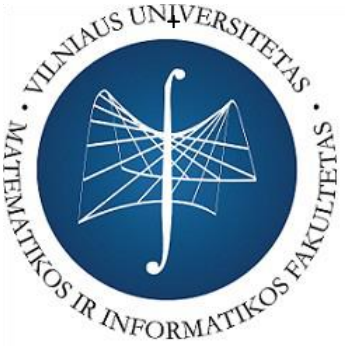


VILNIAUS UNIVERSITETAS

MATEMATIKOS IR INFORMATIKOS FAKULTETAS

DUOMENŲ MOKSLAS. BAKALAURAS

Tiriamasis projektas

Regresinė analizė su R

Ataskaita

Užduotį atliko: Ugnius Vilimas

Rytis Baltaduonis

Justinas Pipiras

Recenzentai:

Aivaras Varkalis

Anton Cifirov

Maksim Čižov

Vilnius 2024

Turinys

Įvadas

Krūties vėžys yra viena iš labiausiai paplitusių ir mirtiniausių onkologinių ligų moterų tarpe visame pasaulyje. Per pastaruosius 30 metų medicinos srityje įvyko didžiulis progresas, tačiau krūties vėžys išlieka rimta sveikatos problema. Šios ligos tyrimai ir gydymas nuolat tobulėja, todėl svarbu gilinti supratimą apie šios ligos mirtingumą, išgyvenimą ir išgyvenamumą.

Dabartinė medicina siekia ne tik ilginti žmonių gyvenimo trukmę, bet ir gerinti gyvenimo kokybę sergantiesiems. Visiems mums rūpi sveikata ir gyvenimo trukmė, todėl prisidėjimas prie medicininių tyrimų yra itin reikšmingas. Šis darbas suteikia galimybę atlikti neįprastą ir naudingą analizę, kuri gali padėti geriau suprasti krūties vėžio dinamiką ir prisidėti prie efektyvesnių gydymo metodų kūrimo.

Šiame darbe bus naudojami „logit“ ir „probit“ binarinio atsako modeliai bei išgyvenamumo analizė, siekiant išsamiai išnagrinėti krūties vėžio mirtis ir išgyvenamumą. Šie statistiniai metodai leis įvertinti įvairius veiksnius, kurie gali turėti įtakos pacientų išgyvenamumui. Tokių metodų taikymas medicinoje yra ypač svarbus, nes leidžia daryti pagrįstas ir logiškas išvadas, kurios gali būti naudingos tiek klinikiniams tyrimams, tiek sveikatos politikos formavimui.

Analizė bus atlikta naudojant „R“ programinę įrangą, kuri yra galingas įrankis statistinei analizei ir duomenų vizualizavimui. Pasirinkome medicinos temą ne tik dėl jos svarbos, bet ir dėl didelio potencialo gauti naudingus rezultatus, kurie galėtų turėti praktinę reikšmę pacientų priežiūroje ir gydyme.

Tikslas

Darbo tikslas yra atlikti regresinę analizę, naudojant du metodus.

Uždaviniai

- Surasti tinkamus duomenis
- Sutvarkyti duomenis
- Atlikti pradinę duomenų analizę
- Pasirinkti tinkamą analizei modelį
- Interpretuoti rezultatus
- Padaryti išvadas

Duomenys

Pasirinkome duomenis apie moterų krūties vėžio išgyvenamumą nuo 1984 iki 1989. Duomenys imti iš R paketo pavadinimu „survival“, duomenų pavadinimas – „gbsg“. Mūsų

duomenų rinkinį sudaro 686 eilutės ir 11 stulpelių su, per 5 metų surinktais duomenimis apie krūties vėžiu sirgusių moterų būklę ir charakteristikomis. Analizei atlikti mes pasirinkome naudoti tokius kintamuosius:

- age – paciento amžius.
- meno – menopauzės statusas (0 - prieš, 1 - po).
- size – naviko dydis (milimetrais).
- grade – naviko stadija.
- nodes – teigiamų limfmazgių skaičius.
- pgr – progesterono receptoriai.
- er – estrogenų receptoriai.
- hormon – hormoninis gydimas (0 - nebuvo, 1- buvo).
- rfstime – laikas iki tiriamo įvykio.
- status – ar žmogus mirė ar ne (0 – pasveiko, 1 – mirė).

Kaip priklausomąjį kintamąjį imame „status“, kiti kintamieji bus nepriklausomi. Naudojant ir testuojant „Logit“ ir „Probit“ binarinio atsako modelius, mes dalinsime duomenis į „train“ ir „test“ santykiu 70/30.

Pradinė analizė

Pirmiausia atliekame pradinę analizę ir pradedame viską nuo priklausomojo kintamojo susipažinimo. Mūsų atveju jis yra kategorinis kintamasis, kuris nusako ar moteris pasveiko ar mirė nuo krūties vėžio, todėl galime sudaryti lentelę. 0 reiškia, kad pacientė pasveiko, o 1 reiškia, kad pacientė mirė.

1 lentelė priklausomojo kintamojo pasiskirstymas

Pasveiko - 0	Mirė - 1
387	299

Matome, kad pasveikusių ir mirusių pacientų skaičius yra gana vienodas. Tai mums gali padėti sudarant modelį.

Toliau pažiūrėsime kitus kategorinius parametrus ir jų pasiskirstymus tokius kaip: menopauzės, hormonų bei krūties vėžio stadija. Visi jie pavaizduoti lentelėmis.

2 lentelė menopauzės statuso pasiskirstymas

Prieš menopauzę - 0	Po menopauzės - 1
290	396

3 lentelė hormonų terapijos statuso pasiskirstymas

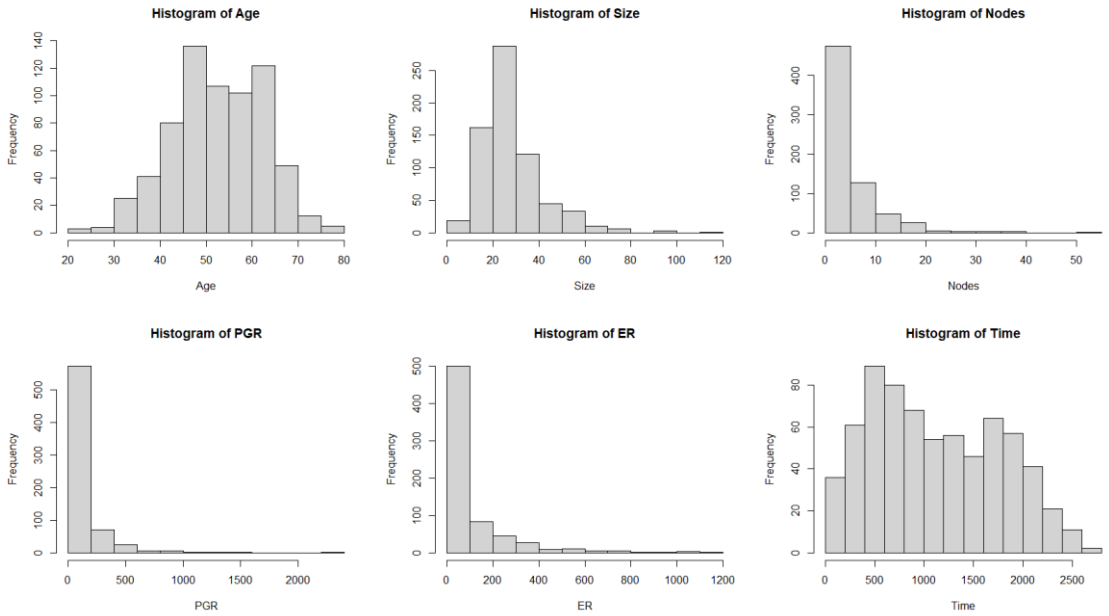
Neskirta hormonų terapija - 0	Skirta hormonų terapija - 1
440	246

4 lentelė vėžio stadijos pasiskirstymas

Pirma stadija - 1	Antra stadija - 2	Trečia stadija - 3
81	444	161

Aiškiai matome, kad daugumoje pasiskirstymai yra nelabai tolygūs, bet tai neturėtų trukdyti mūsų modelio kūrimui ir rezultatų interpretavimui.

Toliau galime pasižiūrėti į mūsų kiekybinių duomenų pasiskirstymo grafikus. Turėjome tokius kiekybinius duomenis: paciento amžius, auglio dydis, progesterono ir estrogenų receptorių kiekis, užkrėsti limfmazgiai, bei laikas, po kurio atsitiko įvykis. Visi jie pavaizduoti dažnių histogramomis.

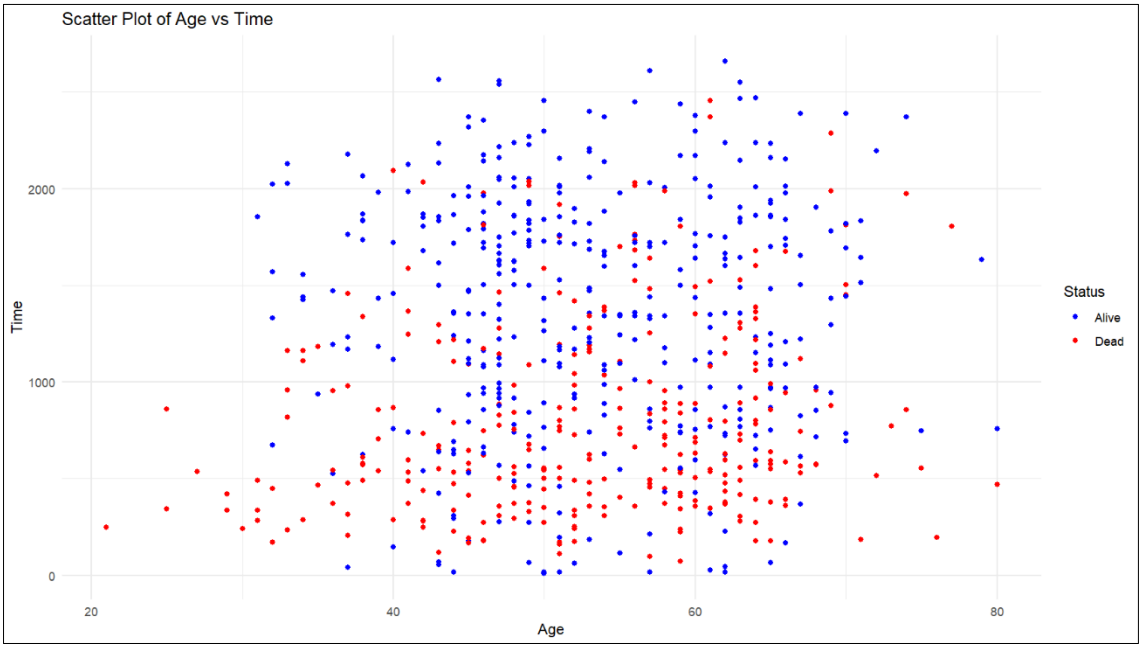


1 pav. kiekybinių kintamųjų pasiskirstymas grafiškai

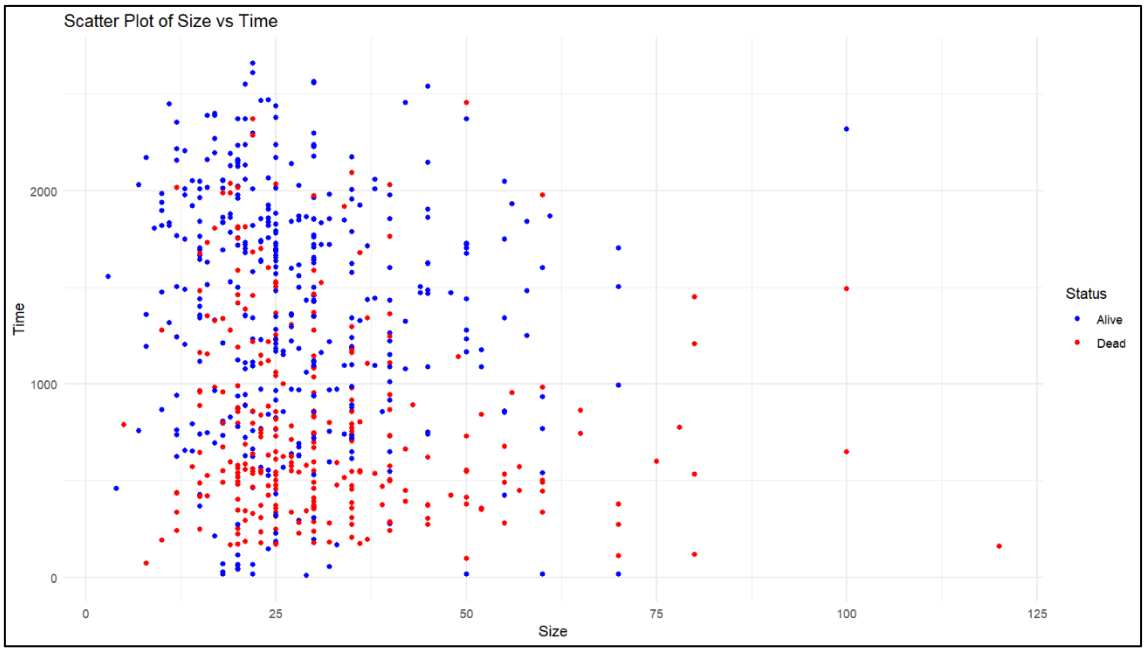
Aiškiai matome, kad turime reikšmių, kurios yra neįprastos ir ekstremalios, tačiau pačias išskirtis tikrinsime jau tik sudarę modelį.

Toliau atliekame pirmo tipo cenzūravimą iš dešinės, kurio metu sujungiame priklausomąjį kintamąjį „status“, kuris nurodo ar žmogus pasveiko ar mirė, su nepriklausomu kintamuoju „rfstime“, kuris nurodo kiek dienų reikėjo įvykiui atsitikti.

Iš histogramų galėjome pastebėti, kad mūsų 3 kintamieji turi perteklines nulių reikšmes, todėl dabar norėtumėme patikrinti, kaip atrodo sklaidos diagramos, likusių narių, laiko atžvilgiu.



2 pav. laiko ir amžiaus taškinė diagrama pagal įvykį

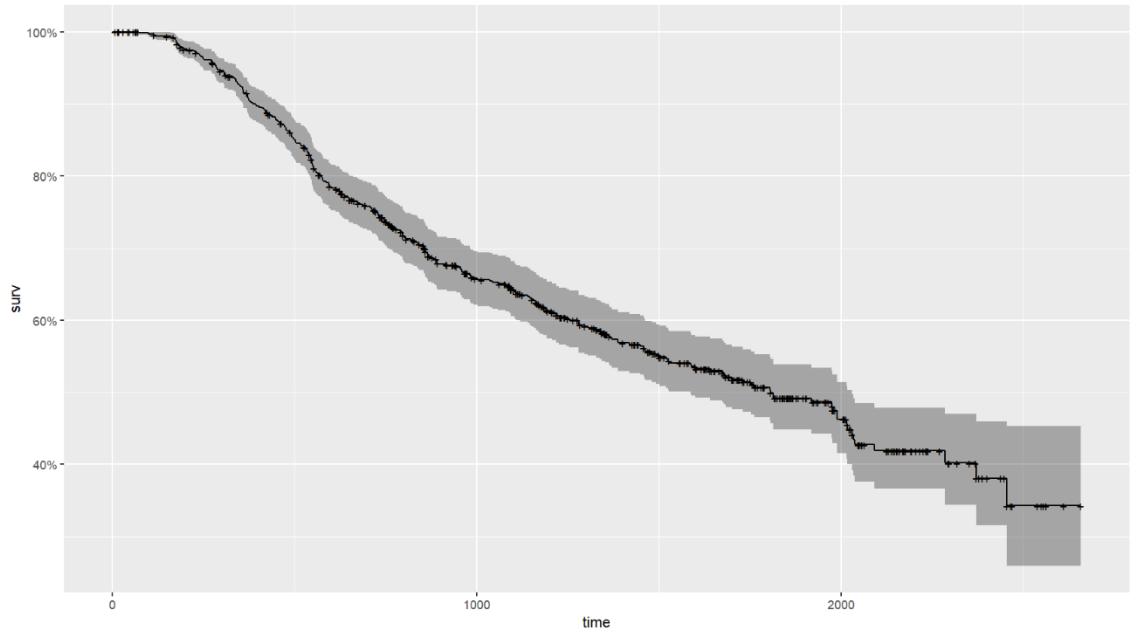


3 pav. laiko ir auglio dydžio taškinė diagrama pagal įvykį

Čia mėlyna spalva žymi pasveikusius žmones, o raudona – mirusius nuo ligos.

Iš šių grafikų matome, kad yra daugiau žmonių kurie serga trumpai ir miršta anksti, taip pat ir atvirkščiai, kurie serga ir gydosi ilgai, ir kuriems pavyksta įveikti liga. Kažkokių išskirtinų požymių lyginant laiką su auglio dydžiu ir paciento amžiumi nepastebėjome.

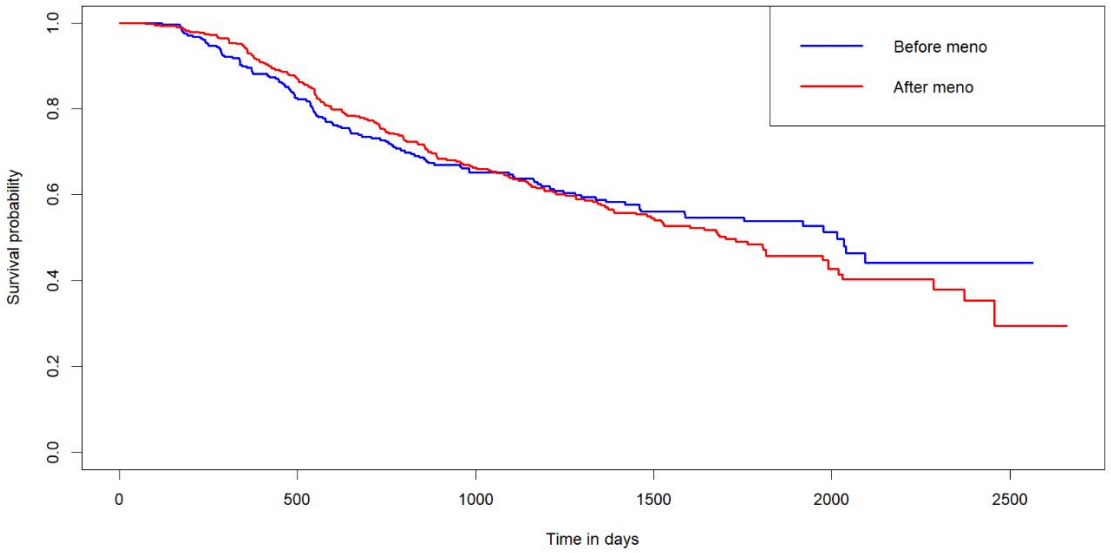
Tada pavaizduojame bendrą „Kaplan-Meier“ įvertį grafike, kuris rodo, kaip keičiasi išgyvenimo tikimybė per laiką.



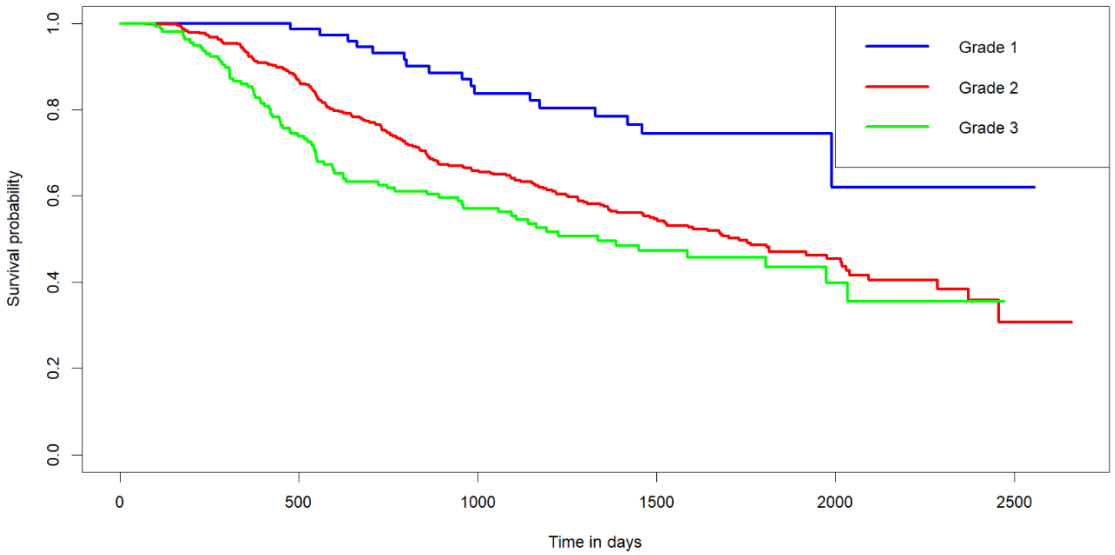
4 pav. Kaplan-Meier grafikas

Mūsų grafike galime pamatyti, kad jeigu jau yra sergama bent 2000 dienų (~6 metų), tikimybė išgyventi yra apie 48%.

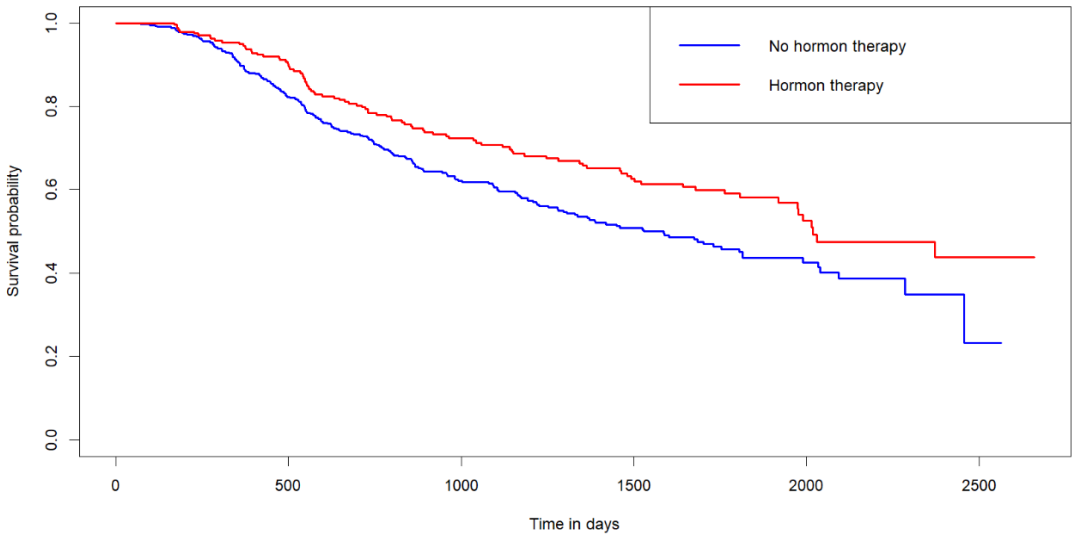
Čia matome tokius pat „Kaplan-Meier“ grafikus, tik jau išskirstytą pagal kategorijas: pagal taikytą hormonų terapiją, pagal menopauzės statusą ir pagal vėžio stadiją. Grafike kategorijos išskirstytos spalvomis.



5 pav. Kaplan-Meier grafikas pagal menopauzės statusą



6 pav. Kaplan-Meier grafikas pagal auglio dydį



7 pav. Kaplan-Meier grafikas pagal hormonų terapijos statusą

Iš grafikų pastebime, kad linijos susikerta, kai pacientus skirstome pagal menopauzės statusą, o skirstant pagal hormonų terapiją ir vėžio stadiją, matome, kad linijos išsiskiria.

Toliau atliekame homogeniškumo hipotezės tikrinimą. Pasitelkiame logranginį kriterijų, taip pat naudojome ir Gehan-Wilcoxon kriterijaus Peto ir Peto modifikaciją. Tokį testą galime atlikti tik paskutiniams dvejiems grafikams, nes matome, kad jų grafiko linijos nesusikerta.

```
> survdiff(Surv(df$rfstime, df$status)~df$grade, rho = 0)
Call:
survdiff(formula = Surv(df$rfstime, df$status) ~ df$grade, rho = 0)

               N Observed Expected (O-E)^2/E (O-E)^2/V
df$grade=1    81       18     42.2   13.8469    16.159
df$grade=2   444      202    198.2    0.0725     0.215
df$grade=3   161       79     58.6    7.0788     8.848

 Chisq= 21.1  on 2 degrees of freedom, p= 3e-05
> survdiff(Surv(df$rfstime, df$status)~df$hormon, rho = 0)
Call:
survdiff(formula = Surv(df$rfstime, df$status) ~ df$hormon, rho = 0)

               N Observed Expected (O-E)^2/E (O-E)^2/V
df$hormon=0  440      205      180      3.37      8.56
df$hormon=1  246       94      119      5.12      8.56

 Chisq= 8.6  on 1 degrees of freedom, p= 0.003
```

8 pav. PH prielaida

Iš čia suprantame ir matome, kad mūsų p reikšmė neviršija mūsų nustatytos alfa reikšmės (0,05), todėl galime daryti išvadą, kad didėjant krūties vėžio stadijai, išgyvenamumas

mažėja, taip pat, kad netaikius hormonų terapijos, išgyvenamumas taip pat mažėja. Matosi aiškūs grupių skirtumai.

Tačiau tokios pat išvados negalime daryti su pirmuoju grafiku, kur matome, kaip grafiko linijos persikerta net keletą kartų ir keičiasi pozicijomis. Tokiu atveju mes negalime naudoti logranginio testo, šiuo atveju naudosime dviejų etapų testą (angl. two-stage test), kuris taip pat leidžia daryti išvadas apie homogeniškumą.

```
> twostage(df$rfstime, df$status, df$meno,
+          nboot = 1000)
      LRPV       MTPV       TSPV
0.59657672 0.00400000 0.02921928
```

9 pav. dviejų etapų testas

Mūsų atveju reikia žiūrėti į „TSPV“ reikšmę, kuri turi būti žemesnė negu mūsų nustatyta p reikšmė. Tokį rezultatą šiuo atveju mes ir gauname.

Atlikus tokia pradinę analizę, peržiūrėjus ir susipažinus su duomenimis, galime judėti prie modelių prielaidų tikrinimo, kūrimo, rezultatų interpretavimo ir išvadų.

„Logit“ ir „Probit“ modeliai

„Logit“

Pirmiausia pradėsime nuo „Logit“ binarinio atsako modelio, kur tiesiog susidarysime modelį, su visais esamais kintamaisiais.

```
Call:
glm(formula = status ~ nodes + age + meno + size + grade + er +
    pgr + hormon + rfstime, family = binomial(logit), data = train_data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.3013  -0.8207  -0.4099   0.8671   2.4389

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  2.2393979  1.0038284   2.231  0.02569 *
nodes        0.0771382  0.0243626   3.166  0.00154 **
age         -0.0191501  0.0178262  -1.074  0.28270
meno         0.6939022  0.3608524   1.923  0.05449 .
size         0.0012434  0.0082504   0.151  0.88020
grade       -0.1568378  0.2028047  -0.773  0.43932
er           0.0006733  0.0008253   0.816  0.41465
pgr         -0.0018027  0.0007590  -2.375  0.01755 *
hormon      -0.2229638  0.2451593  -0.909  0.33310
rfstime     -0.0017066  0.0002014  -8.472  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 655.69  on 480  degrees of freedom
Residual deviance: 506.42  on 471  degrees of freedom
AIC: 526.42

Number of Fisher Scoring iterations: 4
```

10 pav. Pradinis „Logit“ modelis

Matome daug regresorių yra nereikšmingų, nes jų p reikšmės viršija nustatytą reikšmingumo lygmenį alfa (0.05). Tai reiškia, kad turime atlikti pažingsninę regresiją ir pasilikti tik reikšmingus kintamuosius.

```
Call:
glm(formula = status ~ nodes + pgr + rfstime, family = binomial(logit),
    data = train_data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2364  -0.8794  -0.4757   0.9277   2.1381

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.1688709  0.2538494   4.605 4.13e-06 ***
nodes        0.0713623  0.0227392   3.138   0.0017 **
pgr         -0.0015350  0.0006480  -2.369   0.0179 *
rfstime     -0.0014196  0.0001799  -7.889 3.05e-15 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 662.96  on 480  degrees of freedom
Residual deviance: 545.79  on 477  degrees of freedom
AIC: 553.79

Number of Fisher Scoring iterations: 4
```

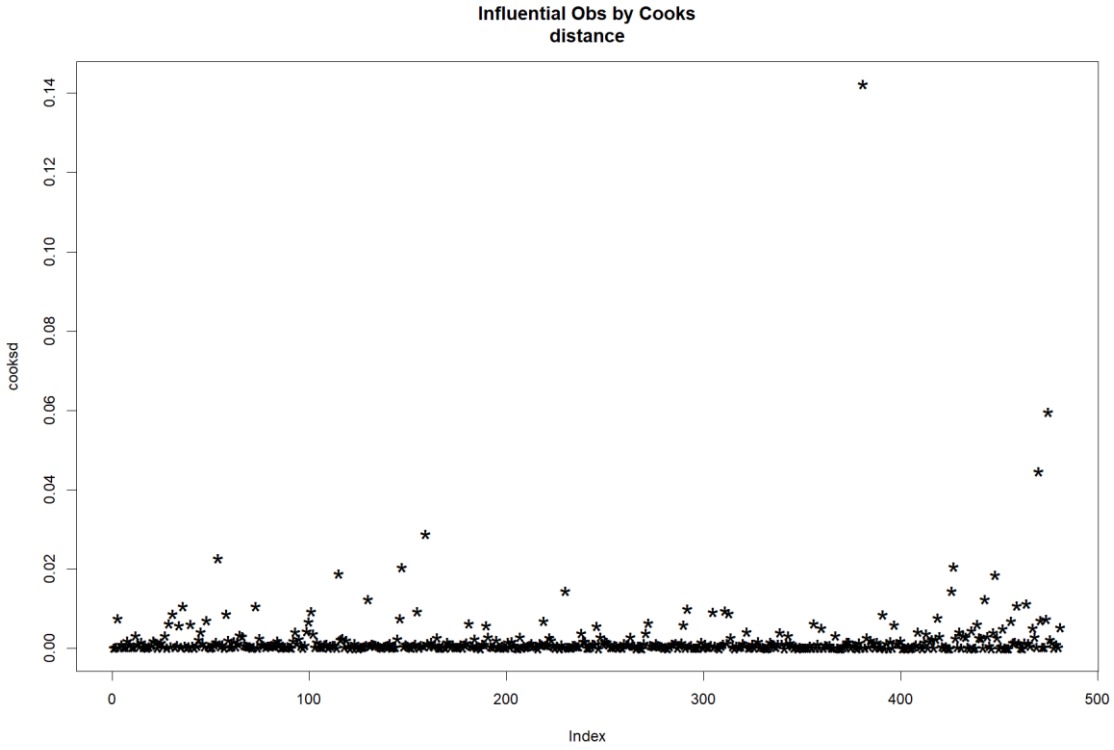
11 pav. „Logit“ modelis su reikšmingais kintamaisiais

Jau dabar matome, kad p reikšmės yra mažesnės negu reikšmingumo lygmuo alfa. Taip pat pastebime AIC indeksą kuris yra 553.8, jo gali mums prireikti lyginant modelius. Toliau tikrinsime multikoliearumą su VIF funkcija.

```
> print(vif_logit_values)
   nodes      pgr  rfstime
1.023918 1.008994 1.020846
```

12 pav. VIF

Matome, kad nei viena reikšmė neviršija 4, Tai reiškia, kad multikolinearumo problemos nėra ir prediktoriai nėra tarpusavy stipriai koreliuojantys. Toliau galime žiūrėti į modelio išskirtis.



13 pav. Kuko matas

Iš Kuko mato grafiko galime pastebėti, kad išskirčių neturime, nes nei vienas taškas neviršija 1, todėl toliau pradėsime tikrinti modelio tinkamumą ir pradėsime nuo tikėtinumo santykių kriterijaus.

```
Analysis of Deviance Table

Model 1: status ~ 1
Model 2: status ~ +nodes + pgr + rfstime
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1       480     662.96
2       477     512.59  3   150.37 < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
```

14 pav. tikėtinumo santykių kriterijaus testas

Tam, kad modelis būtų tinkamas naudoti ir spėti, p reikšmė turi būti mažesnė negu reikšmingumo lygmuo. Būtent tai ir matome atlikę šį testą. Tai parodo, kad modelis tinkamas naudojimui. Toliau atliekame Wald’o testą, kuris parodys ar mūsų prognozuojamųjų kintamųjų regresijos koeficientai nėra nuliai.

```
> wald_result_logit
Wald test

Model 1: status ~ +nodes + pgr + rfstime
Model 2: status ~ 1
  Res.Df Df  F    Pr(>F)
1    477
2    480 -3 30 < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
```

15 pav. Wald‘o testas

Mūsų atveju matome, kad p-reikšmė yra labai netoli 0, todėl atmetame nuline hipotezę ir priimame alternatyvą, kuri mums leidžia daryti prielaidą, kad mūsų prognozuojamųjų kintamųjų regresijos koeficientai nėra nuliai. Na ir galiausiai galime pasižiūrėti į modelio tikslumą pagal jo spėjimus. Tai yra klasifikavimo lentelė.

```
> ClassLog(model_logit_rez_final, train_data$status)
$rawtab
       resp
          0   1
  FALSE 225  63
  TRUE   52 141

$classtab
       resp
                  0         1
  FALSE 0.8122744 0.3088235
  TRUE  0.1877256 0.6911765

$overall
[1] 0.7609148

$mcFadden
[1] 0.2182378
```

16 pav. klasifikavimo lentelė

Matome, kad mūsų modelio bendras tikslumas yra tik 76%, kas yra labai vidutiniškai, tačiau nėra didžiulė tragedija. Modelis teisingai spėju pasveikimus 81% tikslumu, o mirtis – 69%.

Dabar susidarysime „Probit“ modelį, su tais pačiais etapais.

„Probit“

Ir vėl pirmiausia pradėsime nuo „Probit“ binarinio atsako modelio, kur tiesiog susidarysime modelį, su visais esamais kintamaisiais.

```
Call:
glm(formula = status ~ nodes + age + meno + size + grade + er +
    pgr + hormon + rfstime, family = binomial(probit), data = train_data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.3217  -0.8389  -0.3963   0.8783   2.5011

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.2707886  0.5846113   2.174  0.02973 *
nodes        0.0449665  0.0139256   3.229  0.00124 **
age         -0.0102192  0.0104135  -0.981  0.32642
meno         0.4035344  0.2111066   1.912  0.05594 .
size         0.0007068  0.0048159   0.147  0.88332
grade       -0.0923656  0.1192848  -0.774  0.43874
er           0.0004278  0.0004845   0.883  0.37722
pgr         -0.0010623  0.0004276  -2.484  0.01297 *
hormon      -0.1509681  0.1435665  -1.052  0.29300
rfstime     -0.0010020  0.0001133  -8.845  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 655.69  on 480  degrees of freedom
Residual deviance: 507.34  on 471  degrees of freedom
AIC: 527.34

Number of Fisher Scoring iterations: 5
```

17 pav. pradinis „Probit“ modelis

Matome, kad ir vėl daug regresorių yra nereikšmingų, nes jų p reikšmės viršija nustatytą reikšmingumo lygmenį alfa. Tai reiškia, kad turime atlikti pažingsninę regresiją ir pasilikti tik reikšmingus kintamuosius.

```
Call:
glm(formula = status ~ +nodes + pgr + rfstime, family = binomial(probit),
    data = train_data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.3405  -0.8301  -0.4475   0.9086   2.3328

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  0.7579403  0.1606026   4.719 2.37e-06 ***
nodes        0.0435289  0.0134646   3.233  0.00123 **
pgr         -0.0008736  0.0003765  -2.320  0.02032 *
rfstime     -0.0009876  0.0001100  -8.980  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 655.69  on 480  degrees of freedom
Residual deviance: 513.72  on 477  degrees of freedom
AIC: 521.72

Number of Fisher Scoring iterations: 4
```

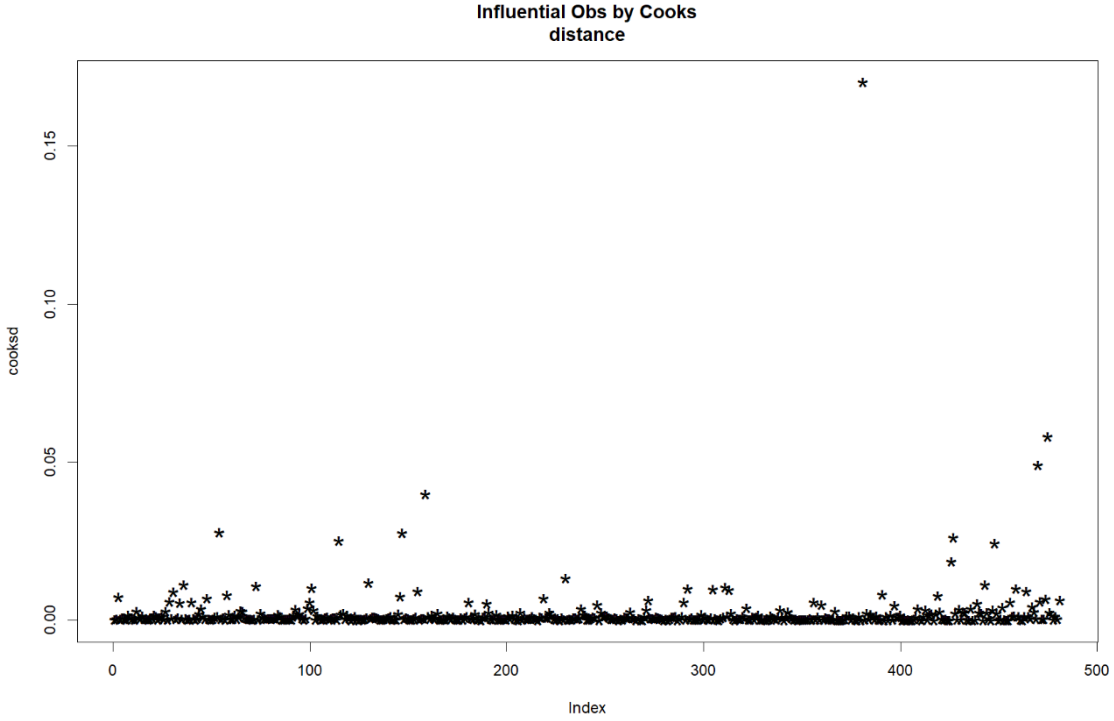
18 pav. „Probit“ modelis su reikšmingais kintamaisiais

Jau dabar matome, kad p reikšmės yra mažesnės negu reikšmingumo lygmuo alfa. Taip pat pastebime AIC indeksą kuris yra 521.7. Jis yra mažesnis negu „logit“ binarinio atsako modelio ir tai indikuoja, kad jis yra šiek tiek geresnis. Toliau tikrinsime multikoliearumą su VIF funkcija.

```
> vif_probit_values <- vif(model_probit_rez_final)
> print(vif_probit_values)
   nodes      pgr  rfstime
1.040434 1.012556 1.037937
```

19 pav. VIF

Matome, kad nei viena reikšmė neviršija 4, Tai reiškia, kad multikolinearumo problemos nėra ir prediktoriai nėra tarpusavy stipriai koreliuojantys. Toliau galime žiūrėti į modelio išskirtis.



20 pav. Kuko matas

Iš Kuko mato grafiko ir vėl galime pastebėti, kad išskirčių neturime, nes nei vienas taškas neviršija 1, todėl toliau pradėsime tikrinti modelio tinkamumą ir pradėsime nuo tikėtinumo santykių kriterijaus.

```
> anova(mprobit.reduced,model_probit_rez_final, test="Chisq")
Analysis of Deviance Table

Model 1: status ~ 1
Model 2: status ~ +nodes + pgr + rfstime
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1       480     655.69
2       477     513.72  3   141.96 < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
```

21 pav. tikėtinumo santykio kriterijaus testas

Tam, kad modelis būtų tinkamas naudoti ir spėti, p reikšmė turi būti mažesnė negu reikšmingumo lygmuo. Būtent tai ir matome atlikę šį testą. Tai parodo, kad modelis tinkamas naudojimui. toliau atliekame Wald’o testą, kuris parodys ar mūsų prognozuojamųjų kintamųjų regresijos koeficientai nėra nuliai.

```
> wald_result_probit <- waldtest(model_probit_rez_final, vcov = vc
> wald_result_probit
Wald test

Model 1: status ~ +nodes + pgr + rfstime
Model 2: status ~ 1
  Res.Df Df      F    Pr(>F)
1    477
2    480 -3 32.853 < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

22 pav. Wald‘o testas

Mūsų atveju ir vėl matome, kad p reikšmė yra labai netoli 0, todėl atmetame nuline hipotezę ir priimame alternatyvą, kuri mums leidžia daryti prielaidą, kad mūsų prognozuojamųjų kintamųjų regresijos koeficientai nėra nuliai. Na ir galiausiai galime pasižiūrėti į modelio tikslumą pagal jo spėjimus. Tai yra klasifikavimo lentelė.

```
> ClassLog(model_probit_rez_final, train_data$status)
$rawtab
       resp
          0   1
  FALSE 225  63
  TRUE   52 141

$classtab
       resp
                 0         1
  FALSE 0.8122744 0.3088235
  TRUE  0.1877256 0.6911765

$overall
[1] 0.7609148

$mcFadden
[1] 0.2165075
```

23 pav. klasifikavimo lentelė

Matome, kad mūsų modelio bendras tikslumas yra taip pat 76%, kas yra labai vidutiniškai, tačiau nėra didžiulė tragedija. Modelis teisingai spėju pasveikimus 81% tikslumu, o mirtis 69 %. Iš atliktų abiejų nuodugnių „logit“ ir „probit“ modelių testavimų, mes renkames „Logit“ modelį dėl šiek tiek lengvesnės interpretacijos.

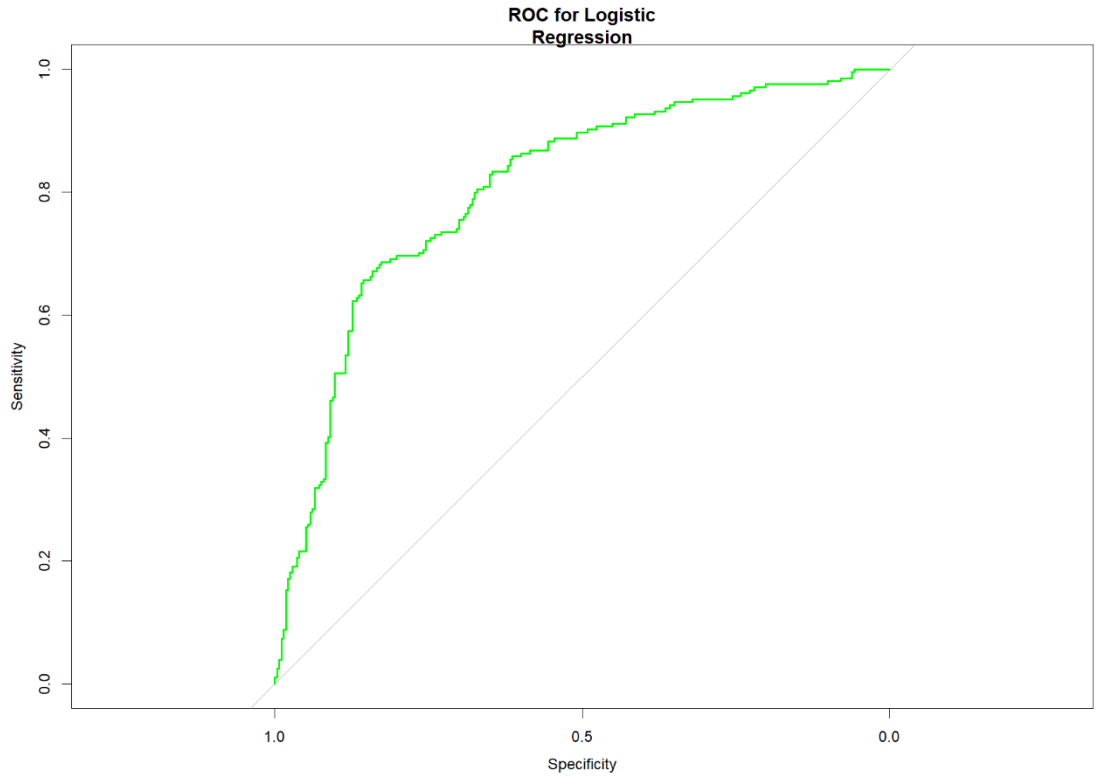
Taigi pasirinkę tinkamesnį modelį, galime interpretuoti ir pavaizduoti rezultatus.

```
> exp(coef(model_logit_rez_final))
(Intercept)       nodes         pgr     rfstime
  3.6230518   1.0773264   0.9985060   0.9983228
> # Intervals
> exp(confint.default(model_logit_rez_final, level=0.9))
                   5 %       95 %
(Intercept) 2.3151869 5.6697385
nodes       1.0363135 1.1199625
pgr         0.9973987 0.9996146
rfstime     0.9980020 0.9986436
>
```

24 pav. galimybių santykiai ir pasikliovimo intervalai

Čia matome galimybių santykius, jie rodo kiek pasikeistų tikimybė jei kažkuris iš kintamųjų padidėtų procentiniu vienetu, o kiti liktų tokie patys.

Toliau nusibrėžėme ROC kreivę. Slenkstis bus imamas 0,5.



25 pav. „Logit“ modelio ROC kreivė

Pabaigoje pratestavome savo duomenis su test duomenų rinkiniu ir matome, kad 73% modelis teisingai atspėjo pasveikusių žmones ir 61% mirusių žmones. Bendras modelio tikslumas pagal testavimo duomenis 67%.

```
> print(confusion_matrix)

             Predicted Alive Predicted Dead
Actual Alive              80              30
Actual Dead               37              58
>
```

Negalime sakyti, kad modelis spėja gerai, bet ir nėra, kad jis spėliotų.

Išgyvenamumo analizė

Pirmiausia susidarome modelį su visais kintamaisiais ir žiūrėsime, kad p reikšmės neviršytų alfa reikšmės.

```
Call:
coxph(formula = Surv(rfstime, status) ~ age + meno + size + grade +
    nodes + pgr + er + hormon, data = df)

  n= 686, number of events= 299

              coef  exp(coef)   se(coef)      z Pr(>|z|)
age    -0.0093924  0.9906516  0.0092733 -1.013 0.311136
meno    0.2672772  1.3064025  0.1833366  1.458 0.144882
size    0.0077164  1.0077463  0.0039497  1.954 0.050739 .
grade   0.2802894  1.3235128  0.1060553  2.643 0.008221 **
nodes   0.0498939  1.0511596  0.0074094  6.734 1.65e-11 ***
pgr    -0.0022378  0.9977647  0.0005758 -3.887 0.000102 ***
er      0.0001674  1.0001674  0.0004477  0.374 0.708431
hormon -0.3372029  0.7137640  0.1289618 -2.615 0.008929 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

       exp(coef) exp(-coef) lower .95 upper .95
age       0.9907     1.0094    0.9728    1.0088
meno      1.3064     0.7655    0.9121    1.8713
size      1.0077     0.9923    1.0000    1.0156
grade     1.3235     0.7556    1.0751    1.6293
nodes     1.0512     0.9513    1.0360    1.0665
pgr       0.9978     1.0022    0.9966    0.9989
er        1.0002     0.9998    0.9993    1.0010
hormon    0.7138     1.4010    0.5543    0.9190

Concordance= 0.688  (se = 0.015 )
Likelihood ratio test= 101.9  on 8 df,   p=<2e-16
Wald test            = 115.3  on 8 df,   p=<2e-16
Score (logrank) test = 120.1  on 8 df,   p=<2e-16
```

26 pav. pilnas išgyvenamumo analizės modelis

Matome tik kelis nereikšmingus parametrus, tačiau neskubame jų mesti, pirmiausia patikrinsime VIF dėl multikolinearumo problemos.

```
> VIF(cox)
     age     size    nodes      pgr       er
1.109832 1.213154 1.207427 1.072860 1.188526
```

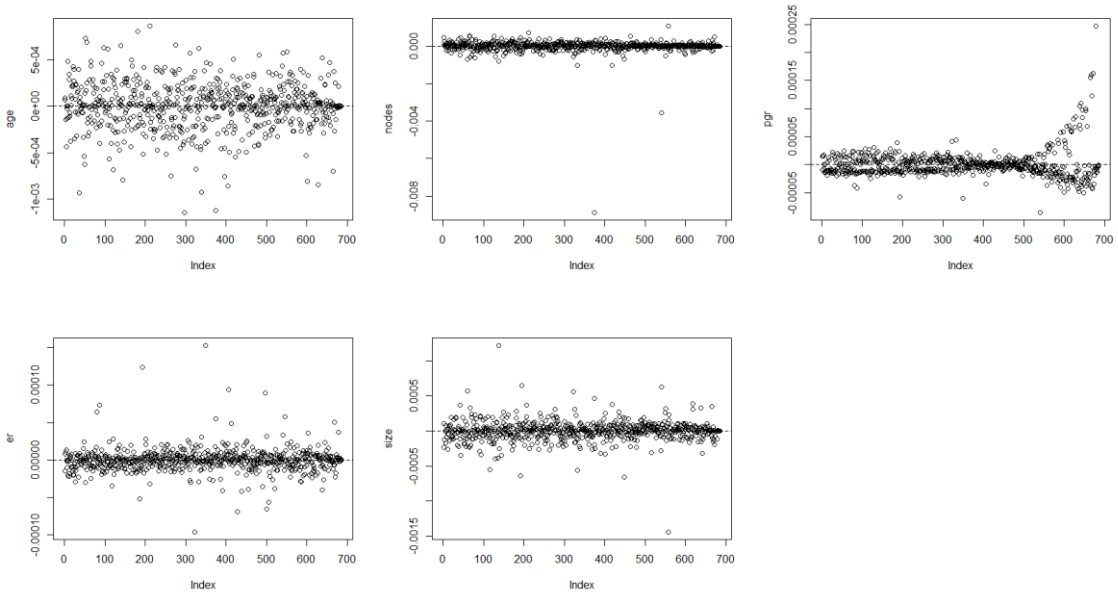
27 pav. VIF

Aiškiai matosi, kad multikolinearumo problemos neturime, nes visi skaičiai neviršija 4. Tai reiškia, kad tada galime patikrinti išskirtis. Pirmiausia pagal formule susiskaičiuojame rėžį, kuris indikuos, kad modelyje yra išskirčių. Viršutinio rėžio formulė yra: 2/n^0.5 , kur n yra visų stebėjimų skaičius.

```
> 2/sqrt(nrow(df))
[1] 0.07636035
```

28 pav. rėžis

Gauname skaičių 0.70636035, taigi dabar pagal šį rėžį žiūrėsime ar neturime išskirčių duomenyse, pagal pritaikytą modelį. Išskirtis tikriname tik kiekybiniams parametrams.



29 pav. PH prielaida grafiškai

Iš šios standartizuotų koeficientų matricos matome, kad mūsų apskaičiuotas rėžis visada yra aukščiausiai ir nėra taško, kuris viršytų rėžį, todėl darome išvadą, kad išskirčių neturėsime.

Po šių etapų, jau galime atlikti pažingsninę regresiją ir pasilikti tik reikšmingus regresorius.

```
Call:
coxph(formula = Surv(rfstime, status) ~ pgr + grade + nodes +
    hormon, data = df)

  n= 686, number of events= 299

             coef  exp(coef)   se(coef)      z Pr(>|z|)
pgr    -0.0022354  0.9977671  0.0005601 -3.991 6.57e-05 ***
grade   0.2935538  1.3411854  0.1054964  2.783  0.00539 **
nodes   0.0552483  1.0568030  0.0067679  8.163 3.26e-16 ***
hormon -0.3103666  0.7331781  0.1255928 -2.471  0.01347 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

       exp(coef) exp(-coef) lower .95 upper .95
pgr       0.9978     1.0022    0.9967    0.9989
grade     1.3412     0.7456    1.0907    1.6493
nodes     1.0568     0.9463    1.0429    1.0709
hormon    0.7332     1.3639    0.5732    0.9378

Concordance= 0.682  (se = 0.015 )
Likelihood ratio test= 96.22  on 4 df,   p=<2e-16
Wald test            = 109.9  on 4 df,   p=<2e-16
Score (logrank) test = 115.4  on 4 df,   p=<2e-16
```

30 pav. modelis su reikšmingomis kovariantėmis

Liekame su tokiais keturiais regresoriais: pgr, grade, nodes ir hormon. Grade ir hormon yra kategoriniai kintamieji, o pgr ir nodes yra kiekybiniai kintamieji.

Toliau testuojame PH prielaidą. Funkcija „cox.zph“ realizuotas PH prielaidos tikrinimo kriterijus. Jis yra skaičiuojamas kiekvienai kovariantei bei grindžiamas Schoenfeld liekanų koreliacija su transformuotu laiko kintamuoju (pagal nutylėjimą naudojamas K-M įvertis).

```
> cox.zph(cox)
        chisq df      p
pgr     4.164  1 0.0413
grade   7.788  1 0.0053
nodes   1.246  1 0.2643
hormon  0.197  1 0.6574
GLOBAL 11.712  4 0.0196
```

31 pav. pirmoji PH prielaida

Matome, kad dvi reikšmės, nodes ir hormon tenkina prielaidą, o pgr ir grade, ne. Todėl nusprendėme išsibandyti keletą saveikų, ir radome 2 sėkmingas, kurios tenkina šią prielaidą ir modelyje yra reikšmingos. Tai buvo „pgr“ ir „grade“, būtent tos kurios atskirai prielaidos netenkino, o įtraukiant jų saveiką – tenkino. Taip pat pabandėme patikrinti ir jau tenkinančių prielaidų sąveikas „nodes“ ir „hormon“, kurios taip pat buvo sėkmingos. Taigi mūsų galutinis modelis atrodo taip:

```
Call:
coxph(formula = Surv(rfstime, status) ~ pgr:grade + nodes * hormon,
    data = df)

  n= 686, number of events= 299

                  coef  exp(coef)   se(coef)      z Pr(>|z|)
nodes         0.0487092  1.0499150  0.0081598  5.969 2.38e-09 ***
hormon       -0.5716502  0.5645930  0.1631710 -3.503 0.000459 ***
pgr:grade    -0.0011073  0.9988933  0.0002778 -3.986 6.72e-05 ***
nodes:hormon  0.0356470  1.0362900  0.0149185  2.389 0.016874 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

             exp(coef) exp(-coef) lower .95 upper .95
nodes           1.0499     0.9525    1.0333    1.0668
hormon          0.5646     1.7712    0.4101    0.7774
pgr:grade       0.9989     1.0011    0.9983    0.9994
nodes:hormon    1.0363     0.9650    1.0064    1.0670

Concordance= 0.675  (se = 0.016 )
Likelihood ratio test= 87.15  on 4 df,   p=<2e-16
Wald test            = 101  on 4 df,   p=<2e-16
Score (logrank) test = 102.9  on 4 df,   p=<2e-16
```

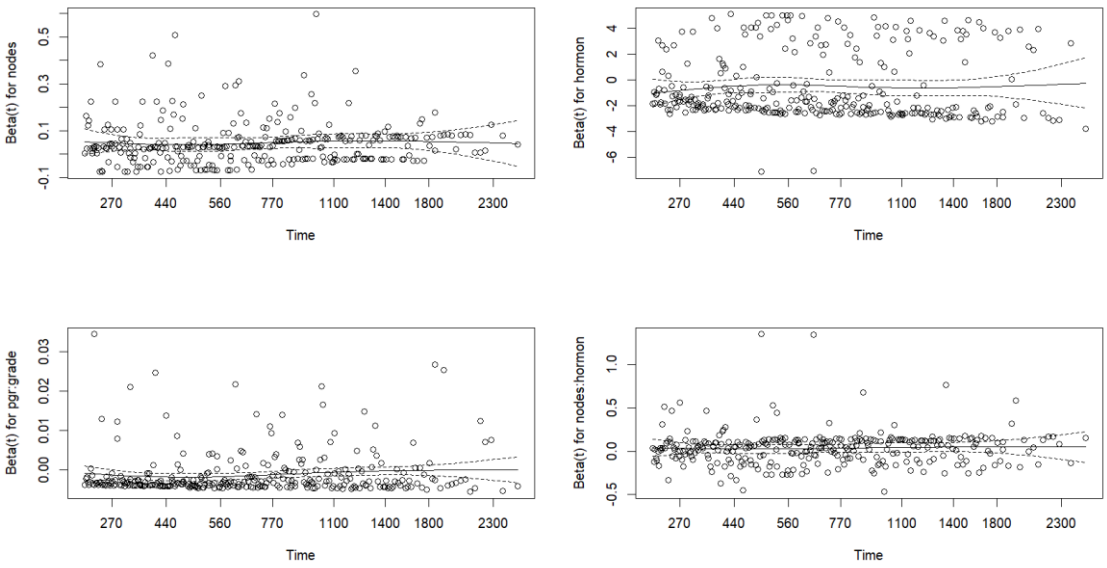
32 pav. galutinis modelis su sąveikomis

Matome, kad konkordacijos koeficientas nėra labai aukštas, tik 0.68, bet tai tikrai nereiškia, kad modelis spėlioja. Taip pat patikrinome ir PH prielaidas su sąveikomis, kurios taip pat tiko.

```
> cox.zph(cox)
              chisq df     p
nodes         1.257  1 0.262
hormon        0.897  1 0.343
pgr:grade     2.797  1 0.094
nodes:hormon  2.120  1 0.145
GLOBAL        5.317  4 0.256
>
```

33 pav. antroji PH prielaida su sąveikomis

Matome, kad jau visos p reikšmės viršija nustatytą reikšmingumo lygmenį, todėl galime daryti išvadą, kad modelį galima interpretuoti ir gauti tinkamus rezultatus. Taip pat patikrinsime prielaidą grafiškai.



34 pav. PH prielaidos tikrinimas grafiškai su sąveikomis

Čia įsitikiname ir gauname tuos pačius rezultatus.

Išvados

Binarinio atsako modeliai ir išgyvenamumo analizė atrodo kaip puiki dviejų metodų kombinacija, kuri darant išvadas ir teikiant rezultatus, viena kitą papildo. Binarinio atsako modeliai padėjo nustatyti gana tikslias įvykio arba ne įvykio reikšmes pagal tam tikrus parametrus, tokius, kaip ir pats laikas iki įvykio, o išgyvenamumo analizė būtent naudojo įvykio ir ne įvykio bei laiko kombinacija rezultatams gauti. Abu modeliai teikia panašius rezultatus, gal net atrodo, kad vienodus, tačiau šie du metodai labai gerai vienas kitą papildo ir leidžia gauti geresnį supratimą apie duomenis. Mūsų manymu šiuos du metodus tikrai galima pritaikyti tokio pobūdžio duomenims ir pridėjus tokius modelių patobulinimus kaip sąveikos, galima gauti tikrai išsamius rezultatus. Mūsų išgyvenamumo analizės modelis veikė geriau, nes rinkomės duomenis, pritaikytus tokiai analizei, o „Logit“ ir „Probit“ modeliai gavosi jau ne tokie geri, kur tikslumai nesiekė net 80%.

Šaltiniai

- http://www.statistika.mif.vu.lt/wp-content/uploads/2014/04/regresine-analize.pdf
- Qiu, P., & Sheng, J. (2008). A two-stage procedure for comparing hazard rate functions. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, *70*(1), 191-208.