tak to wykonuje się dalej. W teście Levene’a ponownie używamy już wyżej wspomnianej funkcji parse_expr(), dzięki której będziemy mogli bez bezpośredniego odwoływania się do poszczególnych kolumn zawierających numeryczne dane, odwołać. W tym przypadku mamy dwie zmienne, y i x gdzie y wiemy już co przechowuje, natomiast x będzie przechowywał nazwy badanych grup. Dzięki funkcji eval() dzięki której będziemy mogli odwołać się pojedynczo do każdego parametru. Tak jak w przypadku Shapiro w jednym ifelsie, sprawdzamy warunek czy dane są zgodne z założeniem o jednorodności wariancji. Więc jeśli wartość p.value > 0.05 to wtedy oznacza to, że dane są zgodne z założeniem o jednorodności wariancji, więc możemy przy tym założyć homogeniczność danych.

W tym punkcie, również musimy porównać nasze grupy i przeprowadzić odpowiednie testy. W przypadku naszych danych mamy 3 grupy, które są poddane testom. Zatem testy, które są głównie brane tutaj pod uwagę to test ANOVA oraz Kruskala-Wallisa.

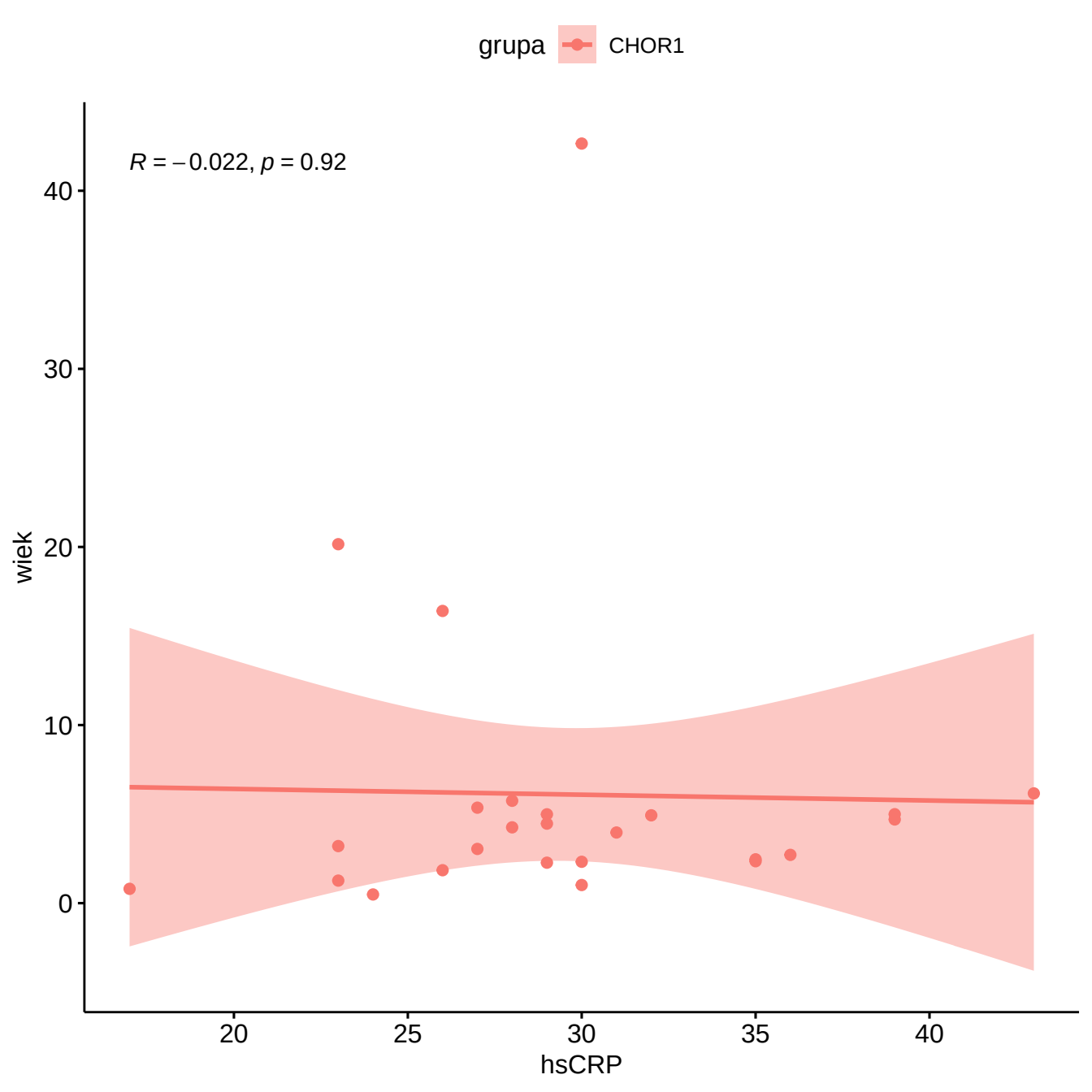
Test Kruskala-Wallisa używamy w momencie, jeśli dane nie spełniają założenia o zgodności z rozkładem normalnym, czyli kiedy p.value < 0.05. W takim przypadku do analizy porównawczej wykorzystuje się testy nieparametryczne, czyli wyżej już wspomniany test Kruskala-Wallisa. Ten sam test używamy również w przypadku, kiedy dane są zgodne z rozkładem normalnym, jednakże nie spełniają one założenia o jednorodności wariancji. W momencie, kiedy wartość p.value będzie mniejsza niż poziom istotności 0.05, można stwierdzić, że istnieją znaczące różnice między grupami. W programie również znajduje się wariant, w którym sprawdzamy między jakimi grupami występują różnice oraz jak duże one są. Za to w programie odpowiada funkcja dunnTest().

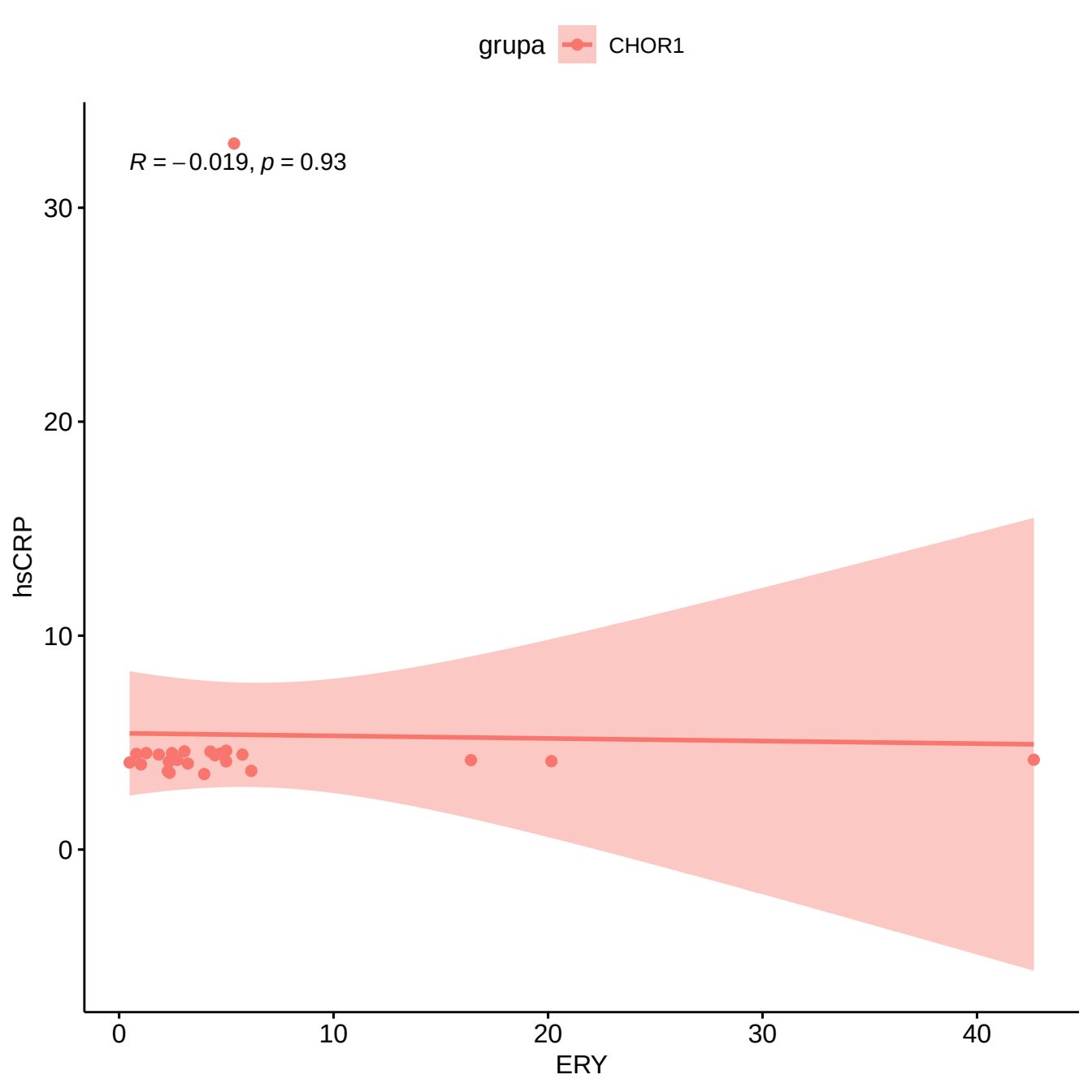
Jednakże musimy wziąć pod uwagę jeszcze jeden możliwy wariant danych. Jest to wariant w którym dane spełniają założenie o zgodności z rozkładem normalnym, czyli p.value > 0.05 oraz spełniają drugie z założeń, czyli to o jednorodności wariancji ---> p.value > 0.05. W takim przypadku stosuje się test ANOVA, który jest parametryczny. Tak jak w przypadku testu Kruskala-Wallisa, musimy rozpatrzeć przypadek, kiedy wartość p.value jest mniejsza niż poziom istotności (<0.05). Jeśli miałoby to miejsce można byłoby stwierdzić, że istnieją

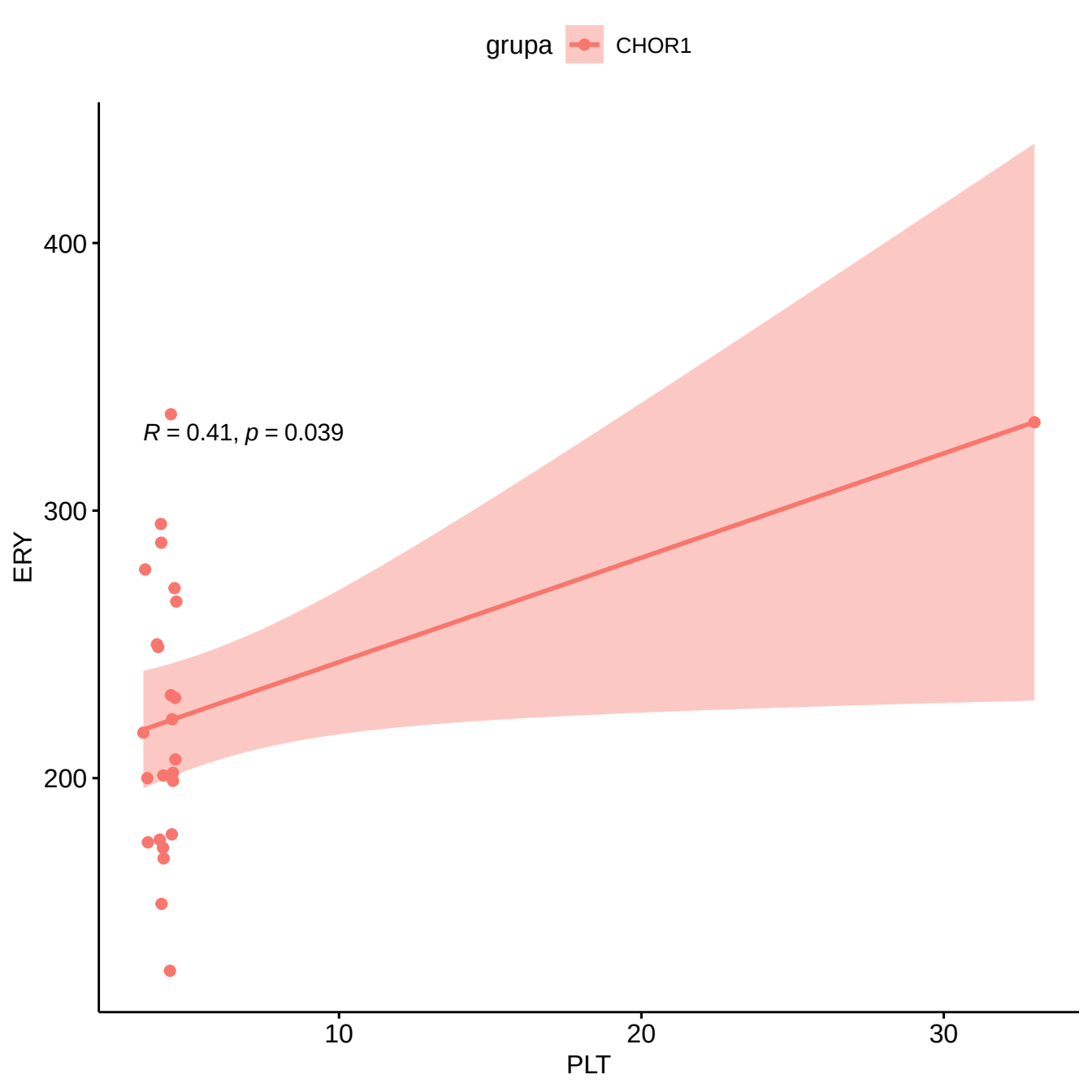
znaczące różnice między grupami. W programie za określenie między jakimi grupami występują te różnice, będzie odpowiadała funkcja TukeyHSD().

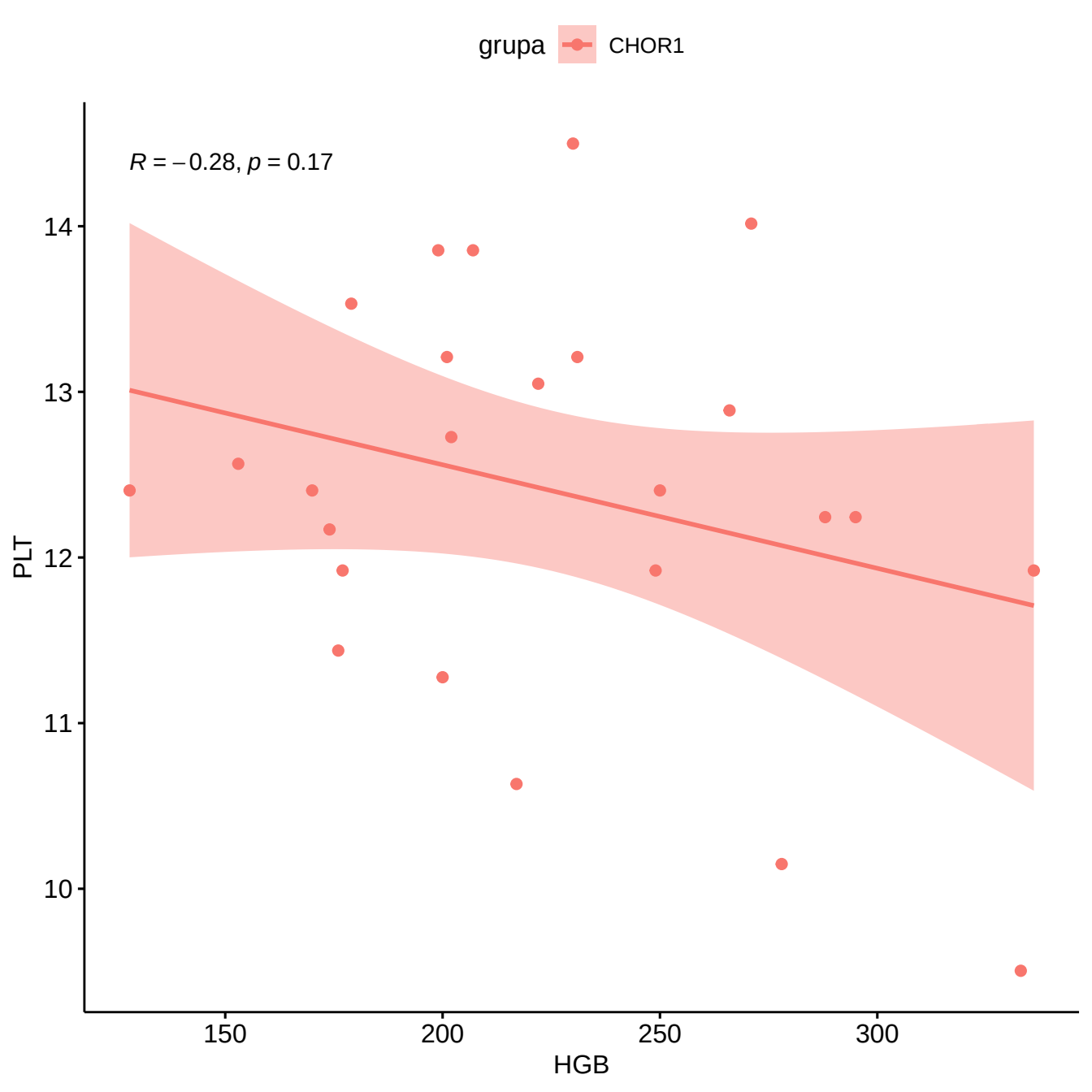
IV. W ostatnim punkcie projektu należało wykonać analizę korelacji, czyli określić między którymi parametrami i w obrębie jakich grup, występują istotne statystycznie korelacje. Dodatkowo należało określić siłę i kierunek korelacji. Stworzona została nowa zmienna testKOR, która przechowuje oddzielnie każdą grupę i wartości parametrów dla poszczególnej grupy. Jest to dokładnie ten sam krok, który pojawił się w punkcie 2 projektu. W pętli idziemy po kolumnach, tak aby potem móc zrobić korelacje i móc wskazać między jakimi grupami została wykonana analiza.

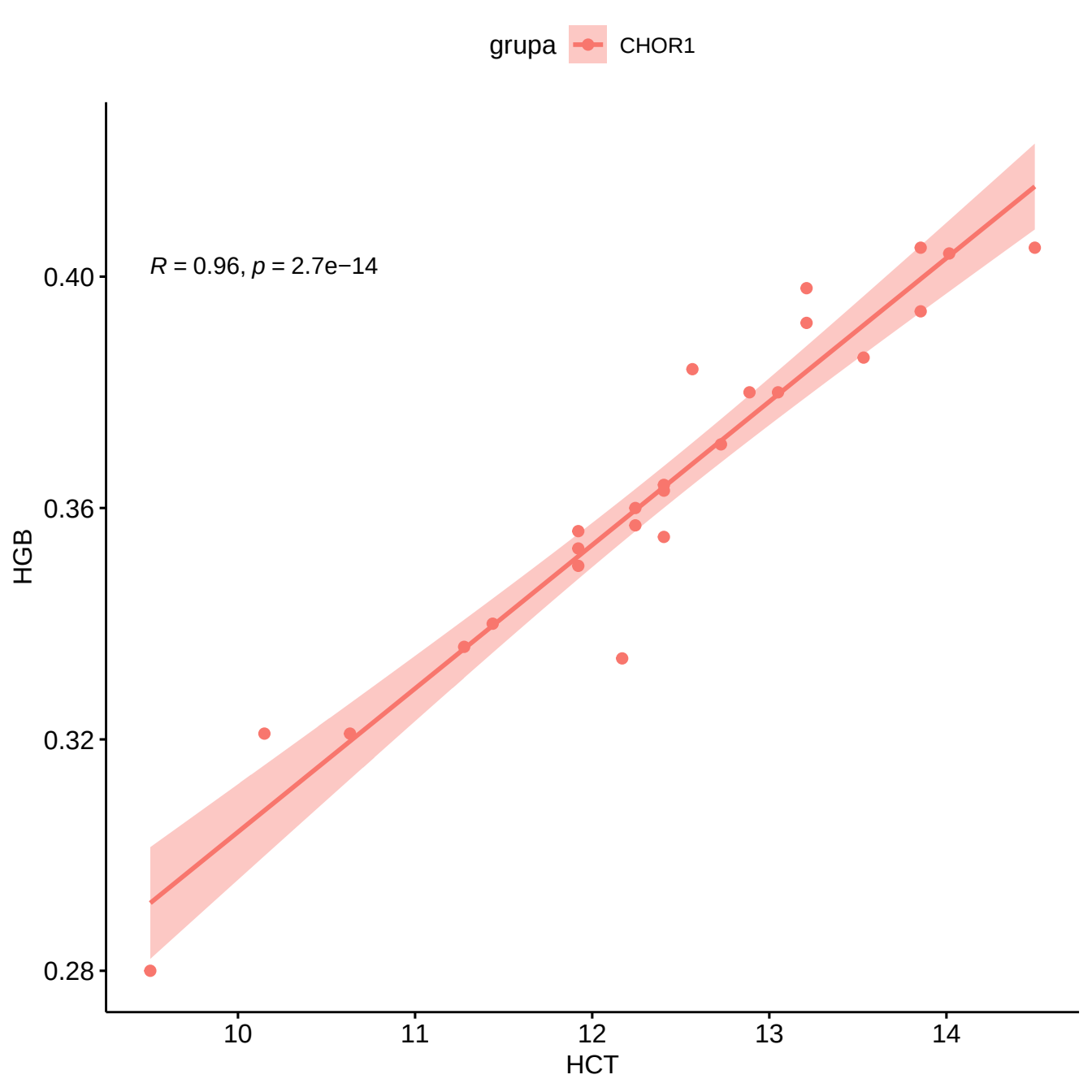
Dodatkowym punktem do analizy są wykresy, które zostały wykonane za pomocą funkcji ggscatter. Wykresy zostały zapisane do pliku typu pdf i umieszczone poniżej.

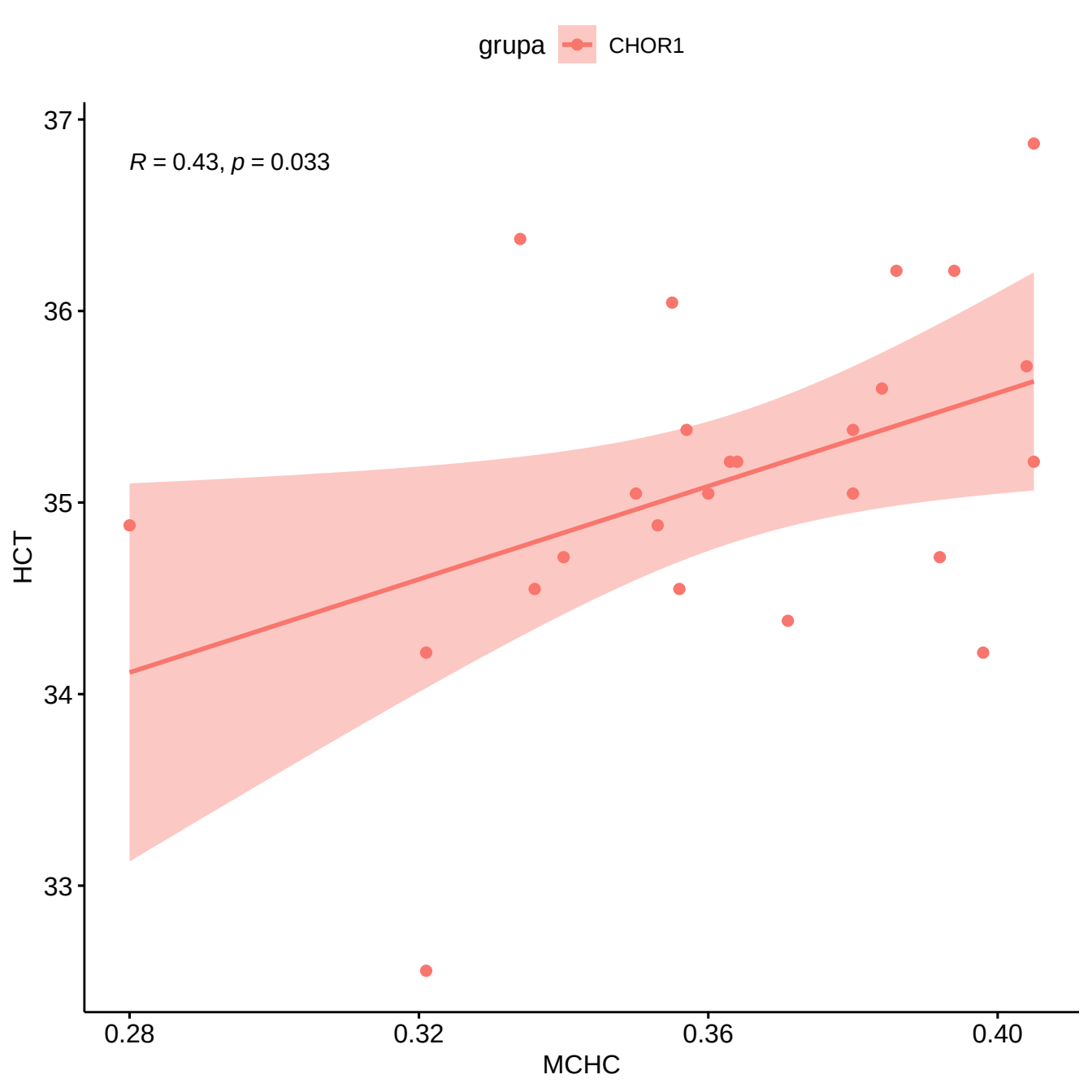

grupa
CHOR1
R = −0.022, p = 0.92
wiek
hsCRP
40
30
20
10
0
20
25
30
35
40

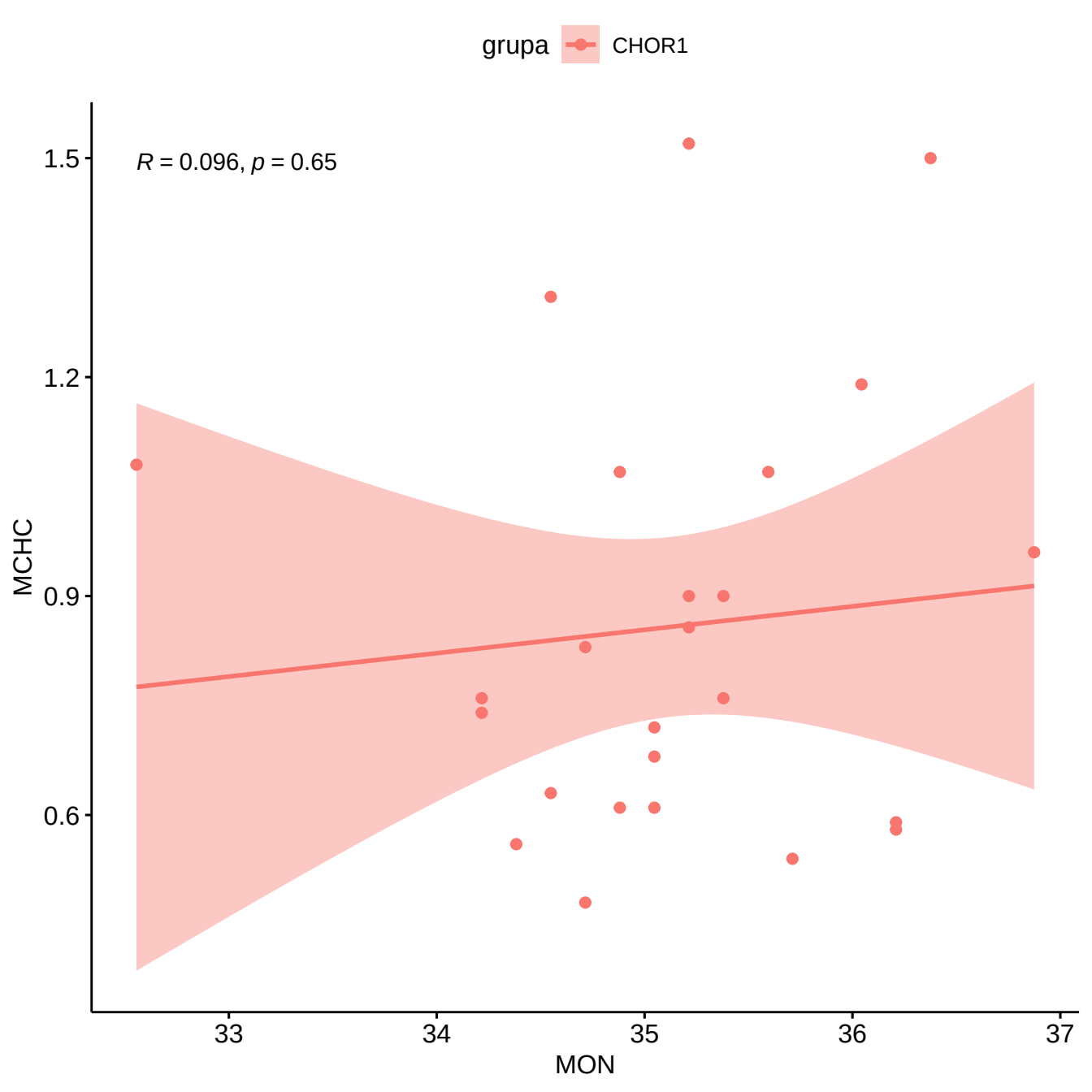

grupa
CHOR1
R = −0.019, p = 0.93
hsCRP
ERY
0
10
20
30
40

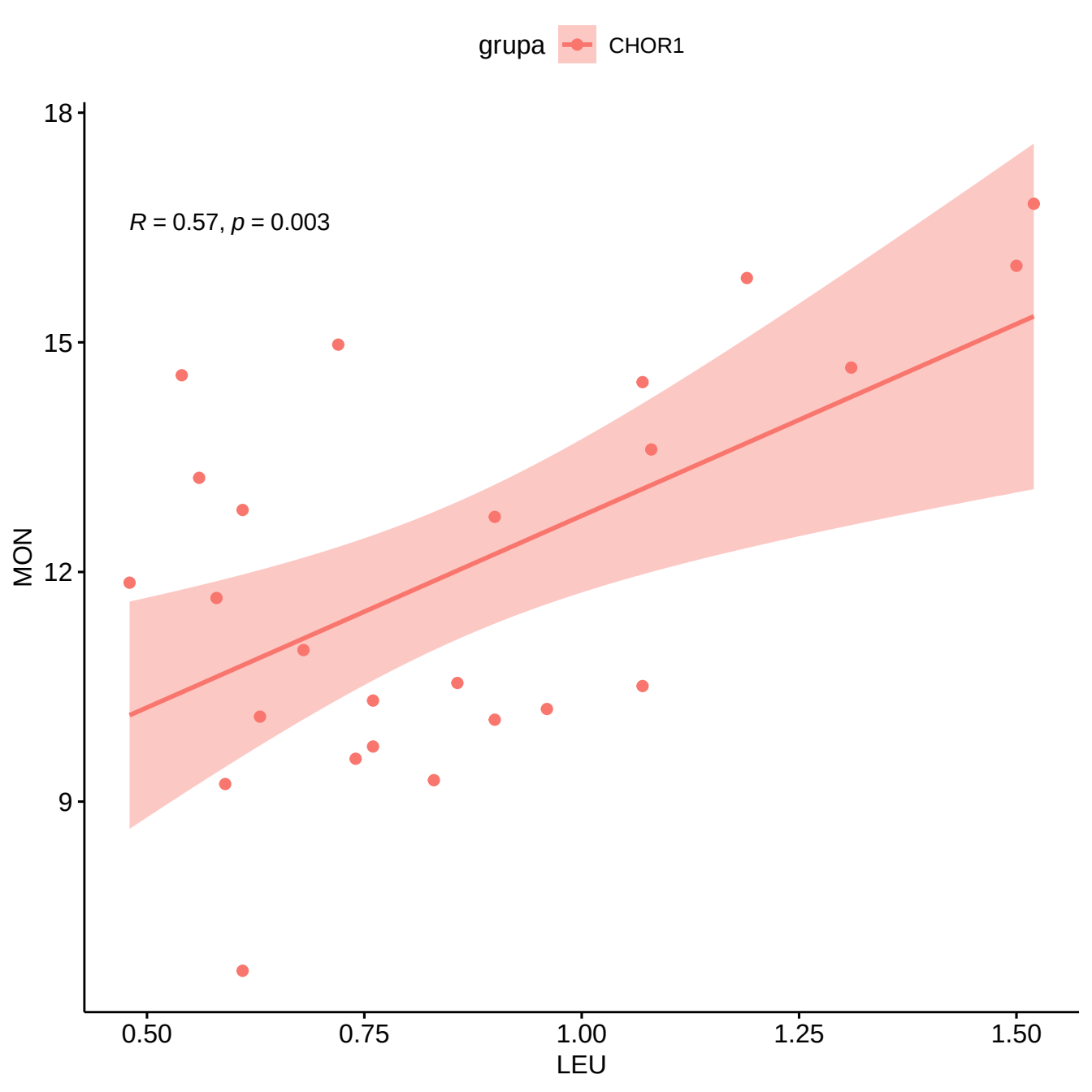

grupa
CHOR1
400
300
200
ERY
R = 0.41, p = 0.039
10
20
30
PLT

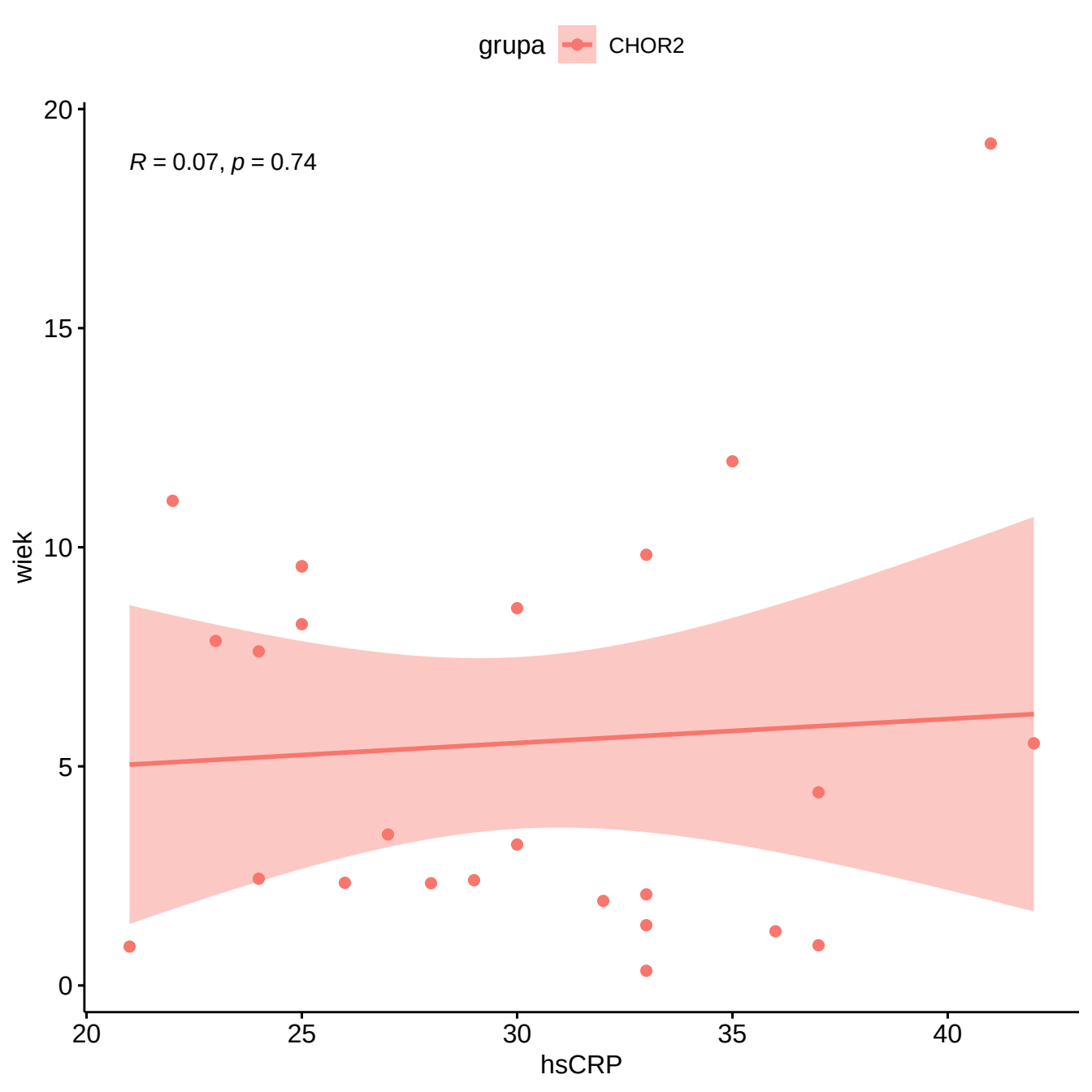

grupa
CHOR1
R = −0.28, p = 0.17
14
13
12
11
10
PLT
150
200
250
300
HGB

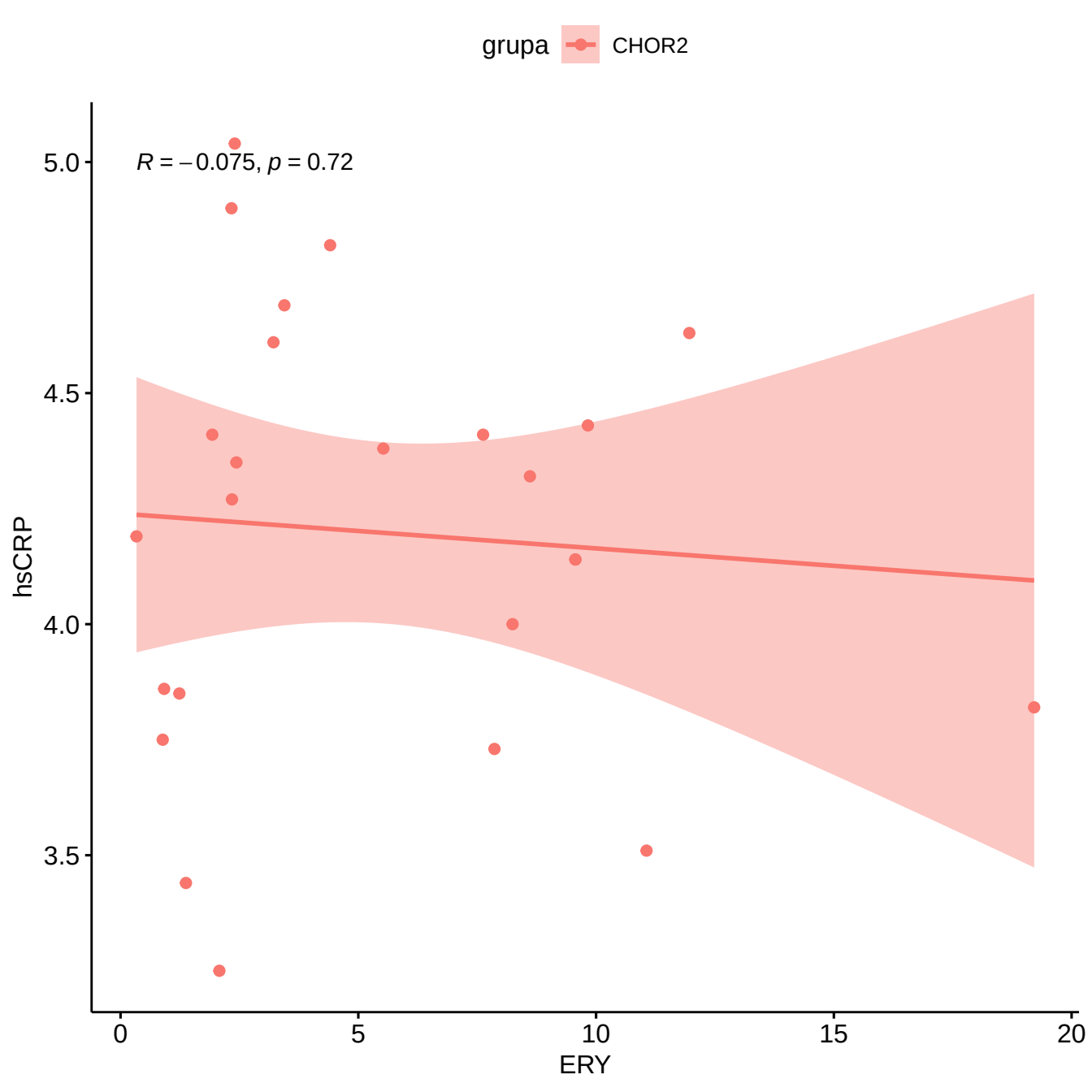

grupa
CHOR1
R = 0.96, p = 2.7e−14
0.40
0.36
0.32
0.28
HGB
10
11
12
13
14
HCT

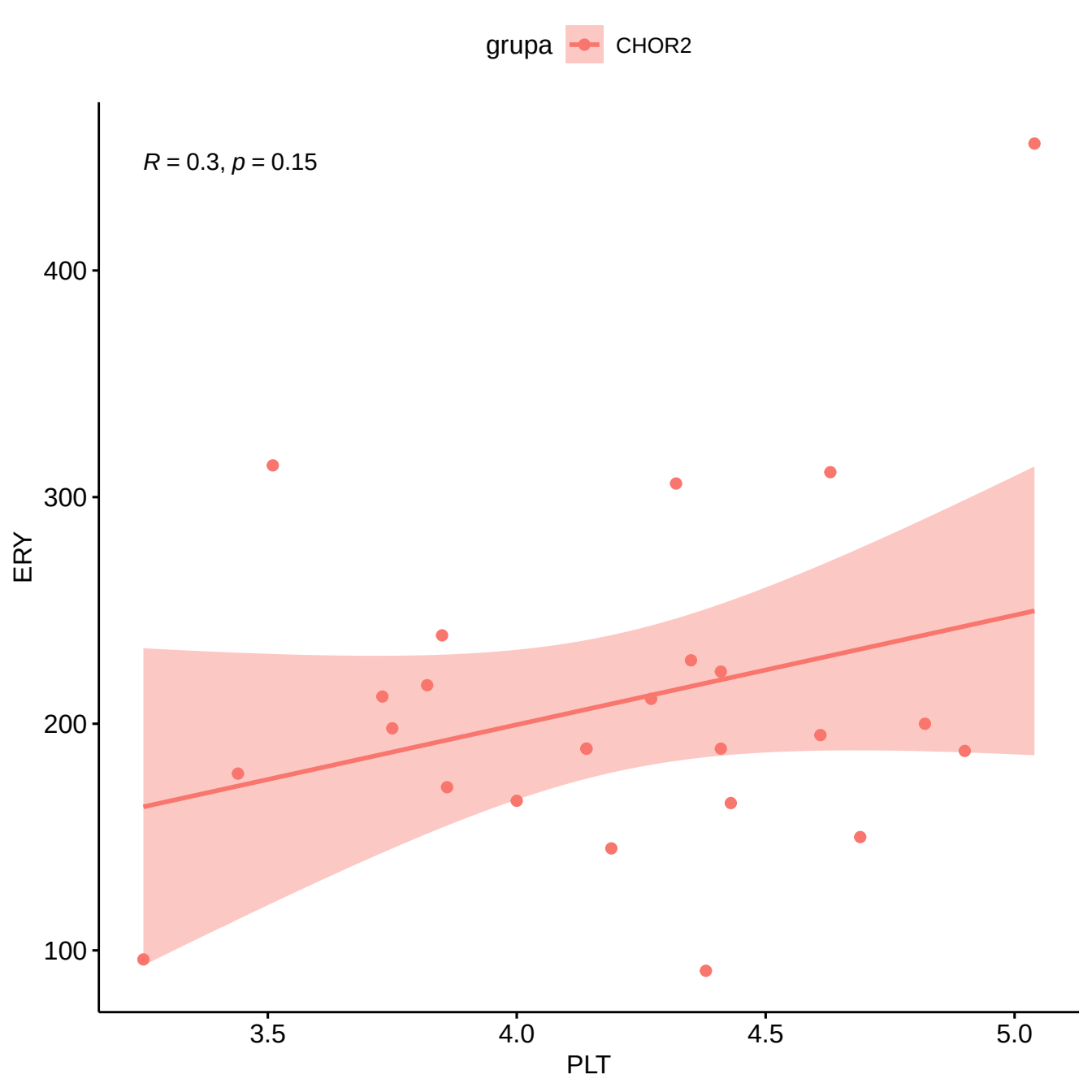

grupa
CHOR1
R = 0.43, p = 0.033
HCT
MCHC
37
36
35
34
33
0.28
0.32
0.36
0.40

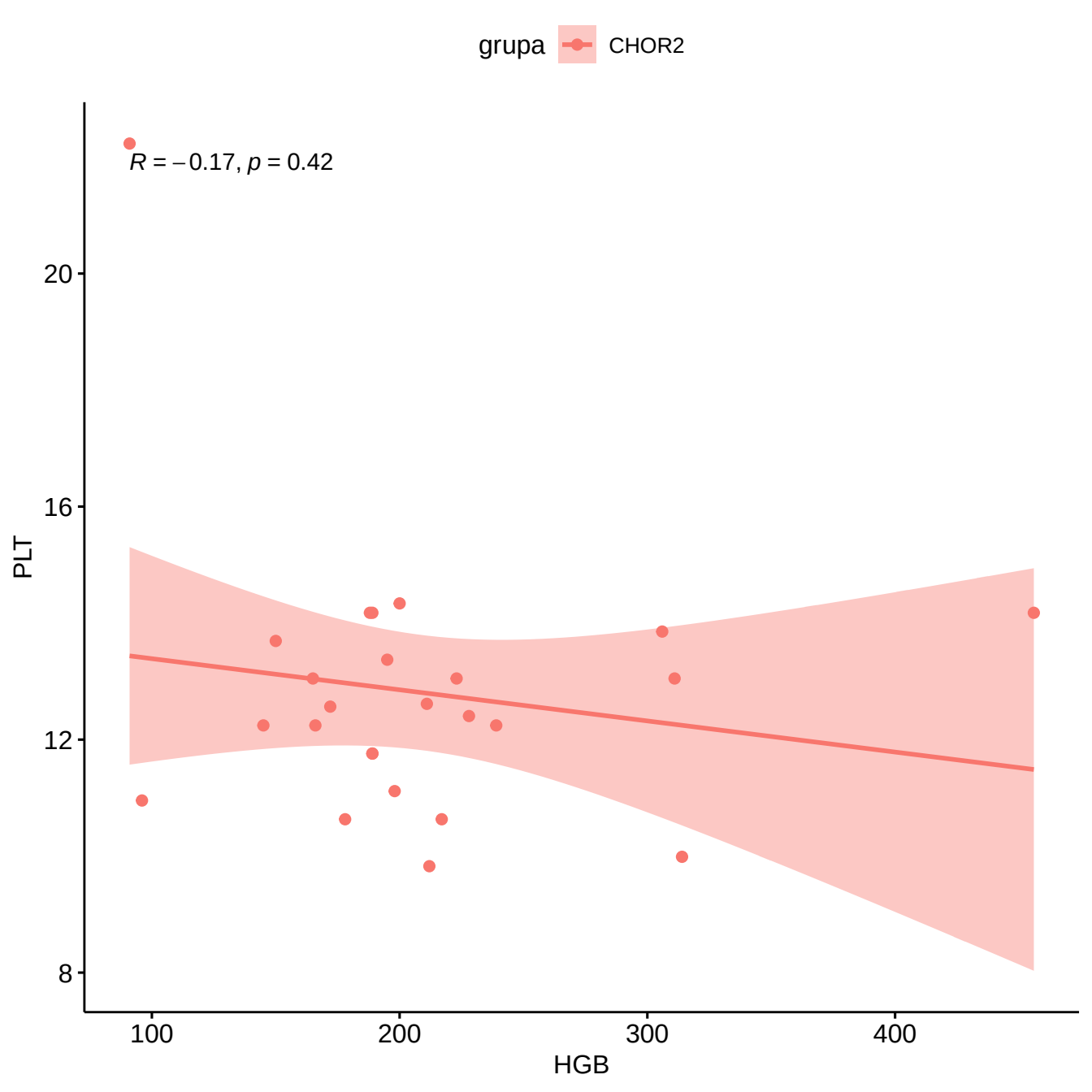

grupa
CHOR1
R = 0.096, p = 0.65
1.5
1.2
0.9
0.6
MCHC
33
34
35
36
37
MON

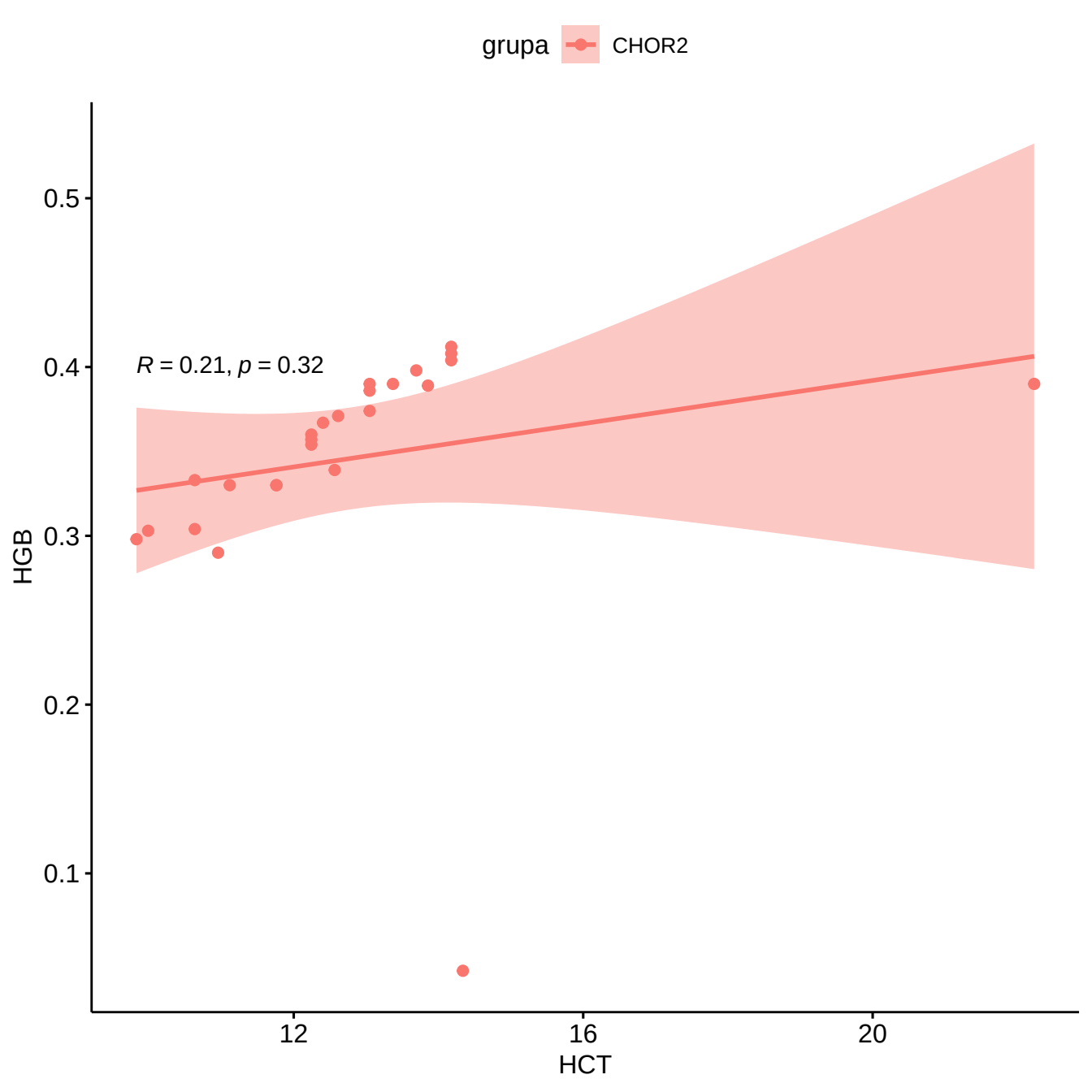

grupa
CHOR1
18
15
12
9
MON
R = 0.57, p = 0.003
0.50
0.75
1.00
1.25
1.50
LEU

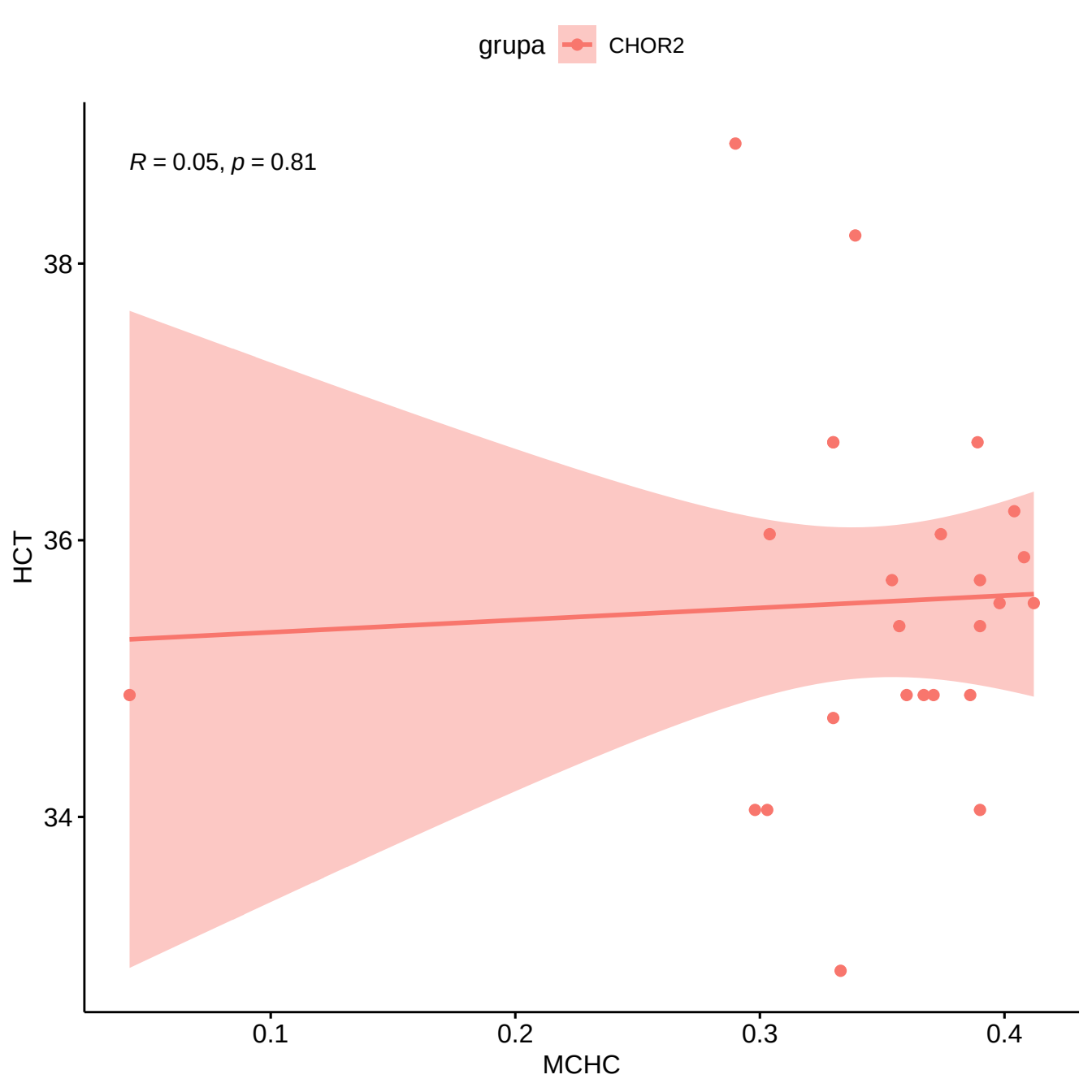

grupa
CHOR2
R = 0.07, p = 0.74
wiek
hsCRP
20
15
10
5
0
20
25
30
35
40

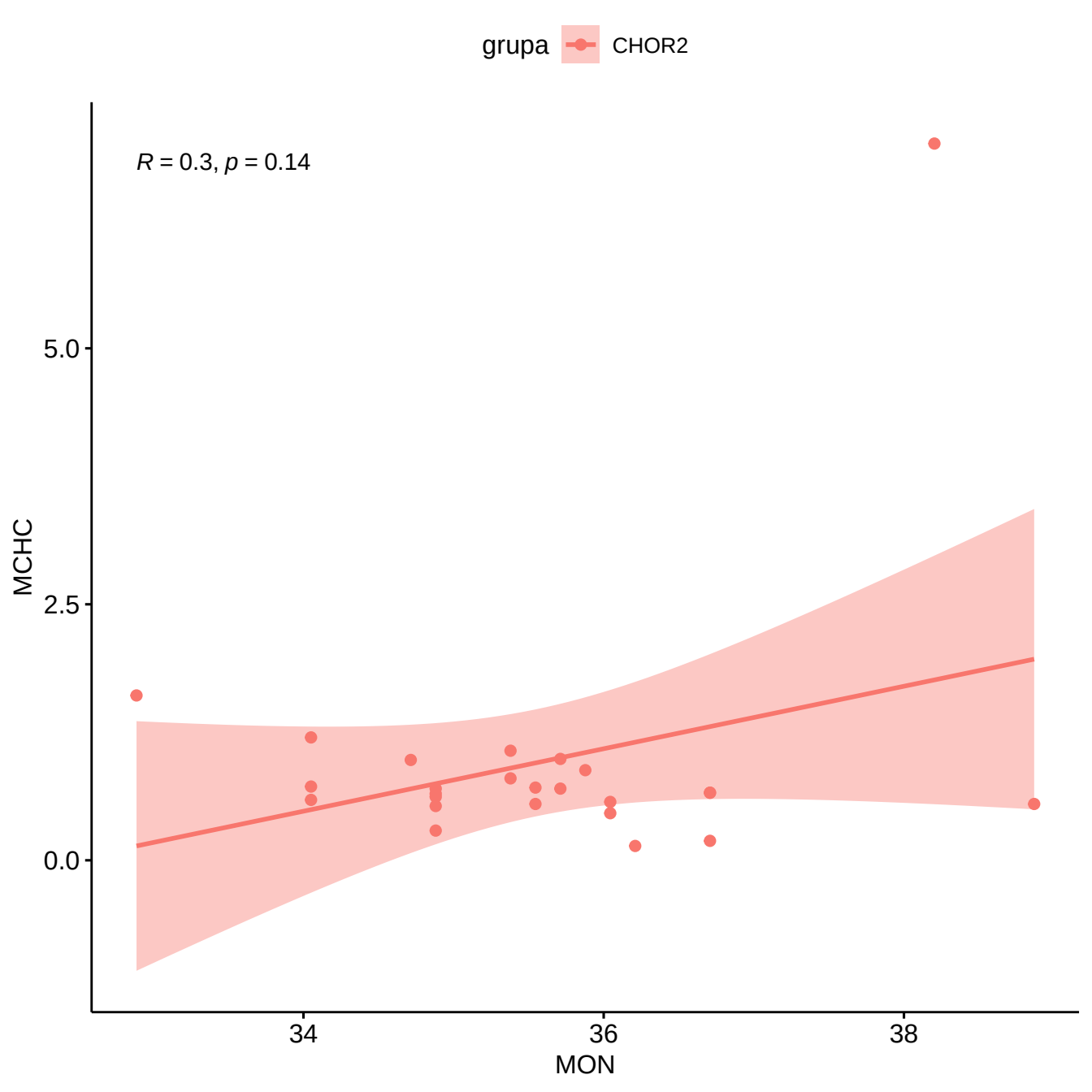

grupa
CHOR2
R = −0.075, p = 0.72
hsCRP
ERY
5.0
4.5
4.0
3.5
0
5
10
15
20

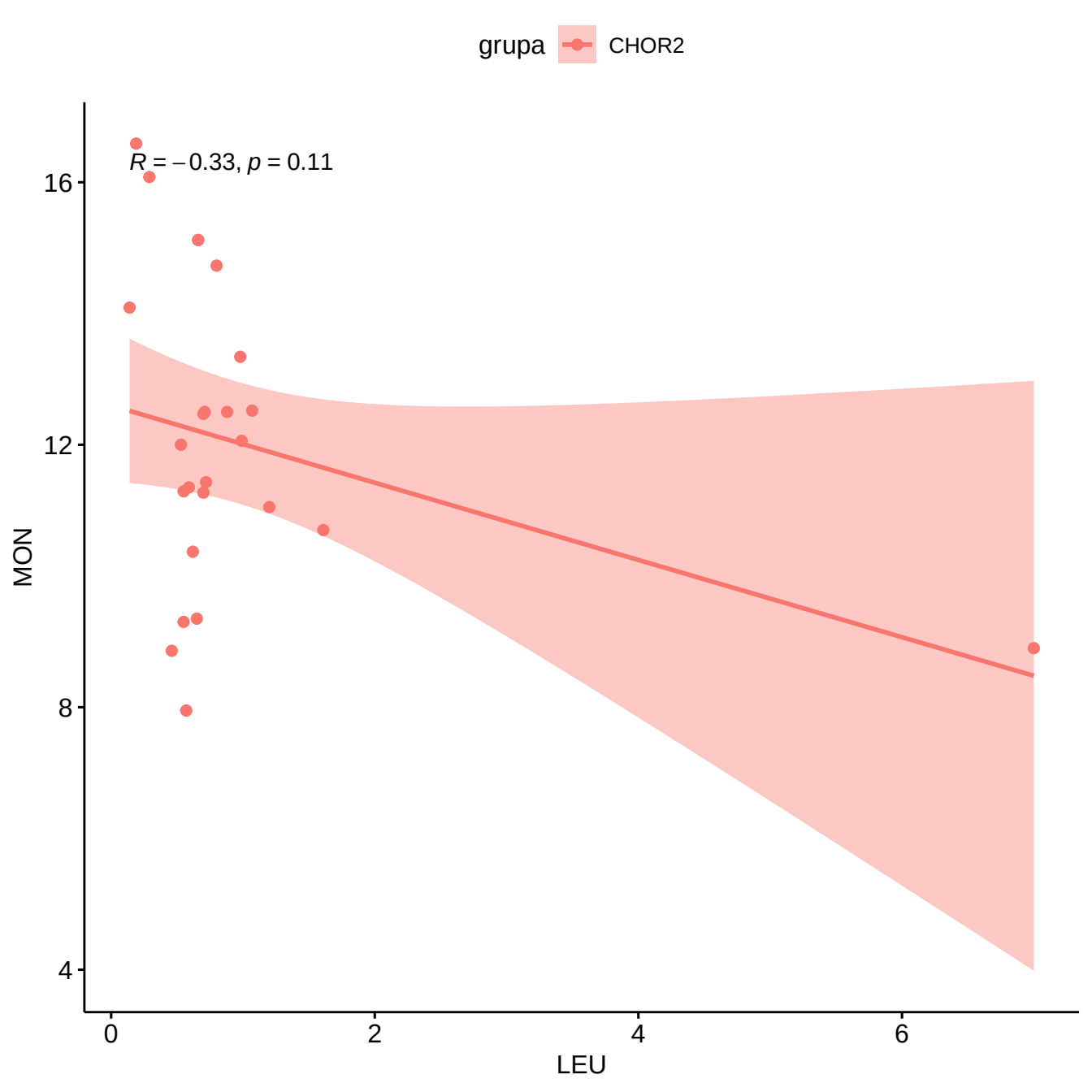

grupa
CHOR2
R = 0.3, p = 0.15
400
300
200
100
ERY
3.5
4.0
4.5
5.0
PLT

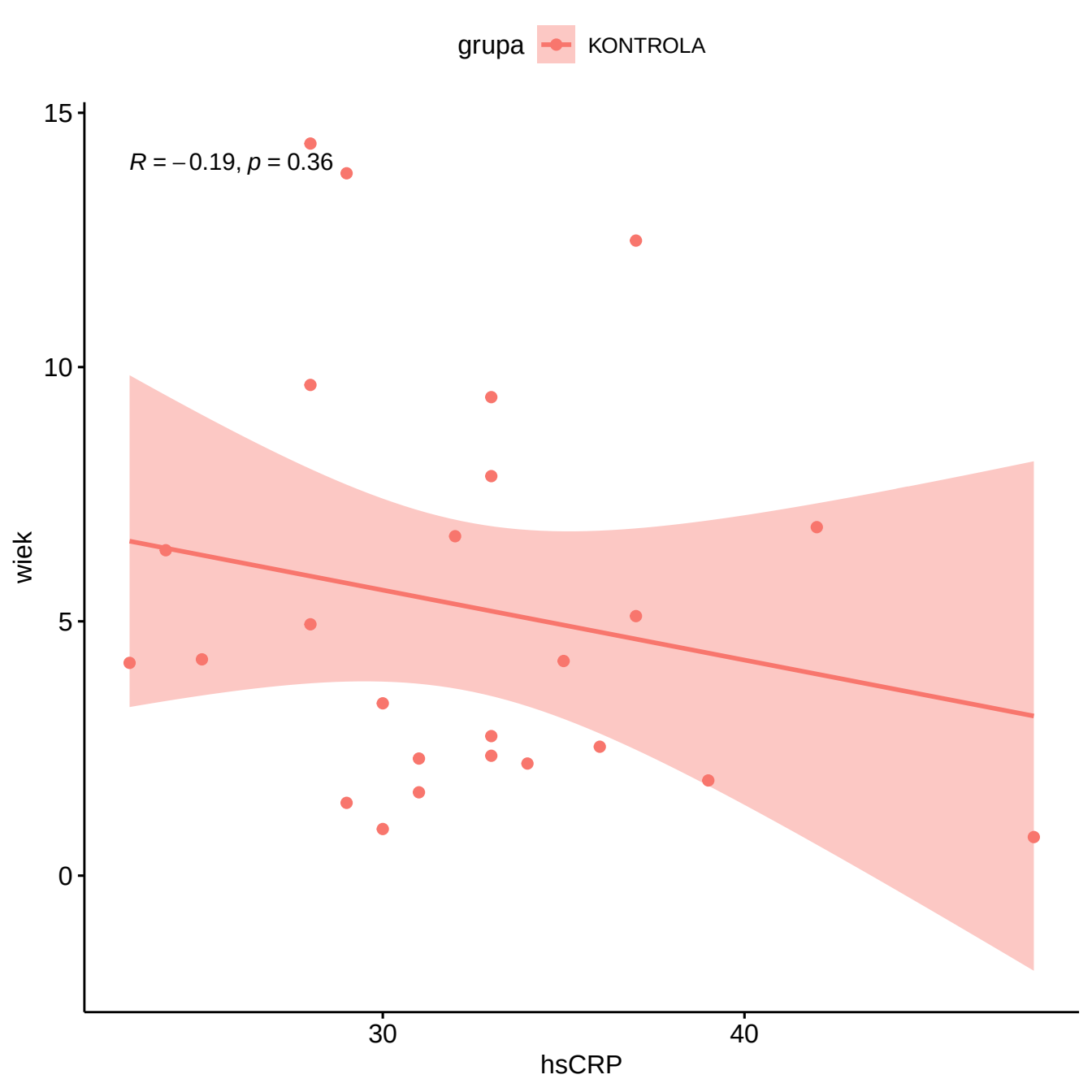

grupa
CHOR2
R = −0.17, p = 0.42
20
16
12
8
PLT
100
200
300
400
HGB

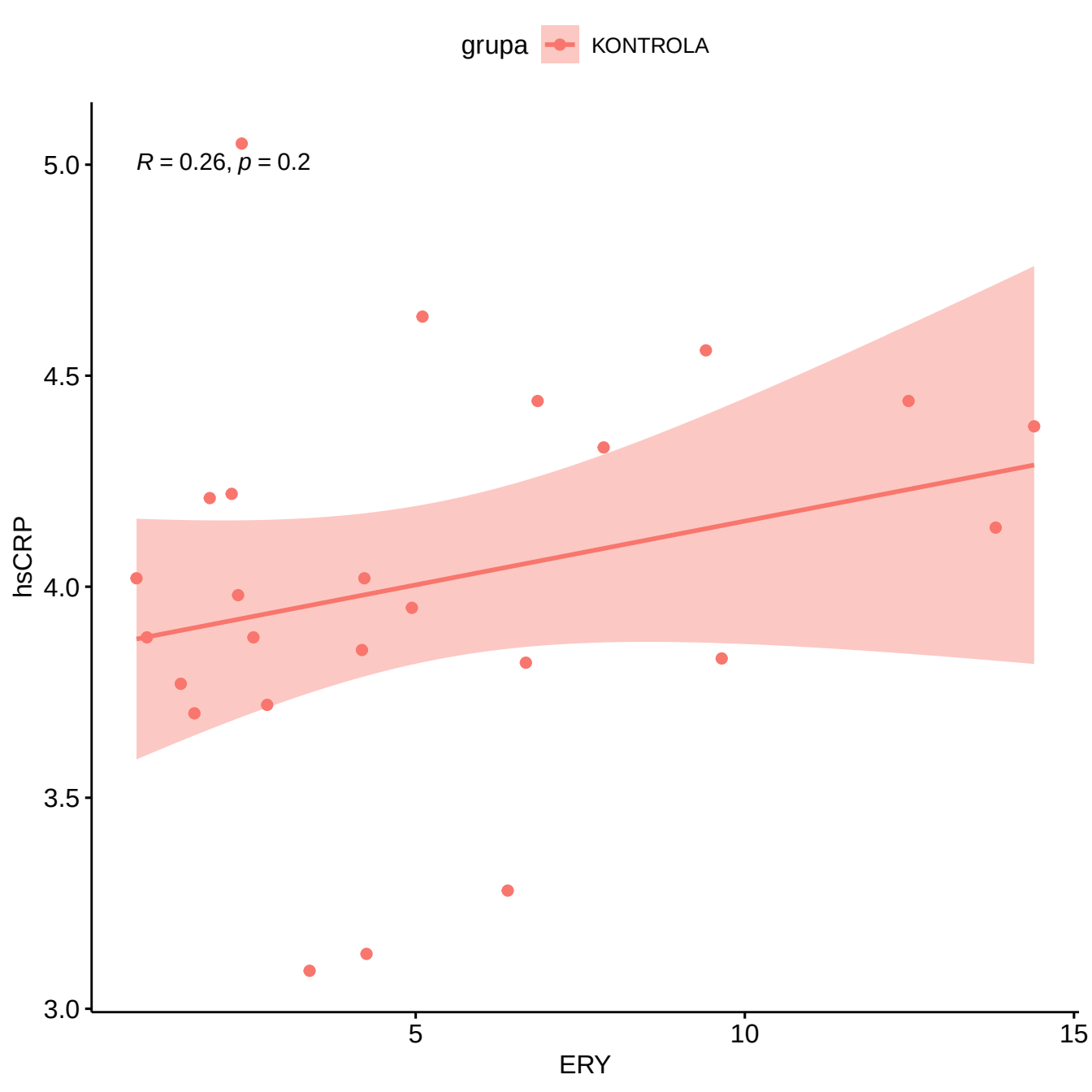

grupa
CHOR2
0.5
0.4
0.3
0.2
0.1
HGB
R = 0.21, p = 0.32
12
16
20
HCT

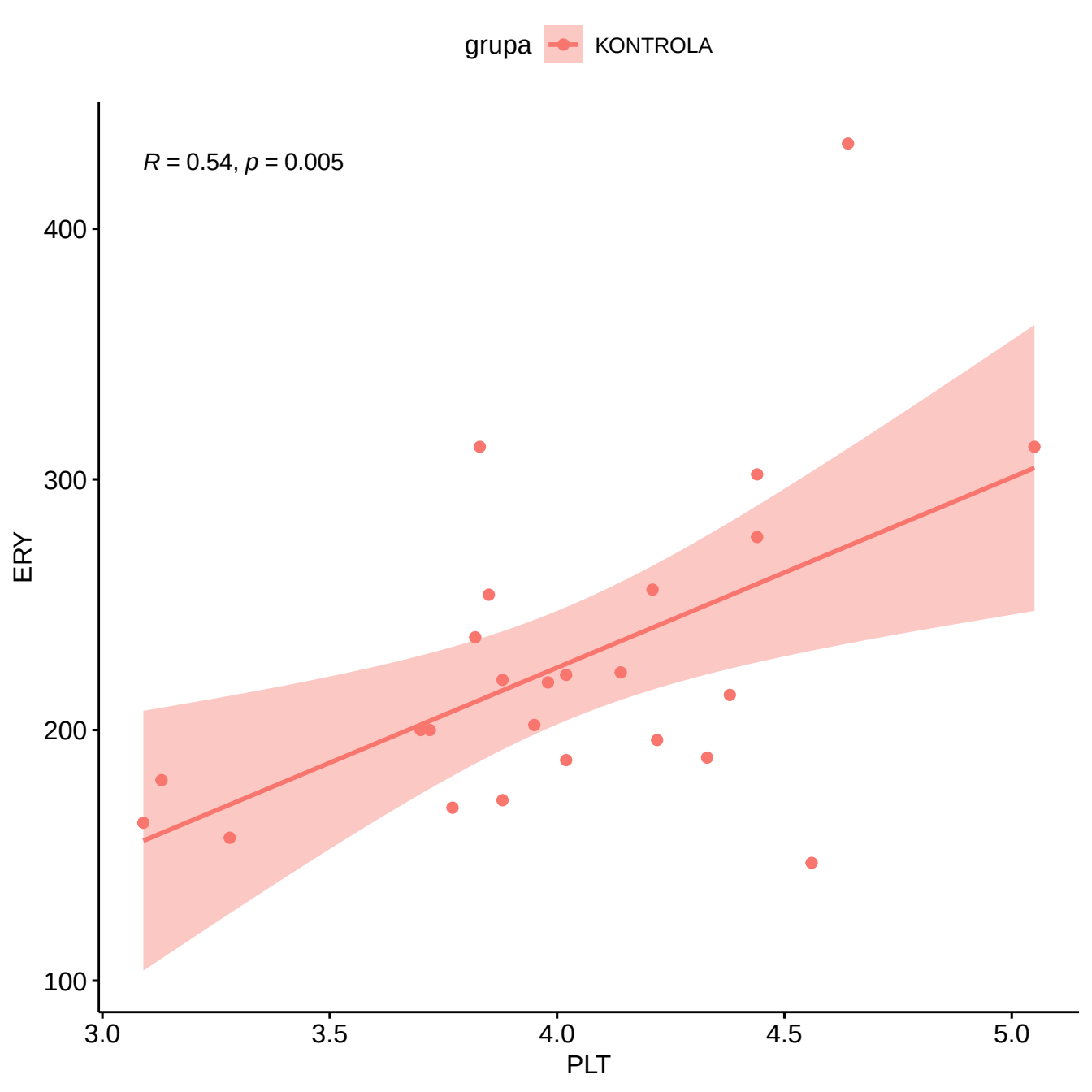

grupa
CHOR2
R = 0.05, p = 0.81
38
36
34
HCT
0.1
0.2
0.3
0.4
MCHC

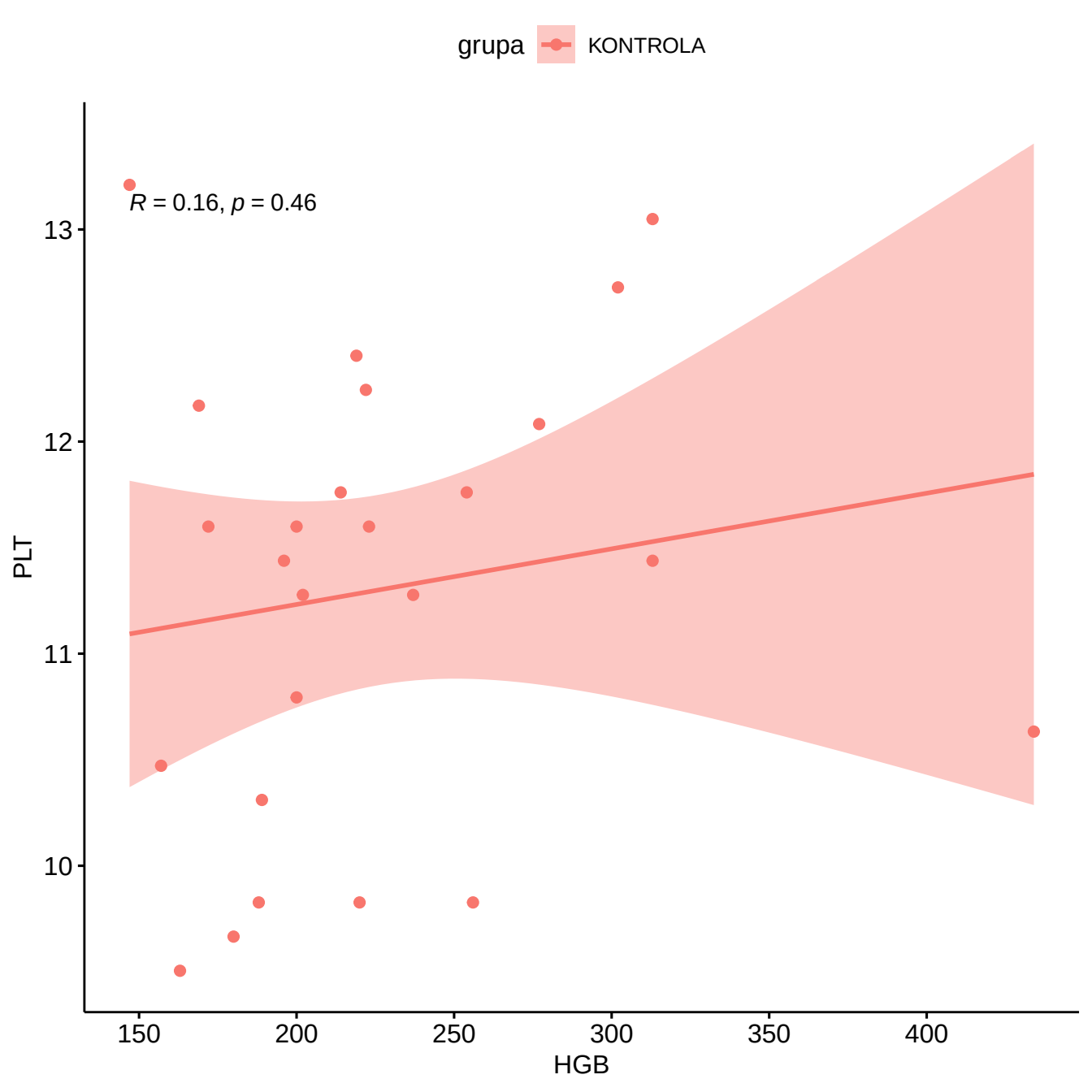

grupa
CHOR2
R = 0.3, p = 0.14
MCHC
5.0
2.5
0.0
34
36
38
MON

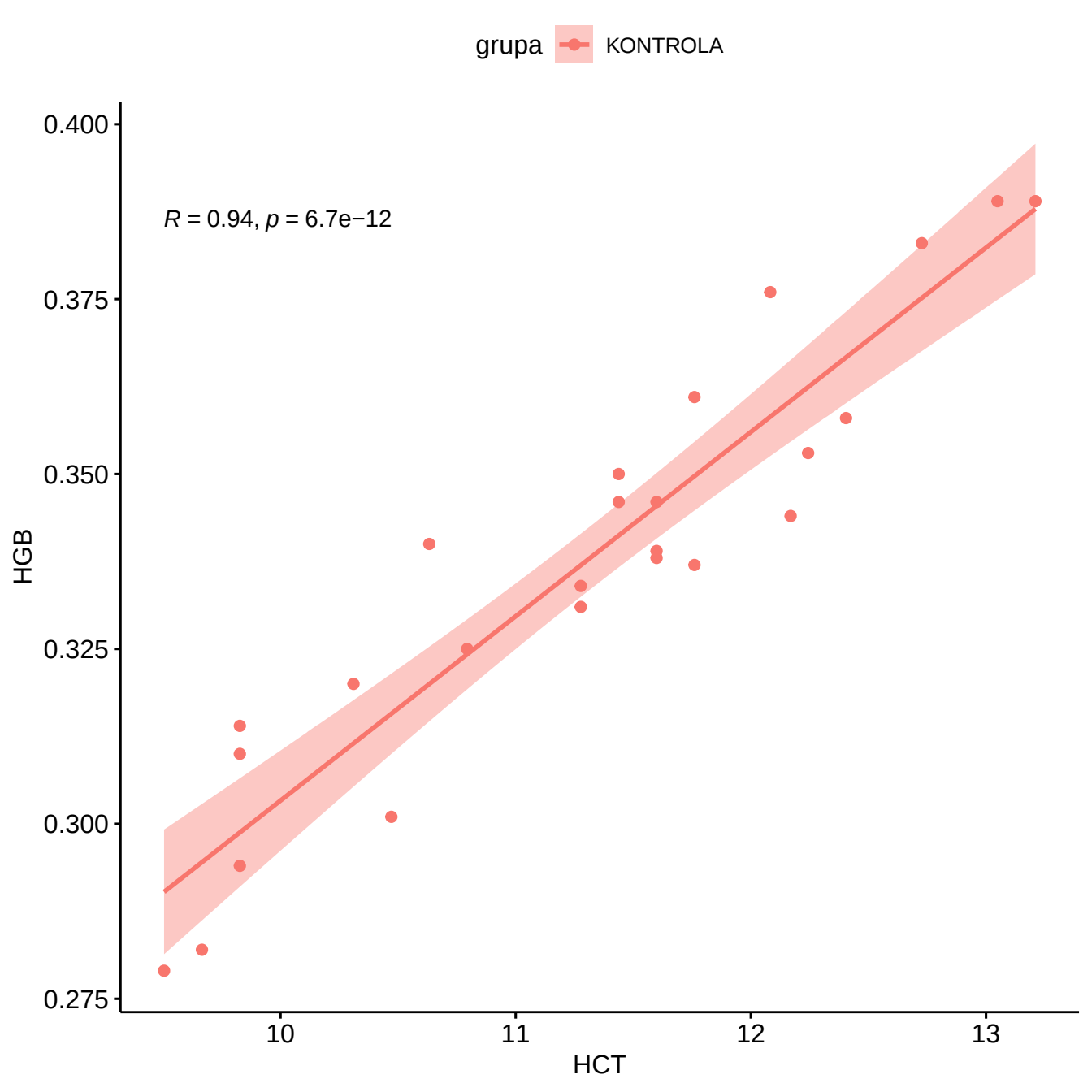

grupa
CHOR2
R = −0.33, p = 0.11
16
12
8
4
MON
0
2
4
6
LEU

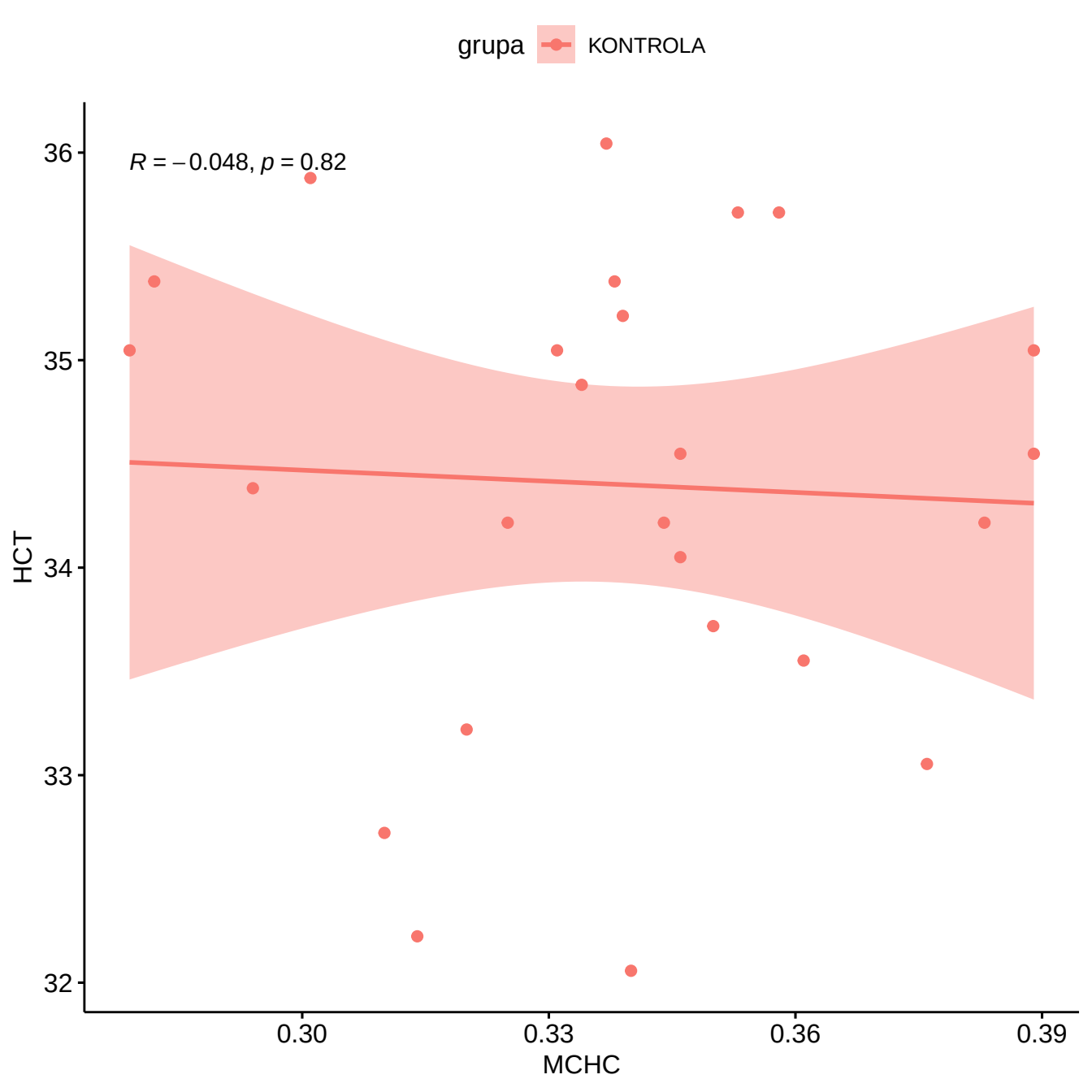

grupa
KONTROLA
15
10
5
0
wiek
R = −0.19, p = 0.36
30
40
hsCRP

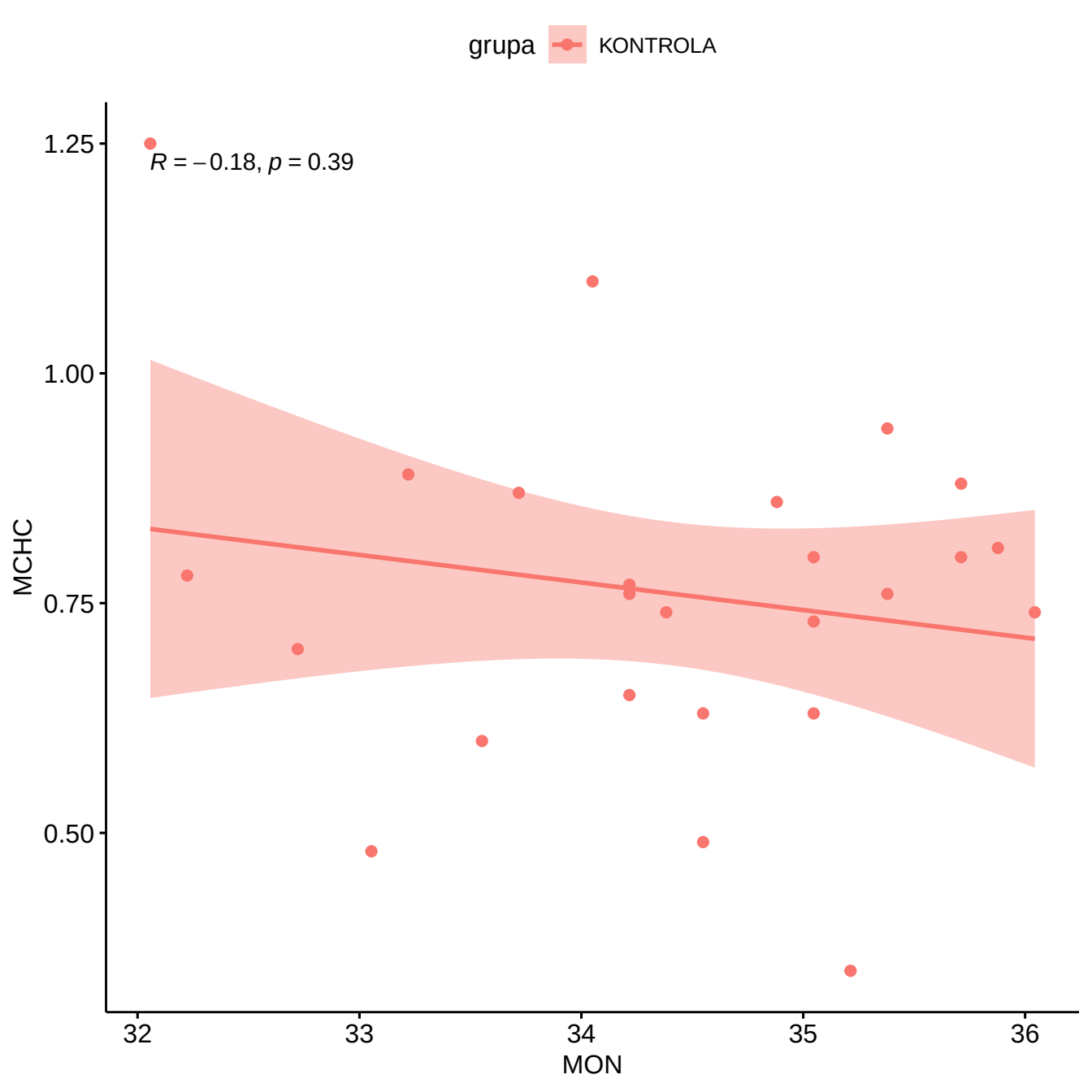

grupa
KONTROLA
R = 0.26, p = 0.2
5.0
4.5
4.0
3.5
3.0
hsCRP
5
10
15
ERY

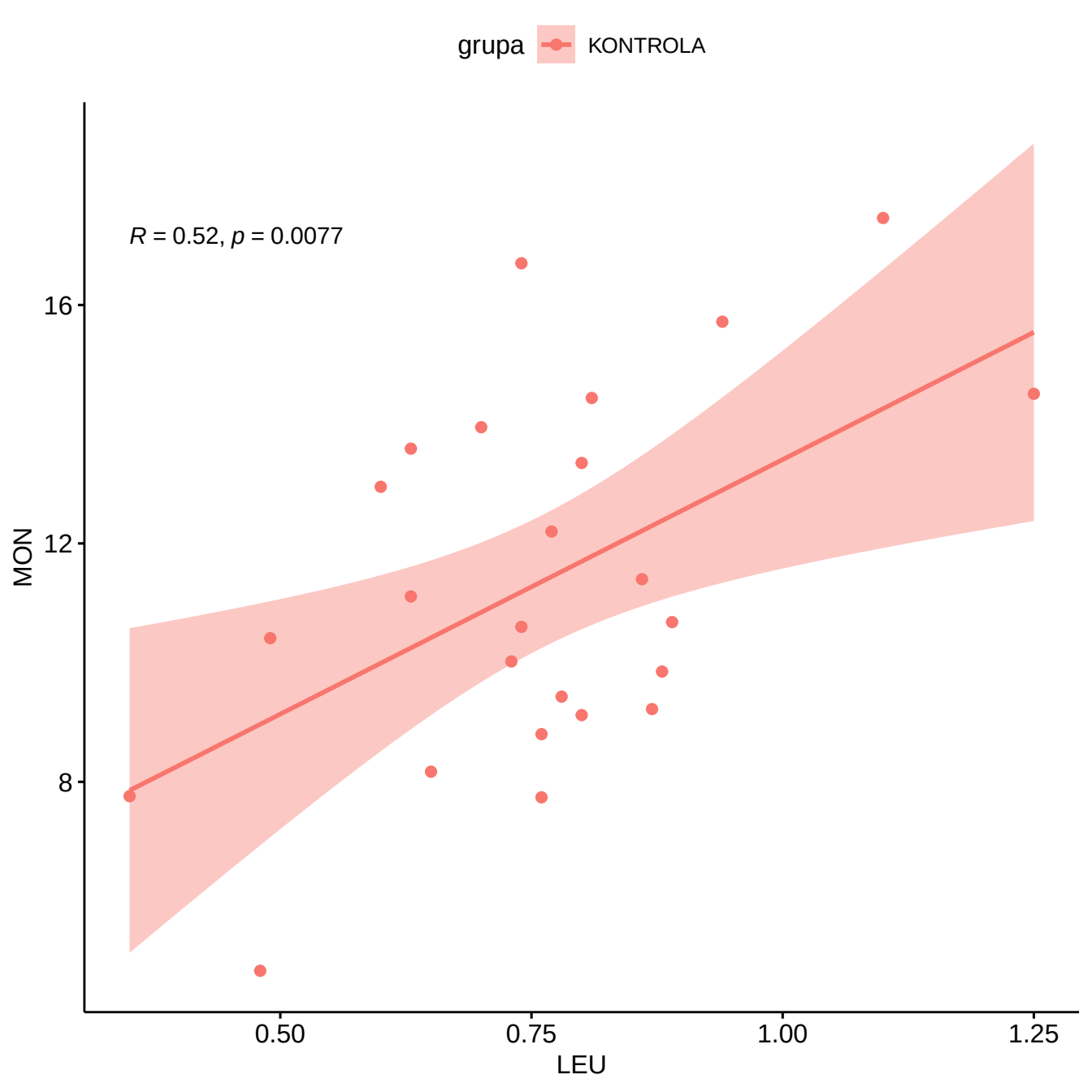

grupa
KONTROLA
R = 0.54, p = 0.005
400
300
200
100
ERY
3.0
3.5
4.0
4.5
5.0
PLT


grupa
KONTROLA
R = 0.16, p = 0.46
13
12
11
10
PLT
150
200
250
300
350
400
HGB


grupa
KONTROLA
0.400
0.375
0.350
0.325
0.300
0.275
HGB
R = 0.94, p = 6.7e−12
10
11
12
13
HCT


grupa
KONTROLA
R = −0.048, p = 0.82
HCT
36
35
34
33
32
0.30
0.33
0.36
0.39
MCHC


grupa
KONTROLA
R = −0.18, p = 0.39
1.25
1.00
0.75
0.50
MCHC
32
33
34
35
36
MON


grupa
KONTROLA
R = 0.52, p = 0.0077
16
12
8
MON
0.50
0.75
1.00
1.25
LEU