



**KAPITAŁ LUDZKI**  
NARODOWA STRATEGIA SPÓJNOŚCI



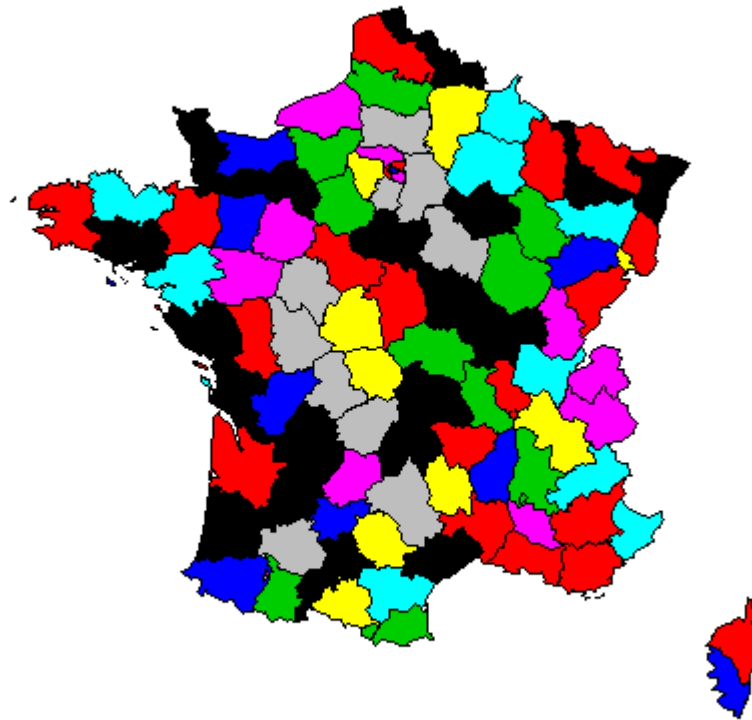
Uniwersytet  
Wrocławski

UNIA EUROPEJSKA  
EUROPEJSKI  
FUNDUSZ SPOŁECZNY

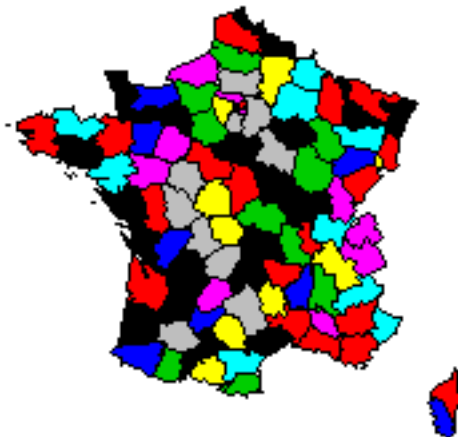


Projekt „*Nowa oferta edukacyjna Uniwersytetu Wrocławskiego odpowiedzią na współczesne potrzeby rynku pracy i gospodarki opartej na wiedzy*”

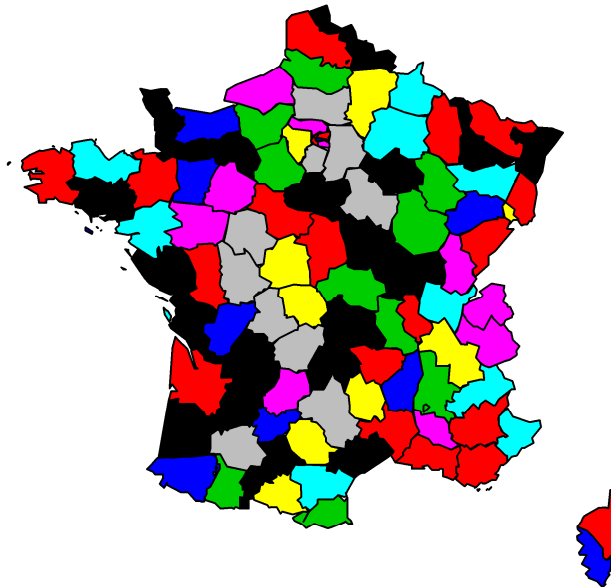
```
library("maps")  
png("francja.png")  
map("france", fill=T, col=1:10) #8 KB  
dev.off()
```



```
png("francjat.png",  
    height=4,  
    width=4,  
    units="in",res=600)  
map("france",fill=T,col=1:10) #51 kB  
dev.off()
```



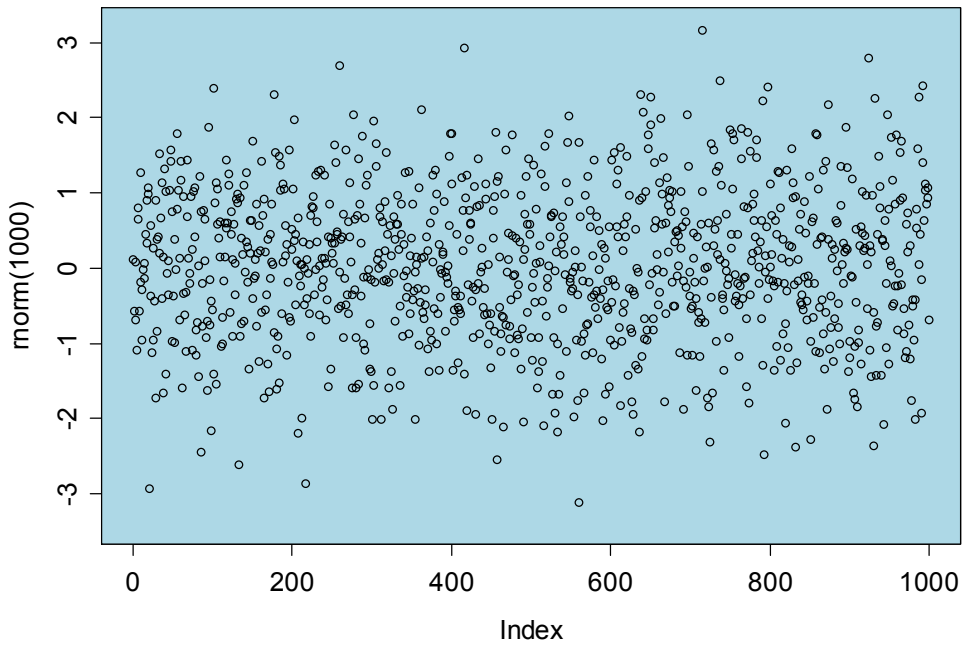
```
png("francjac.png",  
    height=12,  
    width=12,  
    units="cm",res=600)  
map("france",fill=T,col=1:10) #73 kB  
dev.off()
```



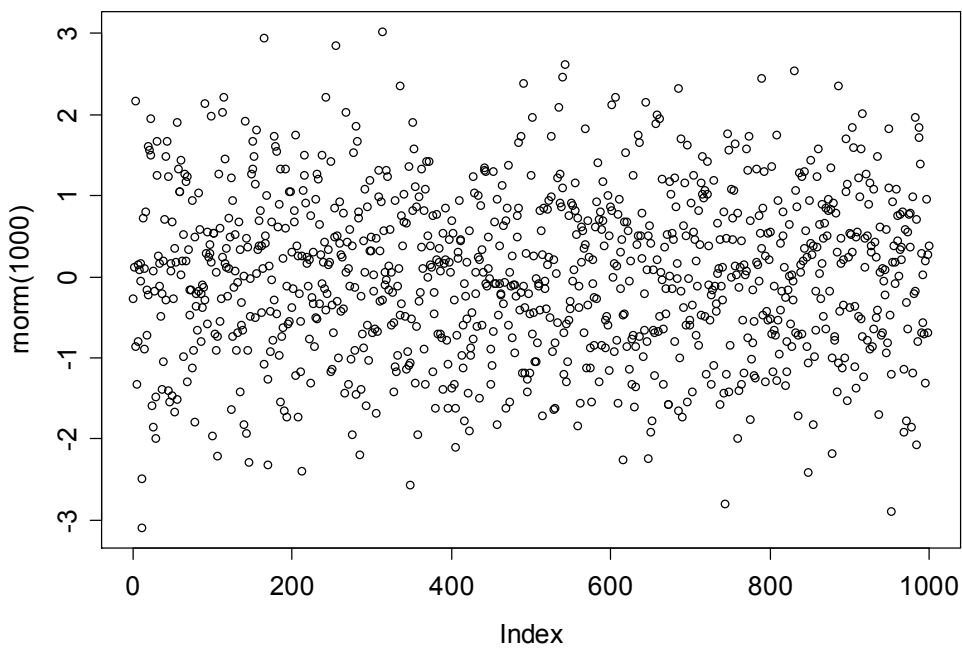
```
pdf("france.pdf",  
    height=12,  
    width=12  
)  
map("france",fill=T,col=1:10) #127kB  
dev.off()
```

```
pdf("france1.pdf") # 83 kB  
map("france",fill=T,col=1:10)  
dev.off()
```

```
plot(rnorm(1000),type="n")
x<-par("usr") # tylko przez par();daje współrzędne okna rysunku
rect(x[1],x[3],x[2],x[4],col="lightblue ")
points(rnorm(1000))
```

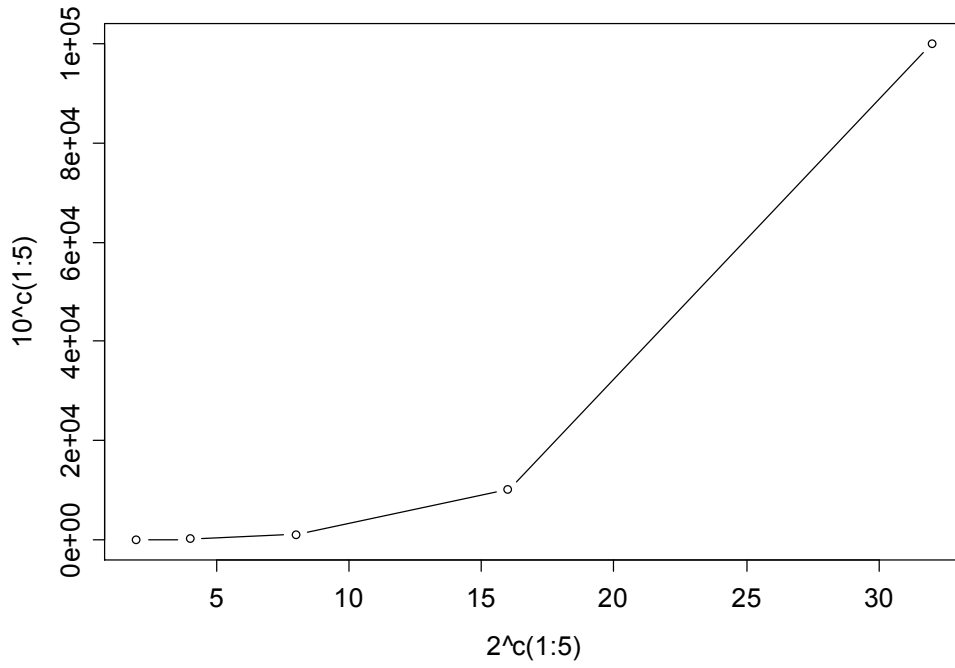


```
plot(rnorm(1000))
```

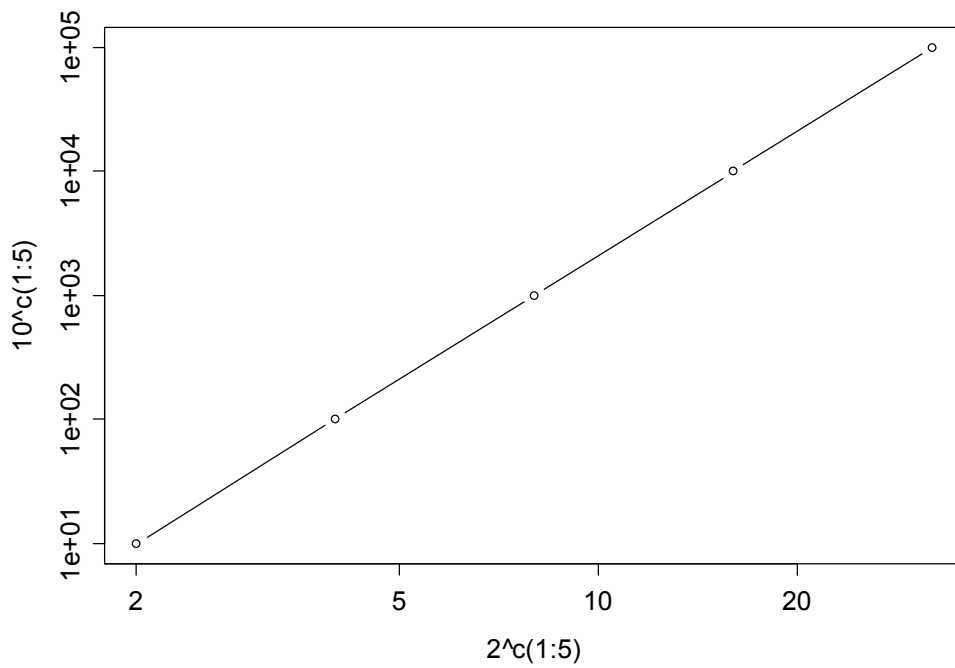


par() stosuje założone ustawienia do odwołania

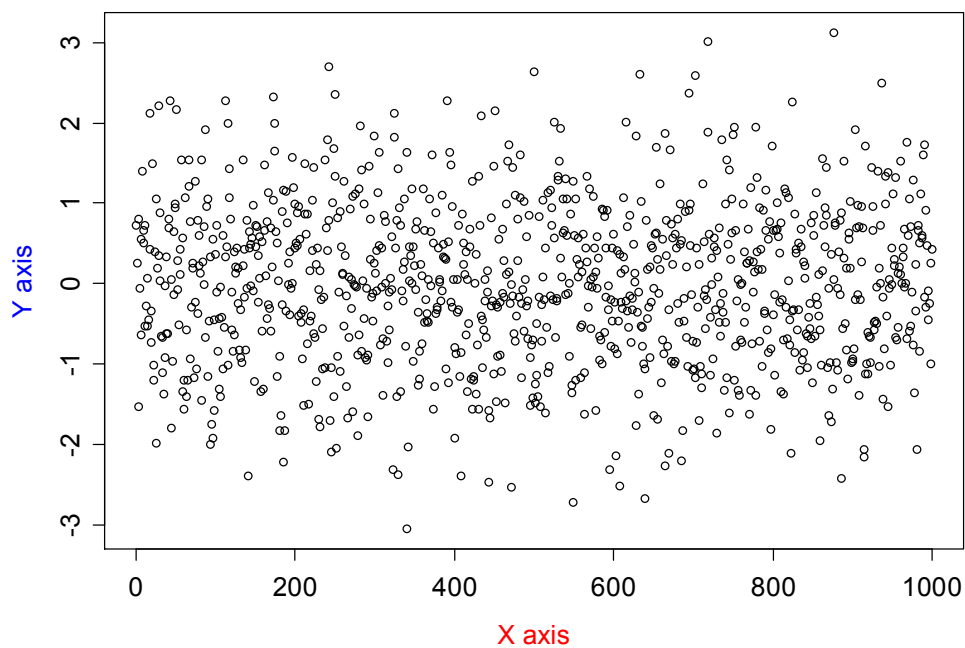
```
plot(2^c(1:5), 10^c(1:5), type="b")
```



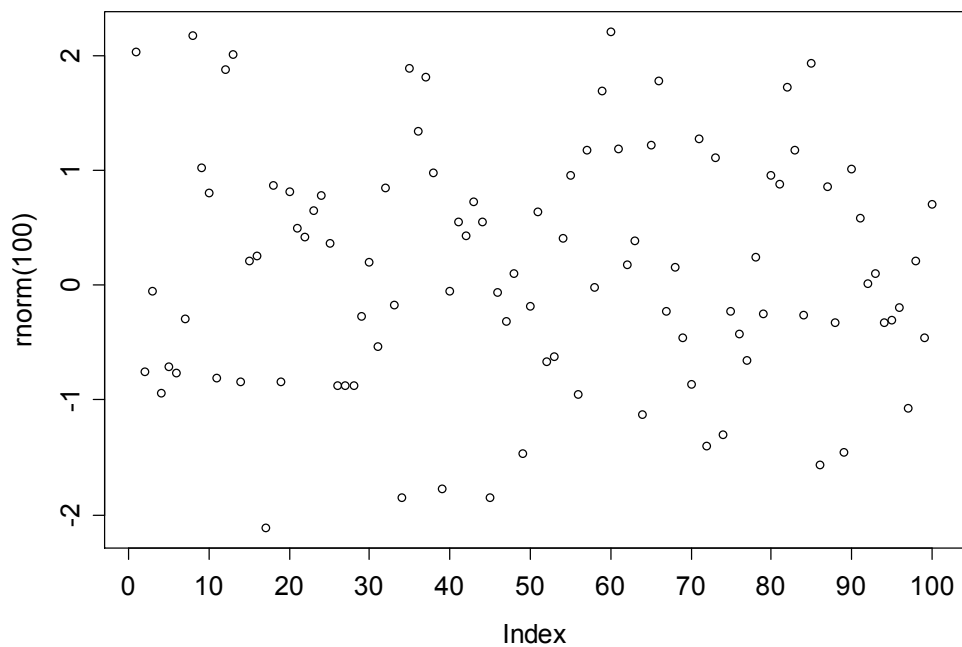
```
plot(2^c(1:5), 10^c(1:5), log="xy", type="b")
```



```
plot(rnorm(1000),xlab="",ylab="")
title(xlab="X axis",col.lab="red")
title(ylab="Y axis",col.lab="blue")
#jedyny sposób aby osie były różnych kolorów
```



```
# podział osi
plot(rnorm(100),xaxp=c(0,100,10))
```



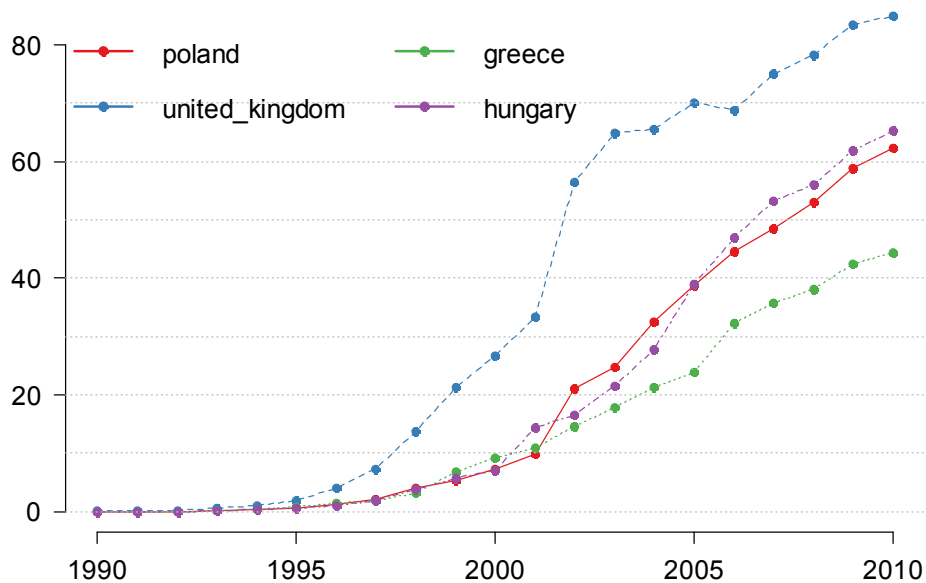
```

library("XML")
© Smarter Poland
kraj <- c("poland", "united_kingdom", "greece", "hungary")
# lista czterech tabel z danymi
dane <- lapply(kraj,
              function(k) readHTMLTable(paste("http://www.indexmundi.com/",
                                             k, "/internet-users.html", sep = ""), which = 3,
                                             colClasses = "numeric"))
# wyciągamy lata i procenty użytkowników
lata <- dane[[1]][, 1]
procenty <- as.data.frame(sapply(dane, `[`, 2))

library("RColorBrewer")
kolory <- brewer.pal(9, "set1")
# c("#E41A1C", "#377EB8", "#4DAF4A", "#984EA3", "#FF7F00", "#FFFF33",
#   "#A65628", "#F781BF", "#999999")
matplot(lata, procenty, type = "o", pch = 19, las = 1, xlab = "",
        ylab = "", col = kolory, main = "% osób z dostępem do Internetu",
        bty = "n")
abline(h = (0:7) * 10, col = "grey", lty = 3)
legend("topleft", kraj, ncol = 2, col = kolory, pch = 19, lwd = 2,
      bty = "n", cex = 1)

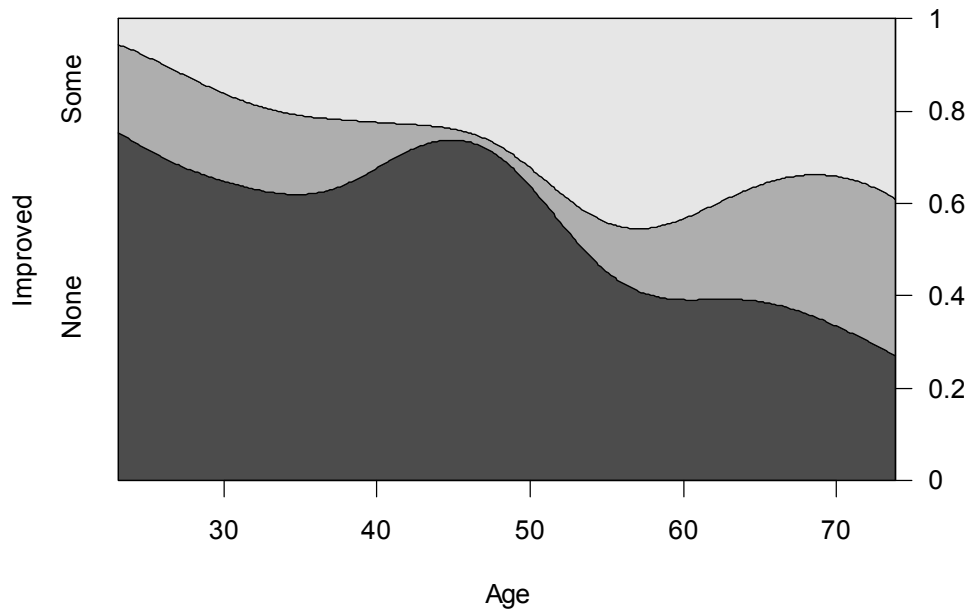
```

**% osób z dostępem do Internetu**



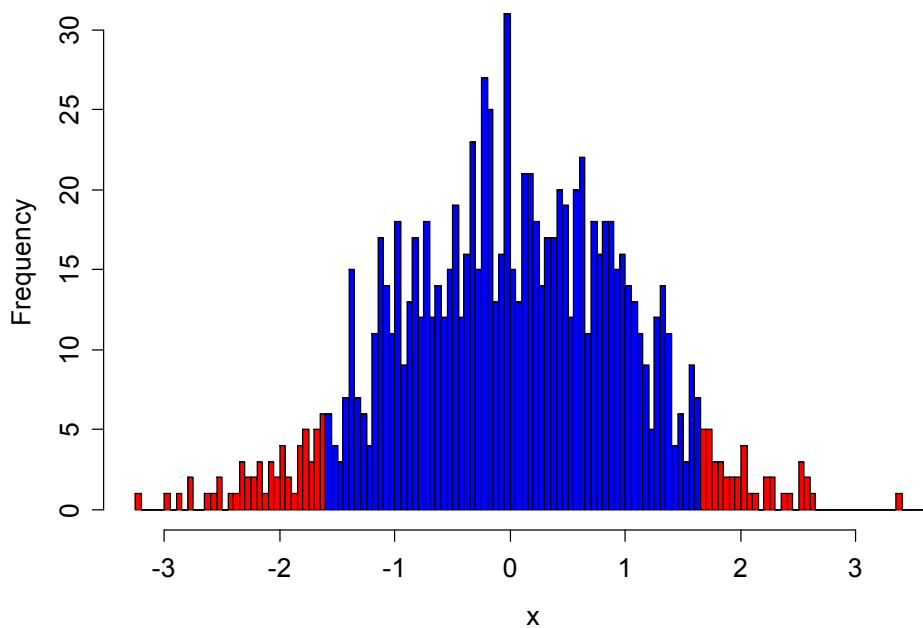
```
require(vcd)
data("Arthritis")
Arthritis[1:5,]
cd_plot(Improved ~ Age, data = Arthritis)
```

	ID	Treatment	Sex	Age	Improved
1	57	Treated	Male	27	Some
2	46	Treated	Male	29	None
3	77	Treated	Male	30	None
4	17	Treated	Male	32	Marked
5	36	Treated	Male	46	Marked



\*

**Histogram z ogonami na poziomie 0.1**

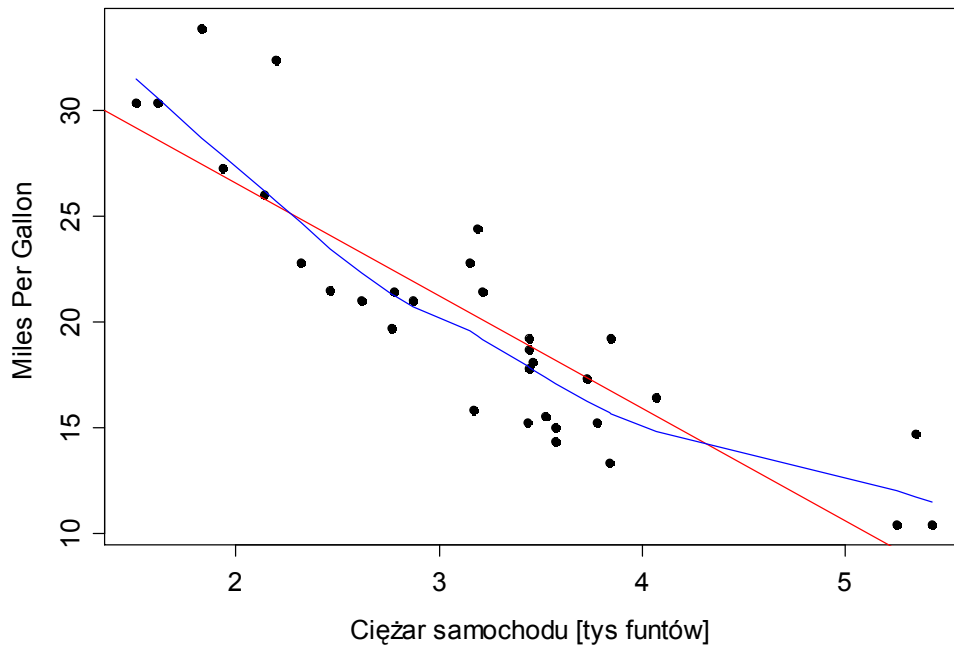


```
x <- rnorm(1000)
hx <- hist(x, breaks=100, plot=F)
plot(hx, col=ifelse(abs(hx$breaks) < 1.65, 4, 2),
     main="Histogram z ogonami na poziomie 0.1")
```



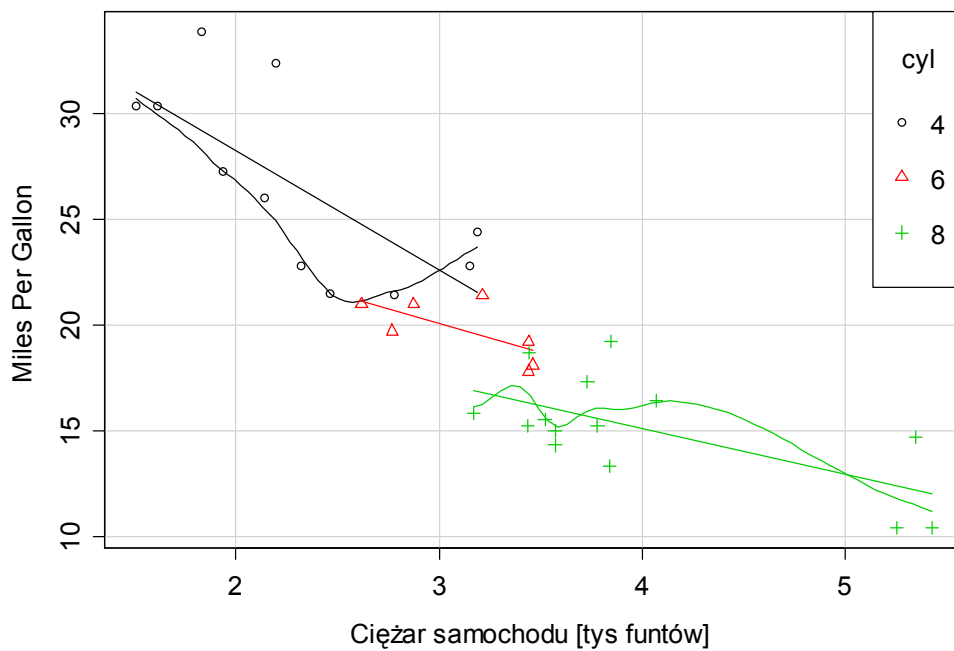
```
# plot z liniami dodatkowymi
```

Wykres rozrzutu



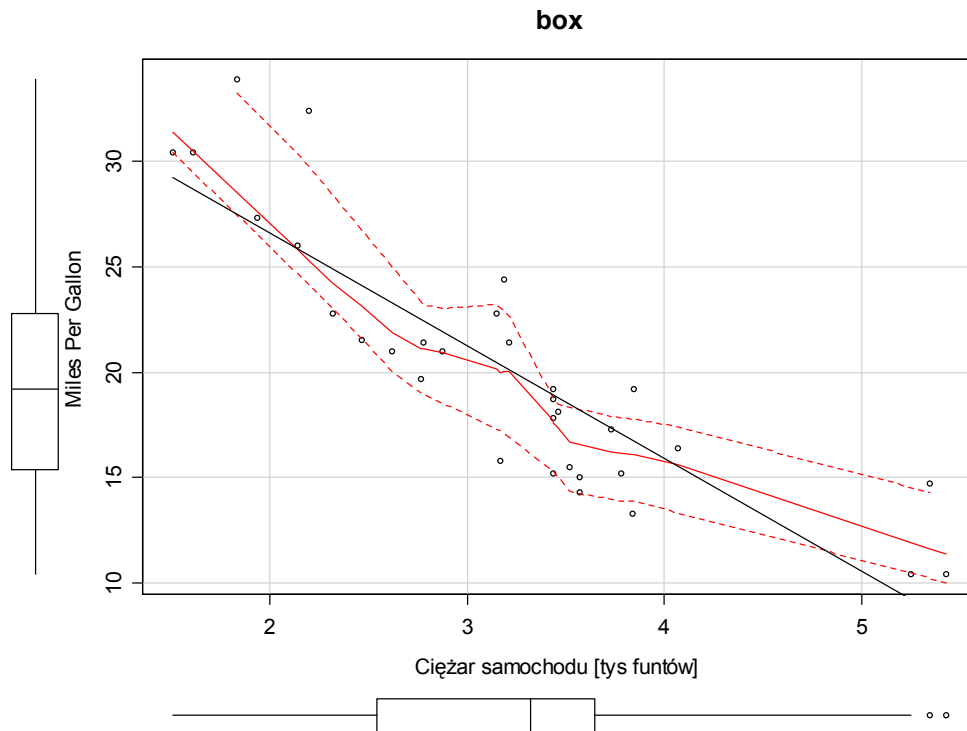
```
with(mtcars,{  
  plot(wt, mpg, main="wykres rozrzutu",  
        xlab="Ciężar samochodu [tys funtów] ", ylab="Miles Per Gallon ",  
        pch=19)  
  abline(lm(mpg~wt), col="red")  
  lines(lowess(wt,mpg), col="blue")  
})
```

Warunkowy

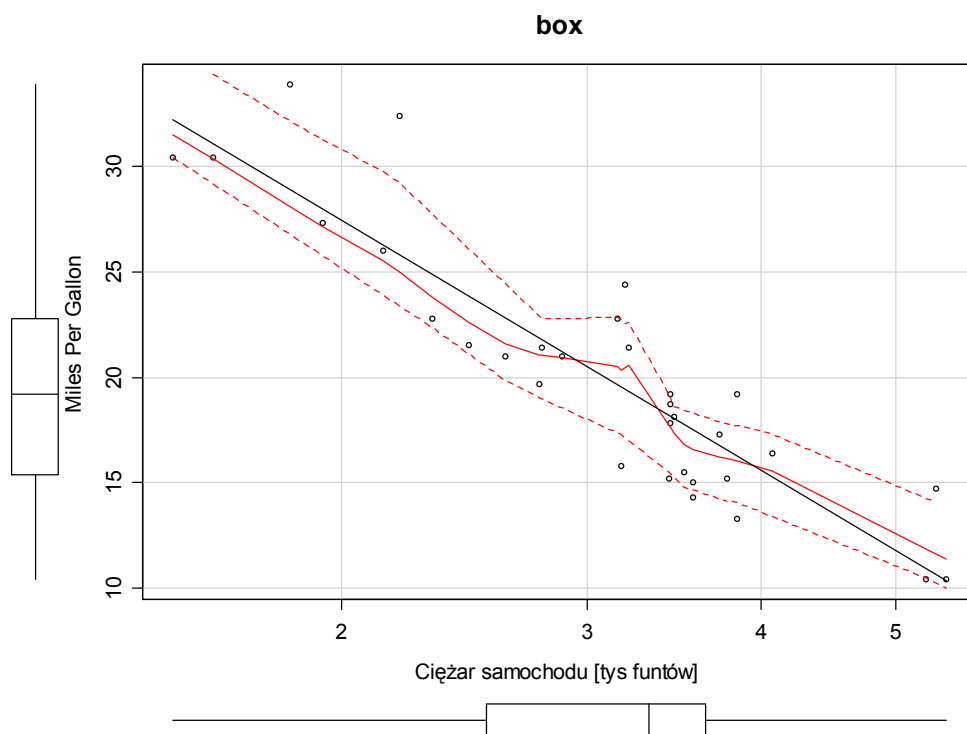


```
library(car)  
scatterplot(mpg ~ wt | cyl, data=mtcars,  
            xlab="Ciężar samochodu [tys funtów] ", ylab="Miles Per Gallon ",  
            main="warunkowy",
```

```
labels=row.names(mtcars), legend.coords="topright")
```

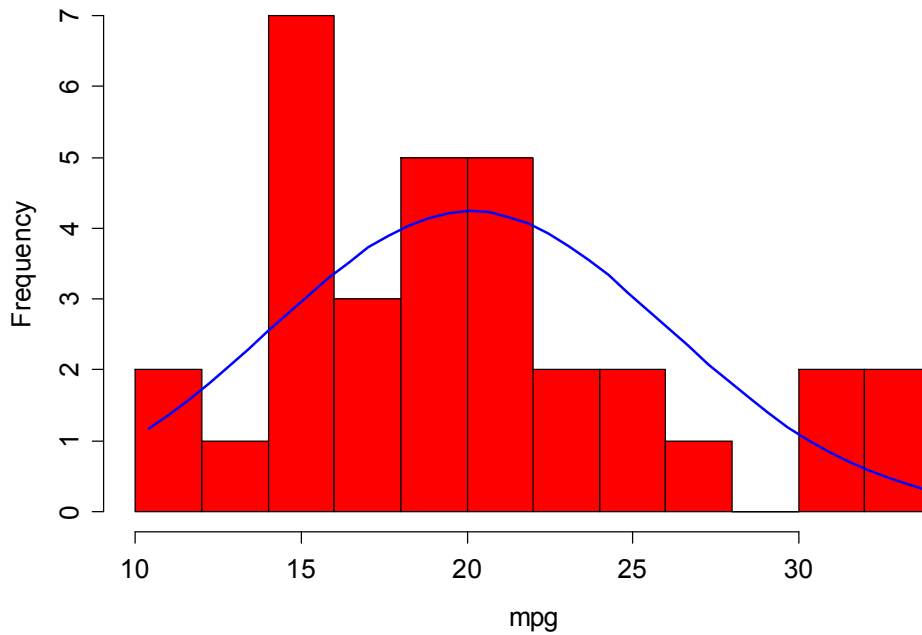


```
scatterplot(mpg ~ wt, data=mtcars,
            xlab="Ciężar samochodu [tys funtów] ", ylab="Miles Per Gallon ",
            main="box", boxplots="xy"
            )
```



```
scatterplot(mpg ~ wt, data=mtcars,
            xlab="Ciężar samochodu [tys funtów] ", ylab="Miles Per Gallon ",
            main="box", boxplots="xy", log="x"
            )
```

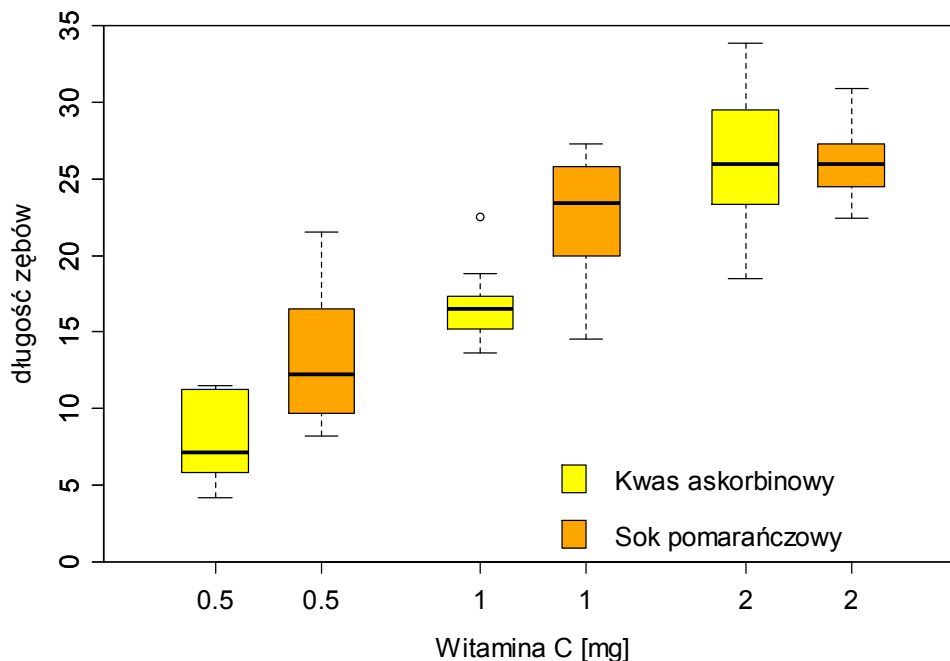
## Histogram z krzywą normalną



### # Nałożenie krzywej Gaussa

```
x <- mtcars$mpg
h<-hist(x, breaks=10, col="red", xlab="mpg",
        main="Histogram z krzywą normalną")
xfit<-seq(min(x),max(x),length=40)
yfit<-dnorm(xfit,mean=mean(x),sd=sd(x))
yfit <- yfit*diff(h$mids[1:2])*length(x)
lines(xfit, yfit, col="blue", lwd=2)
```

## Przyrost zębów świnek morskich



```
# wykres z odrębnymi kolorami dla grup
boxplot(len ~ dose, data = ToothGrowth,
        boxwex = 0.25, at = 1:3 - 0.2,
        subset = supp == "VC", col = "yellow",
        main = "Przyrost zębów świnek morskich",
        xlab = "witamina C [mg]",
        ylab = "długość zębów",
```

```

xlim = c(0.5, 3.5), ylim = c(0, 35), yaxs = "i")
boxplot(len ~ dose, data = ToothGrowth, add = TRUE,
        boxwex = 0.25, at = 1:3 + 0.2,
        subset = supp == "OJ", col = "orange")
legend(2, 9, c("kwas askorbinowy", "Sok pomarańczowy"),
      fill = c("yellow", "orange"), bty="n")

```

# w tytule kopia nazwy

```

baplot = function(x,y){
  xstd = (x - mean(x))/sd(x)
  ystd = (y - mean(y))/sd(y)

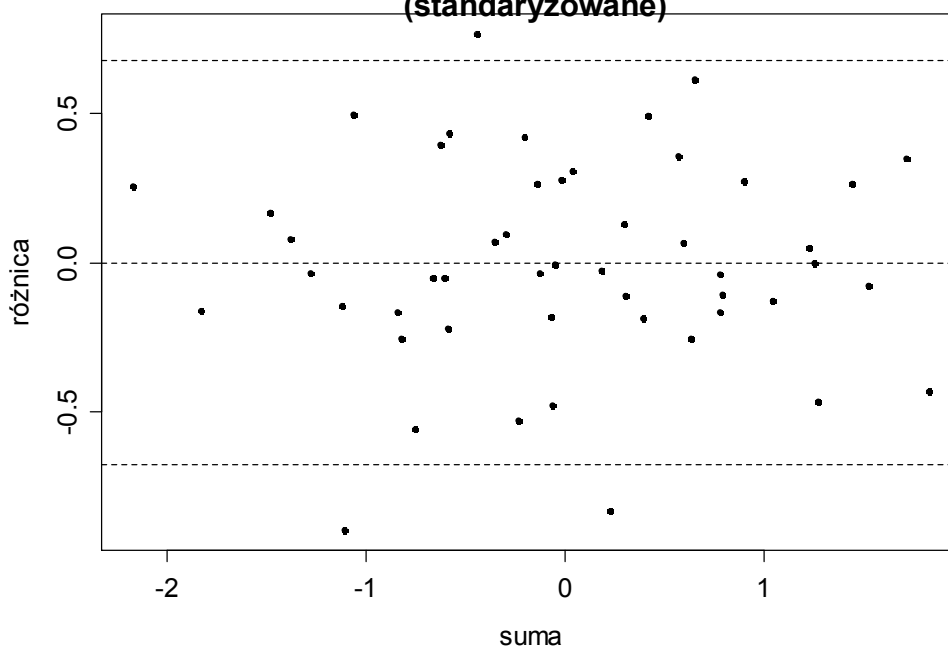
  bamean = (xstd+ystd)/2
  badiff = (ystd-xstd)/2

  plot(badiff~bamean, pch=20, xlab="suma", ylab="różnica")
  # deparse(substitute(varname)) cytuję nazwy danych
  title(main=paste("wykres Blanda-Altmana \n",
                  deparse(substitute(x)), "i", deparse(substitute(y)), "\n",
                  "(standaryzowane)"), adj=".5")
  #linie rferencyjne w locie
  abline(h = c(mean(badiff), mean(badiff)+1.96 * sd(badiff),
               mean(badiff)-1.96 * sd(badiff)), lty=2)
}

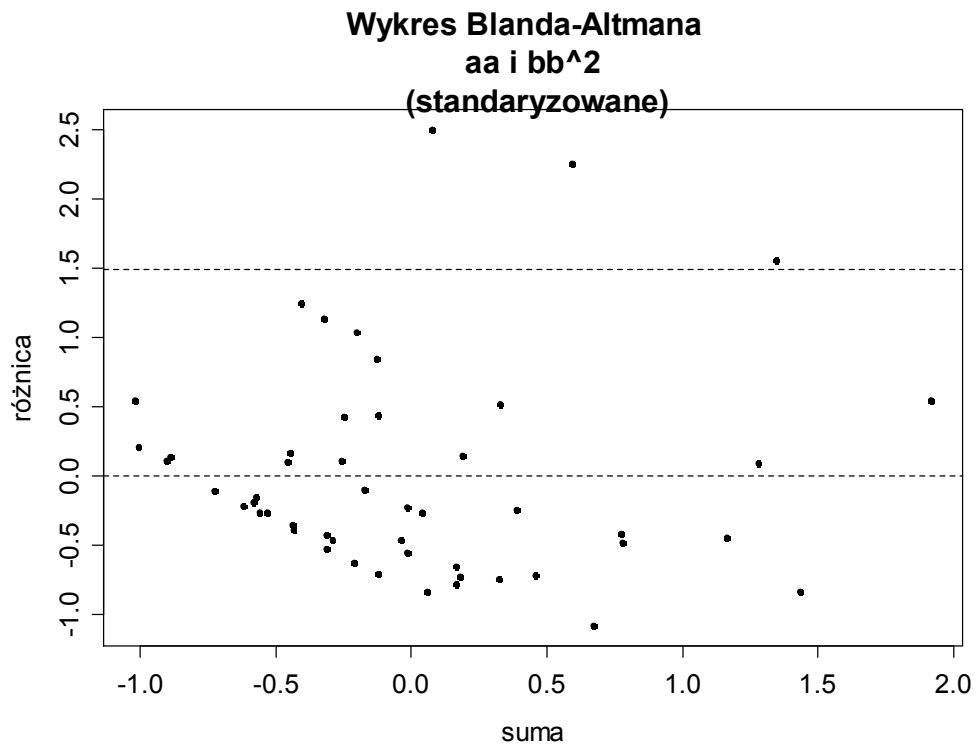
set.seed(78)
aa = rnorm(50)
bb = aa + rnorm(50)
baplot(aa,bb)

```

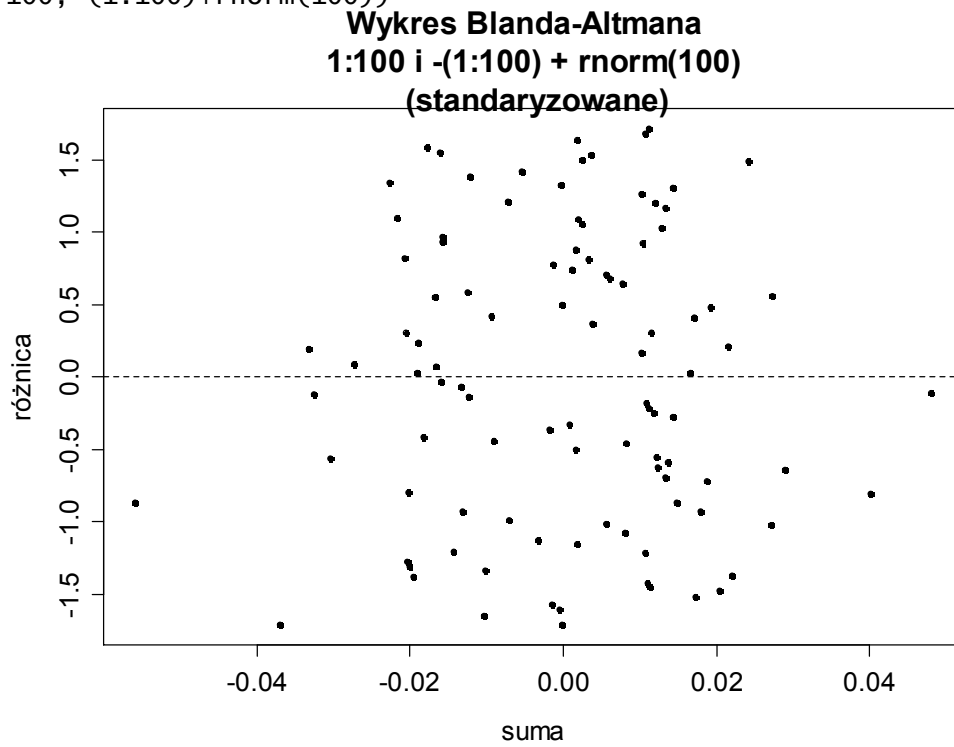
**Wykres Blanda-Altmana  
aa i bb  
(standaryzowane)**



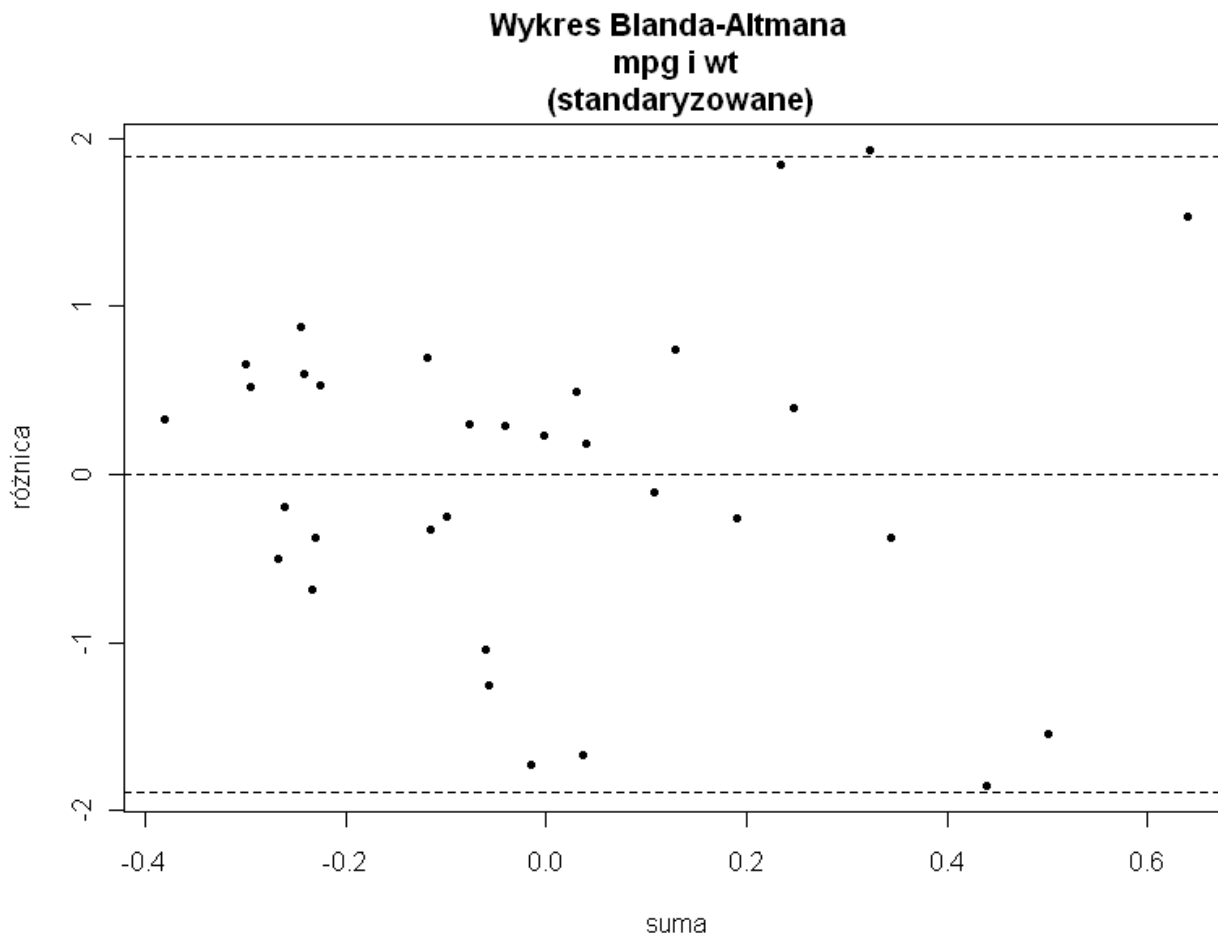
```
baplot(aa,bb^2)
```



```
baplot(1:100, -(1:100)+rnorm(100))
```

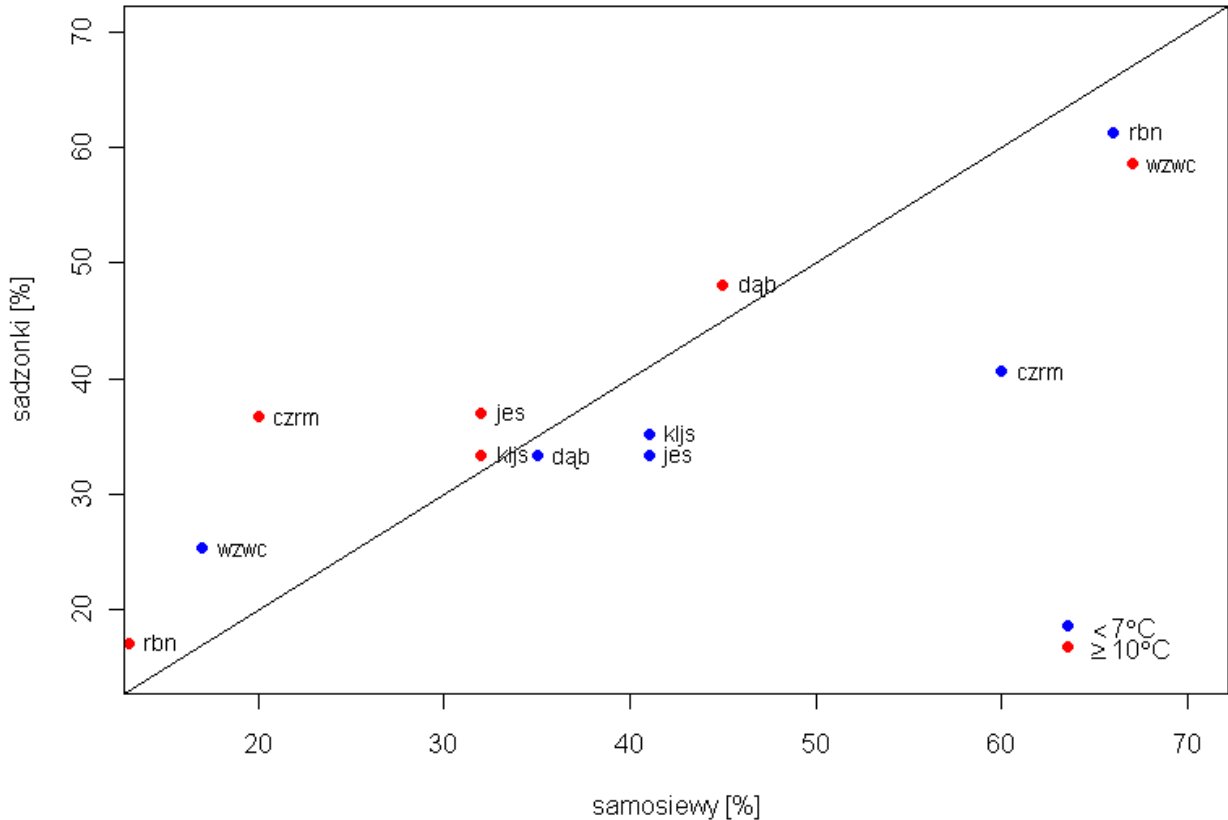


```
with(mtcars, baplot(mpg, wt))
```



\*

## Udział populacji samosiewów i sadzonek



# wykres dwukolorowy; legenda z symbolami matematycznymi

```

sx<-c(0.32, 0.2, 0.67, 0.32, 0.45, 0.13)*100
sy<-c(0.37, 0.367, 0.586, 0.334, 0.481, 0.171)*100

zx <- c(0.41, 0.6, ,0.17 ,0.41 ,0.35 ,0.66)*100
zy <-c(0.334, 0.406, 0.253, 0.352, 0.333, 0.613)*100

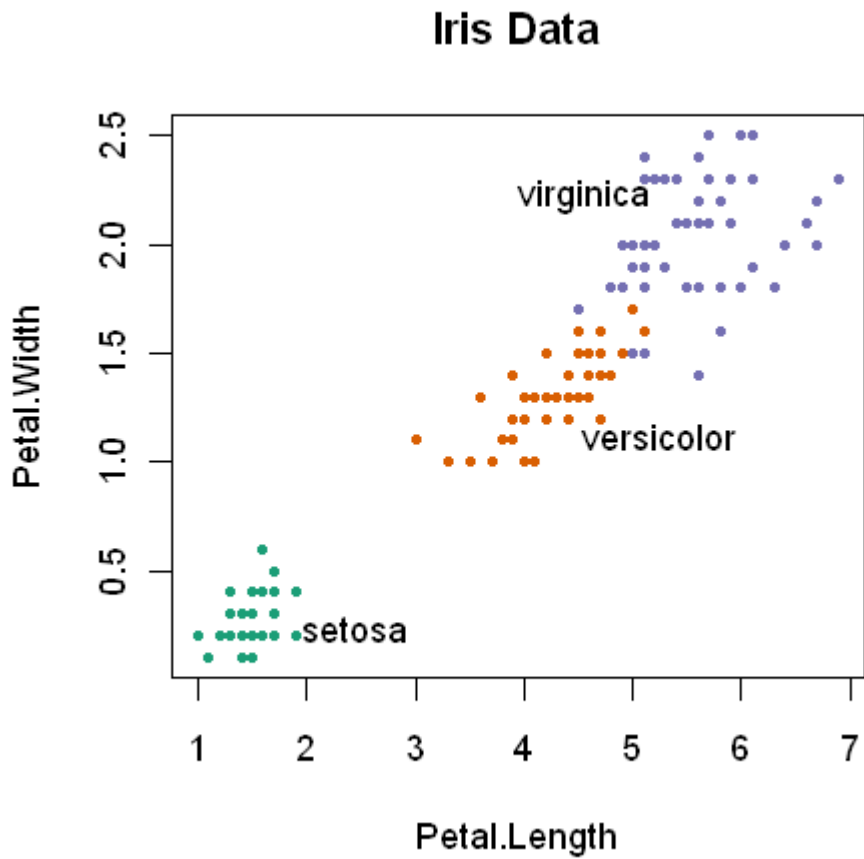
w1<-expression(paste(phantom(0)<7*degree,"C"))
w2<-expression(paste(phantom(0) >= 10*degree,"C"))
plot(sx,sy, col="red",pch=19,ylim=c(15,70),xlim=c(15,70),
      main="udział populacji samosiewów i sadzonek",
      xlab="samosiewy [%]",ylab="sadzonki [%]")
text(sx,sy,c("jes","czrm","wzwc","kljs","dąb","rbn"),cex=0.9,pos=4)
abline(0,1)
points(zx,zy,col="blue",pch=19)
text(zx,zy,c("jes","czrm","wzwc","kljs","dąb","rbn"),cex=0.9,pos=4)
legend("bottomright",legend=c(w1,w2),pch=c(19,19),
      col=c("blue","red"),cex=1,bty="n",y.intersp=0.3,x.intersp=0.1)
  
```

```

# wykres z dużymi punktami
png("czytelne.png", pointsize=18)

plot(iris[3:4], col=iriscolors[unclass(iris$Species)], #zamiast nazw numerki
      main="Iris Data", pch=20)
# etykiety obok średnich pozycje ustalane ręcznie
# (adjusted manually so labels do not overlap points)
text(irismeans[,2]+c(1,1,-1), irismeans[,3]+c(0,-.2,.2),
      irismeans[,1])
dev.off()

```





```
# lepsze na slajdy
png("czytelne 1.png", width=4.5, height=5, units="in",
    pointsize=14, res=200)
plot(iris[3:4], bg=iriscolors[unclass(iris$Species)], main="Iris Data",
     pch=c(21,22,24)[unclass(iris$Species)], las=1)
text(irismeans[,2]+c(1.5,1.5,-1.5), irismeans[,3]+c(0,-.3,.3), irismeans[,1])
dev.off()
```

## Iris Data

