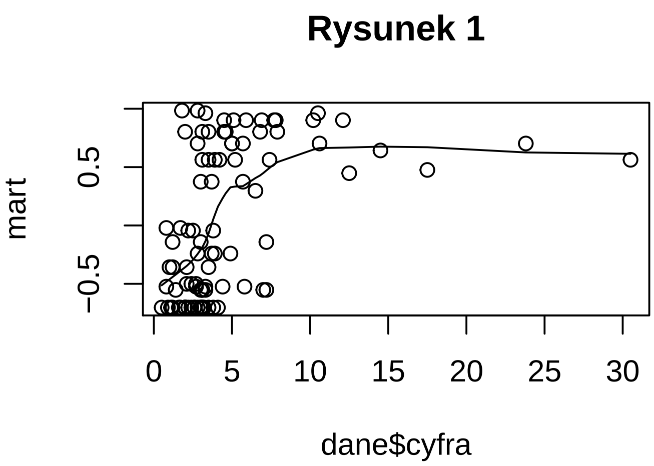
BIOSTATYSTYKA – PRACA DOMOWA 2

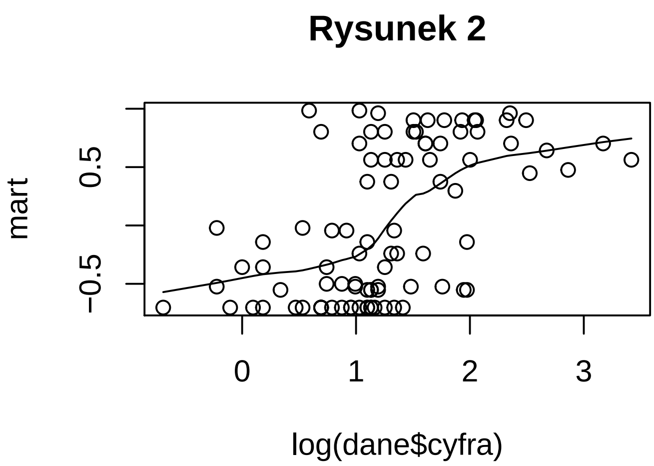
MARTA SOMMER – BSMAD

Dla danych *newcyfra.dta* analizujemy wpływ markera CYFRA na czas przeżycia bezobjawowego. Będziemy robić to dwiema metodami - nieparametryczną (model proporcjonalnych hazardow) i parametryczną (model Weibulla i lognormalny). Na końcu przeanalizujemy, który z tych modeli okazał się najlepszy.

Model PH

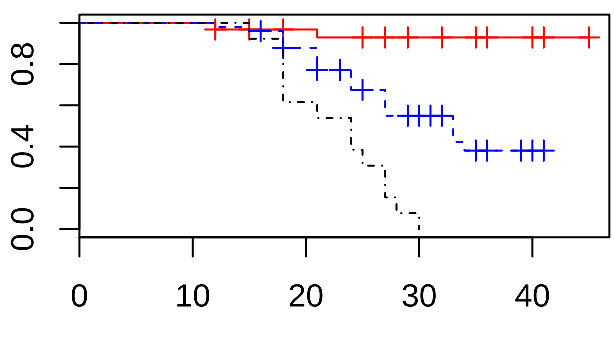
cyfra jest zmienną ciągłą. Trudno jest więc sprawdzić na wykresie, czy ma ona wpływ na czas przeżycia. Dlatego spróbujemy – w sposób lekko sztuczny – zrobić z niej zmienną dyskretną. W tym celu budujemy pusty model PH i rysujemy jego reszty martyngałowe (Rysunek 1). Z Rysunku 1 widać, że dobrym przekształceniem zmiennej *cyfra* będzie przekształcenie logarytmiczne. Narysujmy wykres jeszcze raz, uwzględniając to przekształcenie (Rysunek 2).





Na podstawie Rysunku 2 można zauważyć, że log(*cyfra*) układa się w trzy pasy, których granicą jest 1 i 2, co w języku *cyfry* przekłada się na liczby $\exp(1) = 2,72$ oraz $\exp(2) = 7,39$. Stworzę więc nową dyskretną zmienną objaśniającą *cyfra.new*, która przyjmuje wartości $1, 2$ i 3, gdy zmienna ciągła *cyfra* wpada do przedziałów $(-\infty; 2,72], (2,72; 7,39], (7,39; \infty]$, odpowiednio. W ten sposób choć w przybliżeniu będziemy mogli określić wpływ tej zmiennej na funkcję przeżycia.

Narysujmy krzywe przeżycia wyznaczone metodą Kaplana-Meiera dla trzech poziomów zmiennej *cyfra.new*. Z rysunku widać, że różnią się one istotnie. Zmienna *cyfra* ma więc wpływ na czas przeżycia.



Zbudujmy teraz model proporcjonalnych hazardów dla wszystkich zmiennych objaśniających oprócz *adeno, large, plano, tnm1, tnm2* i *tnm3*, gdyż byłyby one współliniowe ze zmiennymi *histpat* i *newtnm*.

```
## Call:
## coxph(formula = Surv(dftime, dfree) ~ log(cyfra) + wiek + plec +
##     ps + histpat + newtnm, data = dane)
##
```

```
##
##                 coef exp(coef) se(coef)      z      p
## log(cyfra)  1.03060     2.803   0.3180  3.241 0.0012
## wiek        0.00708     1.007   0.0324  0.218 0.8300
## plec        0.10455     1.110   0.5805  0.180 0.8600
## ps          0.69307     2.000   0.3827  1.811 0.0700
## histpat    -0.66222     0.516   0.2208 -2.999 0.0027
## newtnm      0.05986     1.062   0.0282  2.123 0.0340
##
## Likelihood ratio test=46.1  on 6 df, p=2.82e-08  n= 94, number of events= 39
```

Z testu Walda wynika, że zmienne *wiek, plec* i *ps* nie są istotne. Zmienna *ps* jest na granicy istotności tego testu. Rysując dla niej krzywe Kaplana-Meiera widać jednak, że zmienna jest istotna. Brak jej wpływu w naszym modelu może być spowodowany współliniowością ze zmienną *cyfra*. Zbudujmy więc model bez niej i bez *wiek* i *plec*.

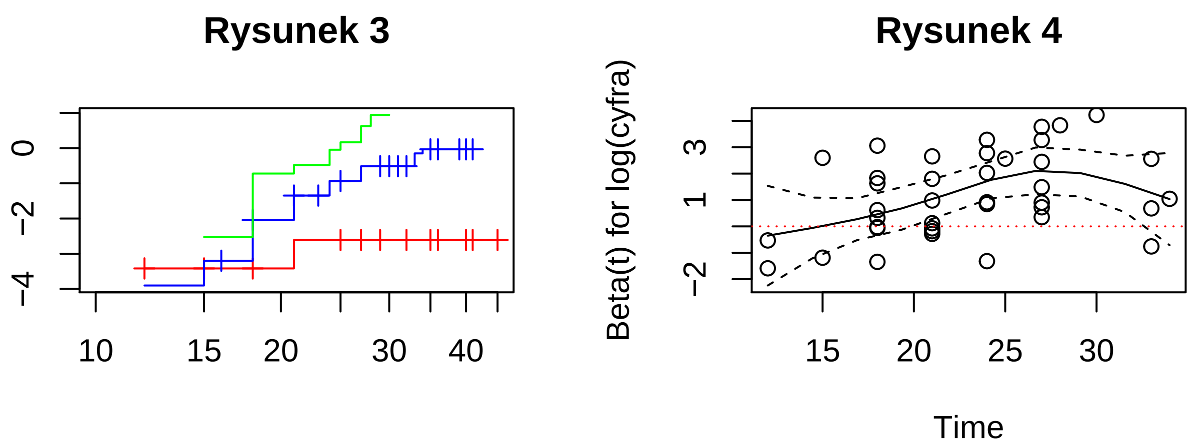
```
## Call:
## coxph(formula = Surv(dftime, dfree) ~ log(cyfra) + histpat +
##     newtnm, data = dane)
##
##
##                coef exp(coef) se(coef)     z       p
## log(cyfra)  1.1785      3.25    0.248  4.76 1.9e-06
## histpat    -0.6539      0.52    0.209 -3.12 1.8e-03
## newtnm      0.0599      1.06    0.028  2.14 3.2e-02
##
## Likelihood ratio test=42.6  on 3 df, p=2.94e-09  n= 94, number of events= 39
```

Teraz już wszystkie trzy zmienne są istotne. Interpretacja wyników jest taka, że gdy zmienna logarytm zmiennej *cyfra* zwiększy się o jeden, to ryzyko śmierci zwiększy się u nas $3,25$ raza. Jest to więc wpływ znaczący.

Przejdźmy teraz do sprawdzenia, czy spełnione jest założenie proporcjonalnych hazardów, czy w ogóle z tego modelu mogliśmy skorzystać. Z wykresu $\log(-\log)$ dla zmiennej *cyfra.new* (Rysunek 3) widać, że założenie PH niekoniecznie jest spełnione. Sprawdźmy jeszcze założenia testem opartym na skalowanych resztach Schoenfelda:

```
##                rho  chisq       p
## log(cyfra)  0.372  5.456 0.01950
## histpat     0.108  0.418 0.51806
## newtnm     -0.298  5.055 0.02456
## GLOBAL         NA 11.355 0.00996
```

Widać, że niestety zmienna *cyfra* nie spełnia założeń, bo mamy podstawy do odrzucenia hipotezy testu. Podobnie w przypadku zmiennej *newtnm*. Potwierdza to wykres skalowanych reszt Schoenfelda dla zmiennej *cyfra*. Niestety model PH nie jest więc całkowicie adekwatny. Niemniej jednak daje jakieś wyobrażenie o czasie przeżycia. Spróbujmy więc dopasować model parametryczny.



ATF

Model Weibulla

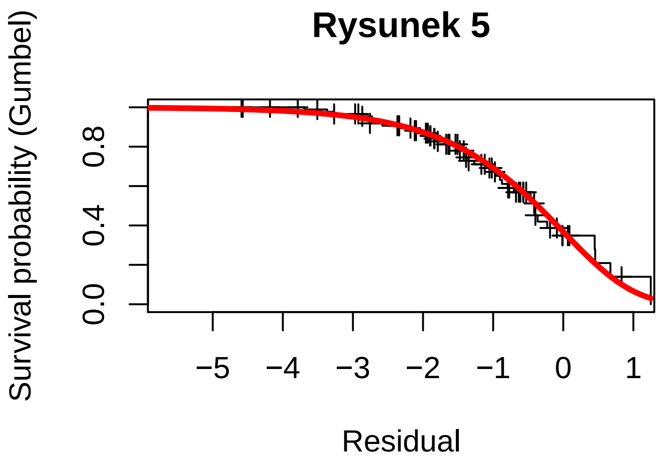
Dopasujmy model Weibulla do wszystkich zmiennych. Wartości współczynników na skali czasu wyglądają następująco:

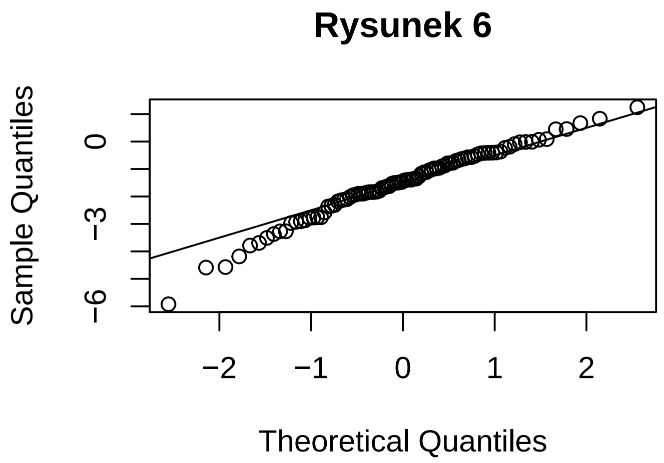
```
## (Intercept)        wiek        plec          ps       cyfra       adeno
##     18.4275      1.0110      1.0932      0.7332      0.9906      0.7699
##       large       plano        tnm1        tnm2        tnm3
##      0.7835      1.0000      1.5599      1.3941      1.0000
```

Zmienne, które są bliskie 1 będą nieistotne, tak więc dopasujemy nowy model tylko ze zmiennymi *ps,cyfra,adeno,large,tnm1* i *tnm2*. Teraz wartości współczynników na skali czasu wyglądają następująco:

```
## (Intercept)          ps       cyfra       adeno       large        tnm1
##     41.6049      0.7331      0.9847      0.7698      0.7863      1.5216
##        tnm2
##      1.3460
```

Wszystkie (niestety oprocz *cyfra*) są istotne. Sprawdźmy, czy założenia modelu są spełnione. W tym celu porównajmy krzywą przeżycia dla standaryzowanych reszt z modelu z krzywą przeżycia odpowiadającą rozkładowi Gumbela (Rysunek 5) oraz wykres kwantylowy dla reszt (Rysunek 6). Na wykresach widać, że w przybliżeniu założenia są spełnione.





Spróbujmy jeszcze zrobić przekształcenie logarytmiczne na zmiennej *cyfra*. Otrzymujemy wtedy takie wartości współczynników:

```
## (Intercept)          ps       cyfra       adeno       large        tnm1
##     63.8214      0.8354      0.7341      0.6781      0.6875      1.2832
##        tnm2
##      1.2738
```

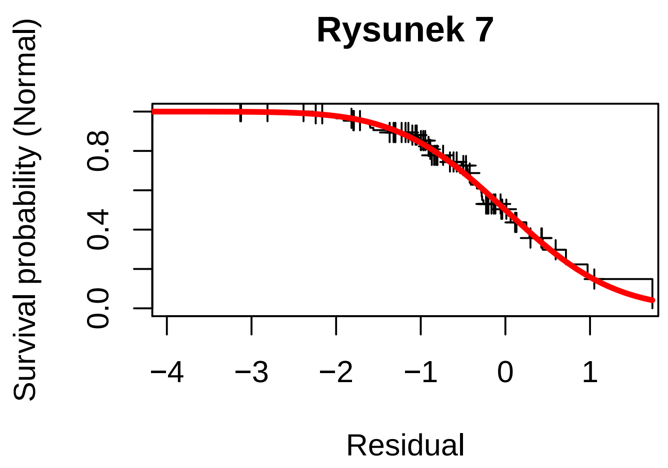
Cyfra jest już istotna, a założenia również są spełnione.

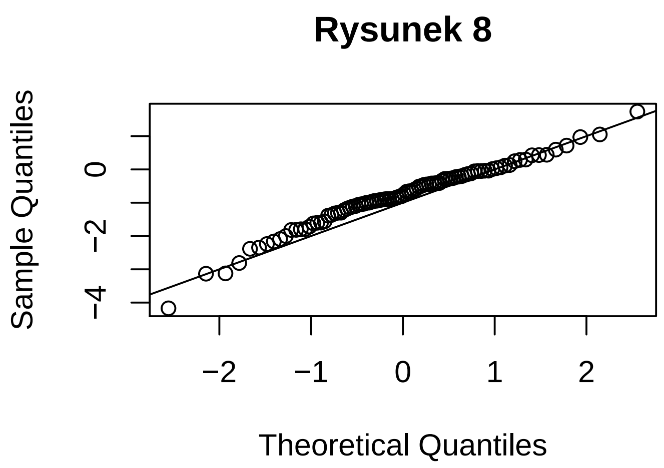
Model log-normalny

Dopasujmy model log-normalny dla tych samych zmiennych, co w modelu Weibulla. Wartości współczynników na skali czasu wyglądają następująco:

```
## (Intercept)          ps       cyfra       adeno       large        tnm1
##     36.4353      0.8285      0.9755      0.7408      0.7816      1.5757
##        tnm2
##      1.3978
```

Współczynniki są bardzo zbliżone do tych uzyskanych z modelu Weibulla. Sprawdźmy założenia modelu log-normlnego (Rysunki 7 i 8). Widać, że założenia są spełnione.





Gdy zrobimy jeszcze przekształcenie logarytmiczne na zmiennej *cyfra*, otrzymamy:

```
## (Intercept)          ps       cyfra       adeno       large        tnm1
##     53.1896      0.8611      0.7255      0.6962      0.7429      1.4286
##        tnm2
##      1.3453
```

Wnioski

W modelu proporcjonalnych hazardów zmienna *cyfra* była bardzo istotna – zwiększała ryzyko zdarzenia ponad trzy razy. Model ten nie był jednak do końca adekwatny, gdyż nie spełniał założeń (był zależny od czasu).

Dwa rozpatrzone przeze mnie modele parametryczne - model Weibulla i log-normalny założenia spełniają już bardzo dobrze. Wskazują jednak na brak istotności interesującej nas zmiennej *cyfra*. Gdy jednak zrobimy na tej zmiennej przekształcenie logarytmiczne i ponownie dopasujemy model, okaże się, że ta zmienna jest już istotna, a założenia modelu nadal są spełnione. Ostatecznie uważam więc, że należałoby zastosować jeden z modeli parametrycznych.

Syntaks R-a

```
library("foreign")
library("survival")
library("rms")
dane <- read.dta("C:\\Users\\Marta\\Desktop\\Marta\\studia\\rok4\\Biostatystyka\\2\\newcyfra.dta")[,
    -1]
m0 <- coxph(Surv(dftime, dfree) ~ 1, data = dane)
mart <- resid(m0)
plot(dane$cyfra, mart, main = "Rysunek 1")
lines(lowess(dane$cyfra, mart, iter = 0, f = 0.6))
plot(log(dane$cyfra), mart, main = "Rysunek 2")
lines(lowess(log(dane$cyfra), mart, iter = 0, f = 0.6))
cyfra.new <- numeric(length(dane$cyfra))
for (i in 1:length(dane$cyfra)) {
    if (dane$cyfra[i] <= exp(1))
        cyfra.new[i] <- 0 else if (dane$cyfra[i] <= exp(2))
        cyfra.new[i] <- 1 else cyfra.new[i] <- 2
}
dane <- cbind(dane, cyfra.new)
km <- survfit(Surv(dftime, dfree) ~ cyfra.new, data = dane)
plot(km, col = c("red", "blue", "black"), lty = c(1, 2, 4))
m2 <- coxph(Surv(dftime, dfree) ~ log(cyfra) + wiek + plec + ps + histpat +
```

```r
    newtnm, data = dane)
m3 <- coxph(Surv(dftime, dfree) ~ log(cyfra) + histpat + newtnm, data = dane)
m3s <- cox.zph(m3, transform = "identity")
plot(km, col = c("red", "blue", "green"), fun = function(x) log(-log(x)), main = "Rysunek 3",
    log = "x", firstx = 10)
plot(m3s, df = 4, nsmo = 10, se = TRUE, var = 1, main = "Rysunek 4")
abline(0, 0, lty = 3, col = "red")
w1 <- survreg(Surv(dftime, dfree) ~ wiek + plec + ps + cyfra + adeno + large +
    plano + tnm1 + tnm2 + tnm3, data = dane, dist = "weibull")
p <- 1/w1$scale
time.w1 <- exp(w1$coefficients)
w2 <- psm(Surv(dftime, dfree) ~ ps + cyfra + adeno + large + tnm1 + tnm2, data = dane,
    dist = "weibull", y = TRUE)
time.w2 <- exp(w2$coefficients)
res.w2 <- resid(w2, type = "cens")
plot(survfit(res.w2 ~ 1), conf = "none", ylab = "Survival probability (Gumbel)",
    xlab = "Residual", main = "Rysunek 5")
lines(res.w2, col = "red")
qqnorm(res.w2[, 1], main = "Rysunek 6")
abline(-1.5, 1)
w3 <- psm(Surv(dftime, dfree) ~ ps + log(cyfra) + adeno + large + tnm1 + tnm2,
    data = dane, dist = "weibull", y = TRUE)
time.w3 <- exp(w3$coefficients)
time.w3
l1 <- psm(Surv(dftime, dfree) ~ ps + cyfra + adeno + large + tnm1 + tnm2, data = dane,
    dist = "lognormal")
time.l1 <- exp(l1$coefficients)
res.l1 <- resid(l1, type = "cens")
plot(survfit(res.l1 ~ 1), conf = "none", ylab = "Survival probability (Normal)",
    xlab = "Residual", main = "Rysunek 7")
lines(res.l1, col = "red")
qqnorm(res.l1[, 1], main = "Rysunek 8")
abline(-1, 1)
l3 <- psm(Surv(dftime, dfree) ~ ps + log(cyfra) + adeno + large + tnm1 + tnm2,
    data = dane, dist = "lognormal", y = TRUE)
time.l3 <- exp(l3$coefficients)
time.l3
```