Booklet of Figures

for

STAD29/STA 1007 Midterm Exam

List of Figures in this document by page:

List of Figures

```r
library(MASS)
library(lubridate)
library(tidyverse)
library(broom)
library(survival)
library(survminer)
```

Figure 1: Packages

```
## # A tibble: 200 x 7
##       id sta     age can   cpr   inf   race
##    <int> <fct> <int> <fct> <fct> <fct> <fct>
##  1     4 Died     87 No    No    Yes   White
##  2     8 Lived    27 No    No    Yes   White
##  3    12 Lived    59 No    No    No    White
##  4    14 Lived    77 No    No    No    White
##  5    27 Died     76 No    No    Yes   White
##  6    28 Lived    54 No    No    Yes   White
##  7    32 Lived    87 No    No    Yes   White
##  8    38 Lived    69 No    No    Yes   White
##  9    40 Lived    63 No    No    No    White
## 10    41 Lived    30 No    No    No    White
## 11    42 Lived    35 No    No    No    Black
## 12    47 Died     78 No    No    Yes   White
## 13    50 Lived    70 Yes   No    No    White
## 14    51 Lived    55 No    No    Yes   White
## 15    52 Died     63 No    No    Yes   White
## 16    53 Lived    48 Yes   No    No    Black
## 17    58 Lived    66 No    No    No    White
## 18    61 Lived    61 No    No    No    White
## 19    73 Lived    66 No    No    No    White
## 20    75 Lived    52 No    No    Yes   White
## 21    82 Lived    55 No    No    Yes   White
## 22    84 Lived    59 No    No    Yes   White
## 23    92 Lived    63 No    No    No    White
## 24    96 Lived    72 No    No    No    White
## 25    98 Lived    60 No    Yes   Yes   White
## 26   100 Lived    78 No    No    No    White
## 27   102 Lived    16 No    No    No    White
## 28   111 Lived    62 No    No    No    White
## 29   112 Lived    61 No    No    Yes   White
## 30   127 Died     19 No    No    No    White
## # ... with 170 more rows
```

Figure 2: ICU data (some)

```
icu.1=glm(sta~age+can+cpr+inf+race, family=binomial, data=icu)
summary(icu.1)

##
## Call:
## glm(formula = sta ~ age + can + cpr + inf + race, family = binomial,
##     data = icu)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3703  -0.6823  -0.5421  -0.3082   2.5124
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.51152    0.81443  -4.312 1.62e-05 ***
## age          0.02712    0.01159   2.340  0.01926 *
## canYes       0.24451    0.61681   0.396  0.69180
## cprYes       1.64650    0.62341   2.641  0.00826 **
## infYes       0.68067    0.38042   1.789  0.07357 .
## raceBlack   -0.95708    1.08445  -0.883  0.37748
## raceOther    0.25975    0.87127   0.298  0.76561
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 200.16  on 199  degrees of freedom
## Residual deviance: 179.30  on 193  degrees of freedom
## AIC: 193.3
##
## Number of Fisher Scoring iterations: 5

drop1(icu.1, test="Chisq")

## Single term deletions
##
## Model:
## sta ~ age + can + cpr + inf + race
##        Df Deviance    AIC    LRT Pr(>Chi)
## <none>      179.30 193.30
## age     1   185.63 197.63 6.3305 0.011868 *
## can     1   179.45 191.45 0.1521 0.696555
## cpr     1   186.14 198.14 6.8360 0.008934 **
## inf     1   182.53 194.53 3.2263 0.072463 .
## race    2   180.41 190.41 1.1069 0.574959
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 3: ICU model 1

```r
icu.2 <- step(icu.1)
```

```r
summary(icu.2)

##
## Call:
## glm(formula = sta ~ age + cpr + inf, family = binomial, data = icu)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3633  -0.6810  -0.5524  -0.3091   2.4868
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.57604    0.77306  -4.626 3.73e-06 ***
## age          0.02792    0.01136   2.458  0.01397 *
## cprYes       1.63066    0.61553   2.649  0.00807 **
## infYes       0.69708    0.37750   1.847  0.06481 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 200.16  on 199  degrees of freedom
## Residual deviance: 180.51  on 196  degrees of freedom
## AIC: 188.51
##
## Number of Fisher Scoring iterations: 5
```

Figure 4: ICU model 2

```
##       id            sta            age           can       cpr       inf        race
##  Min.   :  4.0   Lived:160   Min.   :16.00   No :180   No :187   No :116   White:175
##  1st Qu.:210.2   Died : 40   1st Qu.:46.75   Yes: 20   Yes: 13   Yes: 84   Black: 15
##  Median :412.5               Median :63.00                                 Other: 10
##  Mean   :444.8               Mean   :57.55
##  3rd Qu.:671.8               3rd Qu.:72.00
##  Max.   :929.0               Max.   :92.00
```

Figure 5: ICU data summary

```
##   rowid     type  predicted  std.error   conf.low conf.high   age cpr inf
## 1     1 response 0.09357968 0.02869834 0.05050719 0.1669266 46.75  No  No
## 2     2 response 0.17170188 0.04839832 0.09616344 0.2876904 46.75  No Yes
## 3     3 response 0.34524029 0.14429291 0.13111672 0.6481838 46.75 Yes  No
## 4     4 response 0.51425833 0.15016833 0.24578739 0.7747457 46.75 Yes Yes
## 5     5 response 0.17283618 0.04243829 0.10456819 0.2721284 72.00  No  No
## 6     6 response 0.29554942 0.05745944 0.19631342 0.4188083 72.00  No Yes
## 7     7 response 0.51624519 0.16335833 0.22845505 0.7936490 72.00 Yes  No
## 8     8 response 0.68180517 0.13170056 0.39466548 0.8756540 72.00 Yes Yes
```

Figure 6: ICU predictions (probability of dying)

```
incomes

## # A tibble: 14 x 3
##    year  income counts
##    <fct> <ord>   <dbl>
##  1 1960  0-3        65
##  2 1960  3-5        82
##  3 1960  5-7       113
##  4 1960  7-10      235
##  5 1960  10-12     156
##  6 1960  12-15     127
##  7 1960  15+       222
##  8 1970  0-3        43
##  9 1970  3-5        60
## 10 1970  5-7        77
## 11 1970  7-10      132
## 12 1970  10-12     105
## 13 1970  12-15     163
## 14 1970  15+       421
```

Figure 7: Income data

```
tidy(income.1)

##
## Re-fitting to get Hessian

## # A tibble: 7 x 5
##   term        estimate std.error statistic coef.type
##   <chr>          <dbl>     <dbl>     <dbl> <chr>
## 1 year1970      0.795     0.0811     9.81  coefficient
## 2 0-3|3-5      -2.54      0.104    -24.5   scale
## 3 3-5|5-7      -1.62      0.0749   -21.6   scale
## 4 5-7|7-10     -0.928     0.0635   -14.6   scale
## 5 7-10|10-12   -0.0295    0.0586    -0.503 scale
## 6 10-12|12-15   0.523     0.0602     8.69  scale
## 7 12-15|15+     1.17      0.0650    17.9   scale

drop1(income.1, test="Chisq")

## Single term deletions
##
## Model:
## income ~ year
##        Df    AIC    LRT  Pr(>Chi)
## <none>    7081.2
## year    1 7176.9 97.762 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 8: Income data: fitted model

```
##
## Re-fitting to get Hessian

## # A tibble: 7 x 3
##   income `1960` `1970`
##   <chr>   <dbl>  <dbl>
## 1 0-3    0.0732 0.0344
## 2 3-5    0.0925 0.0479
## 3 5-7    0.118  0.0691
## 4 7-10   0.209  0.153
## 5 10-12  0.135  0.128
## 6 12-15  0.135  0.159
## 7 15+    0.238  0.408
```

Figure 9: Income data: predictions

```
## # A tibble: 100 x 8
##    trt   status        enrolled.m enrolled.d enrolled.y last_follow.m last_follow.d last_follow.y
##    <chr> <chr>         <chr>           <dbl>      <dbl> <chr>                 <dbl>         <dbl>
##  1 C     recurrence    Apr                30       2018 Feb                       2          2019
##  2 B     recurrence    Nov                18       2018 Apr                      20          2019
##  3 D     recurrence    Oct                27       2018 Feb                       3          2019
##  4 C     no recurrence May                14       2019 Feb                      29          2020
##  5 B     recurrence    May                 9       2018 Mar                       5          2019
##  6 B     recurrence    Aug                 3       2019 Jan                      24          2020
##  7 A     recurrence    Mar                22       2019 Aug                      19          2019
##  8 C     recurrence    Jul                17       2019 Sep                      27          2019
##  9 D     recurrence    Feb                 1       2019 May                       3          2019
## 10 C     recurrence    Aug                23       2018 Jan                      22          2019
## 11 B     recurrence    May                21       2018 Mar                      17          2019
## 12 B     recurrence    Jan                12       2019 Apr                      15          2019
## 13 D     recurrence    Sep                25       2019 Jan                       8          2020
## 14 D     recurrence    Oct                29       2018 Aug                       2          2019
## 15 D     recurrence    Sep                13       2018 Feb                      12          2019
## 16 B     recurrence    Jun                 1       2019 Aug                      10          2019
## 17 A     recurrence    Jul                24       2018 May                       5          2019
## 18 B     recurrence    Jan                25       2019 Jun                      28          2019
## 19 D     recurrence    Aug                 2       2018 May                      29          2019
## 20 A     recurrence    Sep                 6       2018 Feb                      17          2019
## 21 D     recurrence    Nov                10       2018 Aug                      28          2019
## 22 B     recurrence    Dec                10       2018 Sep                      20          2019
## 23 C     recurrence    Feb                 5       2019 Feb                       9          2019
## 24 B     recurrence    Sep                 2       2019 Jan                      27          2020
## 25 B     recurrence    Mar                11       2019 Jan                       5          2020
## 26 C     recurrence    Apr                26       2019 Feb                      20          2020
## 27 B     recurrence    Aug                27       2019 Feb                       2          2020
## 28 D     no recurrence May                28       2019 Feb                      29          2020
## 29 B     recurrence    Sep                 4       2019 Feb                       3          2020
## 30 C     recurrence    Mar                31       2019 Jan                      21          2020
## 31 D     recurrence    Mar                26       2019 Sep                       5          2019
## 32 C     recurrence    Mar                27       2019 Jan                      12          2020
## 33 D     recurrence    Apr                18       2019 Feb                      12          2020
## 34 A     recurrence    Nov                22       2018 Mar                      11          2019
## 35 B     recurrence    Dec                26       2018 Oct                      14          2019
## 36 B     recurrence    Sep                17       2018 Jan                       3          2019
## 37 B     no recurrence Sep                23       2019 Feb                      29          2020
## 38 D     recurrence    Aug                13       2018 Jan                      14          2019
## 39 A     no recurrence May                 5       2019 Feb                      29          2020
## 40 C     recurrence    Nov                20       2018 Sep                       9          2019
## # ... with 60 more rows
```

Figure 10: Disease data (some)

```
disease1

## # A tibble: 100 x 4
##    trt   status        enrolled   last_follow
##    <chr> <chr>         <date>     <date>
##  1 C     recurrence    2018-04-30 2019-02-02
##  2 B     recurrence    2018-11-18 2019-04-20
##  3 D     recurrence    2018-10-27 2019-02-03
##  4 C     no recurrence 2019-05-14 2020-02-29
##  5 B     recurrence    2018-05-09 2019-03-05
##  6 B     recurrence    2019-08-03 2020-01-24
##  7 A     recurrence    2019-03-22 2019-08-19
##  8 C     recurrence    2019-07-17 2019-09-27
##  9 D     recurrence    2019-02-01 2019-05-03
## 10 C     recurrence    2018-08-23 2019-01-22
## # ... with 90 more rows
```

Figure 11: Disease data: tidied dates

```
disease

## # A tibble: 100 x 3
##    trt   status         days
##    <chr> <chr>         <dbl>
##  1 C     recurrence      278
##  2 B     recurrence      153
##  3 D     recurrence       99
##  4 C     no recurrence   291
##  5 B     recurrence      300
##  6 B     recurrence      174
##  7 A     recurrence      150
##  8 C     recurrence       72
##  9 D     recurrence       91
## 10 C     recurrence      152
## # ... with 90 more rows
```

Figure 12: Disease data ready for survival analysis

```r
disease.1 <- coxph(y~trt, data=disease)
summary(disease.1)

## Call:
## coxph(formula = y ~ trt, data = disease)
##
##   n= 100, number of events= 93
##
##         coef exp(coef) se(coef)      z Pr(>|z|)
## trtB  0.7318    2.0789   0.2958  2.474   0.0133 *
## trtC  0.5084    1.6626   0.3179  1.599   0.1098
## trtD -0.1544    0.8570   0.3071 -0.503   0.6152
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##      exp(coef) exp(-coef) lower .95 upper .95
## trtB     2.079     0.4810    1.1643     3.712
## trtC     1.663     0.6015    0.8917     3.100
## trtD     0.857     1.1669    0.4694     1.564
##
## Concordance= 0.63  (se = 0.031 )
## Likelihood ratio test= 12.06  on 3 df,   p=0.007
## Wald test            = 12.16  on 3 df,   p=0.007
## Score (logrank) test = 12.68  on 3 df,   p=0.005

drop1(disease.1, test="Chisq")

## Single term deletions
##
## Model:
## y ~ trt
##        Df    AIC    LRT Pr(>Chi)
## <none>    675.61
## trt     3 681.67 12.059 0.007183 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 13: Survival analysis and output

```
joint %>% print(n=30)

## # A tibble: 30 x 3
##    treatment sex     time
##    <chr>     <chr>  <dbl>
##  1 A         male      12
##  2 A         female    21
##  3 A         male      15
##  4 A         female    19
##  5 A         male      16
##  6 A         female    18
##  7 A         male      17
##  8 A         female    24
##  9 A         male      14
## 10 A         female    25
## 11 B         male      14
## 12 B         female    21
## 13 B         male      17
## 14 B         female    20
## 15 B         male      19
## 16 B         female    23
## 17 B         male      20
## 18 B         female    27
## 19 B         male      17
## 20 B         female    25
## 21 C         male      25
## 22 C         female    37
## 23 C         male      27
## 24 C         female    34
## 25 C         male      29
## 26 C         female    36
## 27 C         male      24
## 28 C         female    26
## 29 C         male      22
## 30 C         female    29
```

Figure 14: Pain relief times data

```
joint.1 <- aov(time~treatment*sex, data=joint)
summary(joint.1)

##               Df Sum Sq Mean Sq F value  Pr(>F)
## treatment      2  651.5   325.7   34.84 8.0e-08 ***
## sex            1  313.6   313.6   33.54 5.7e-06 ***
## treatment:sex  2    1.9     0.9    0.10   0.905
## Residuals     24  224.4     9.4
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

joint.2 <- update(joint.1, .~.-treatment:sex)
summary(joint.2)

##             Df Sum Sq Mean Sq F value   Pr(>F)
## treatment    2  651.5   325.7   37.43 2.22e-08 ***
## sex          1  313.6   313.6   36.04 2.44e-06 ***
## Residuals   26  226.3     8.7
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 15: Pain relief analyses of variance

```
joint %>% filter(sex=="female") -> females
females.1 <- aov(time~treatment, data=females)
summary(females.1)

##             Df Sum Sq Mean Sq F value   Pr(>F)
## treatment    2  348.1  174.07   13.12 0.000955 ***
## Residuals   12  159.2   13.27
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

TukeyHSD(females.1)

##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = time ~ treatment, data = females)
##
## $treatment
##     diff       lwr       upr     p adj
## B-A  1.8 -4.345745  7.945745 0.7210392
## C-A 11.0  4.854255 17.145745 0.0012146
## C-B  9.2  3.054255 15.345745 0.0046858

joint %>% filter(sex=="male") -> males
males.1 <- aov(time~treatment, data=males)
summary(males.1)

##             Df Sum Sq Mean Sq F value   Pr(>F)
## treatment    2  305.2  152.60   28.09 2.97e-05 ***
## Residuals   12   65.2    5.43
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

TukeyHSD(males.1)

##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = time ~ treatment, data = males)
##
## $treatment
##     diff       lwr       upr     p adj
## B-A  2.6 -1.333026  6.533026 0.2229758
## C-A 10.6  6.666974 14.533026 0.0000304
## C-B  8.0  4.066974 11.933026 0.0004165
```

Figure 16: Pain relief further analysis 1

```
TukeyHSD(joint.2)

##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = time ~ treatment + sex, data = joint)
##
## $treatment
##     diff       lwr       upr     p adj
## B-A  2.2 -1.078283  5.478283 0.2365089
## C-A 10.8  7.521717 14.078283 0.0000000
## C-B  8.6  5.321717 11.878283 0.0000019
##
## $sex
##                  diff       lwr       upr   p adj
## male-female -6.466667 -8.680866 -4.252467 2.4e-06
```

Figure 17: Pain relief further analysis 2

```
## # A tibble: 36 x 4
##    subject method familiar program
##    <chr>   <chr>     <dbl>   <dbl>
##  1 a1      a            14      29
##  2 a2      a            10      24
##  3 a3      a             7      14
##  4 a4      a            18      27
##  5 a5      a            14      27
##  6 a6      a            16      28
##  7 a7      a            13      27
##  8 a8      a            15      32
##  9 a9      a             5      13
## 10 a10     a            18      35
## 11 a11     a            16      32
## 12 a12     a            10      17
## 13 b1      b             6      15
## 14 b2      b            16      28
## 15 b3      b             9      13
## 16 b4      b            19      36
## 17 b5      b            13      29
## 18 b6      b            14      27
## 19 b7      b            15      31
## 20 b8      b            18      33
## 21 b9      b            17      32
## 22 b10     b             8      15
## 23 b11     b            15      30
## 24 b12     b            16      26
## 25 c1      c            15      32
## 26 c2      c             9      27
## 27 c3      c             7      15
## 28 c4      c            12      23
## 29 c5      c            12      26
## 30 c6      c             9      17
## 31 c7      c            12      25
## 32 c8      c             3      14
## 33 c9      c            13      29
## 34 c10     c            10      22
## 35 c11     c            11      30
## 36 c12     c             8      25
```

Figure 18: Programming data (in data frame `prog`)

```
prog.1 <- lm(program~familiar*method, data=prog)
anova(prog.1)

## Analysis of Variance Table
##
## Response: program
##                 Df  Sum Sq Mean Sq  F value    Pr(>F)
## familiar         1 1237.72 1237.72 120.5920 4.949e-12 ***
## method           2   74.50   37.25   3.6292   0.03877 *
## familiar:method  2    6.18    3.09   0.3010   0.74229
## Residuals       30  307.91   10.26
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 19: Programming study ANCOVA 1

```
prog.2 <- lm(program~familiar+method, data=prog)
anova(prog.2)

## Analysis of Variance Table
##
## Response: program
##           Df  Sum Sq Mean Sq F value    Pr(>F)
## familiar   1 1237.72 1237.72 126.101 1.234e-12 ***
## method     2   74.50   37.25   3.795   0.03319 *
## Residuals 32  314.09    9.82
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 20: Programming study ANCOVA 2

```
prog %>% group_by(method) %>%
  summarize(n=n(), mean_program=mean(program), sd_program=sd(program),
            mean_familiar=mean(familiar), sd_familiar=sd(familiar))

## # A tibble: 3 x 6
##   method     n mean_program sd_program mean_familiar sd_familiar
##   <chr>  <int>        <dbl>      <dbl>         <dbl>       <dbl>
## 1 a         12         25.4       7.15          13          4.18
## 2 b         12         26.2       7.69          13.8        4.11
## 3 c         12         23.8       5.83          10.1        3.18

prog.3 <- lm(program~method, data=prog)
anova(prog.3)

## Analysis of Variance Table
##
## Response: program
##           Df  Sum Sq Mean Sq F value Pr(>F)
## method     2   38.89  19.444  0.4042 0.6708
## Residuals 33 1587.42  48.104
```

Figure 21: Programming study: further analysis

```
trts <- c("D", "C", "B", "A")
new <- tibble(trt=trts)
new

## # A tibble: 4 x 1
##   trt
##   <chr>
## 1 D
## 2 C
## 3 B
## 4 A

s <- do.call(survfit, list(formula=disease.1, newdata=new, data=disease))
ggsurvplot(s, conf.int=F)
```

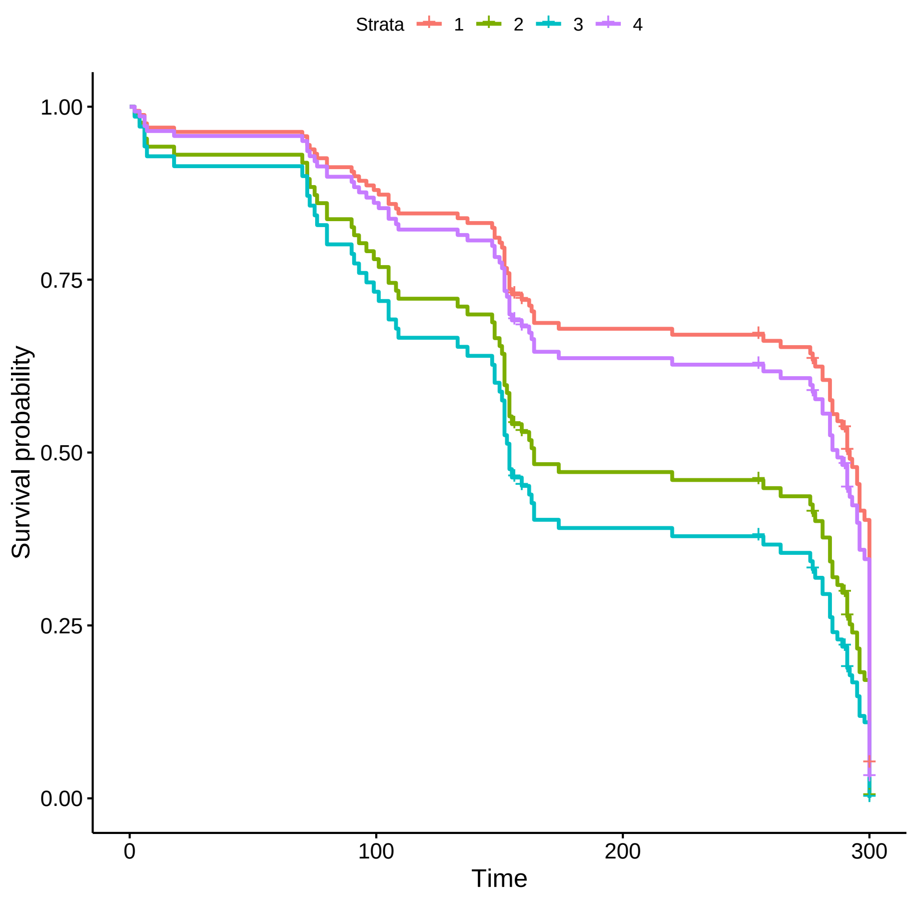

Figure 22: Disease data survival curve plot

```
## Warning:  `fun.y` is deprecated.  Use `fun` instead.
## `fun.y` is deprecated.  Use `fun` instead.
```

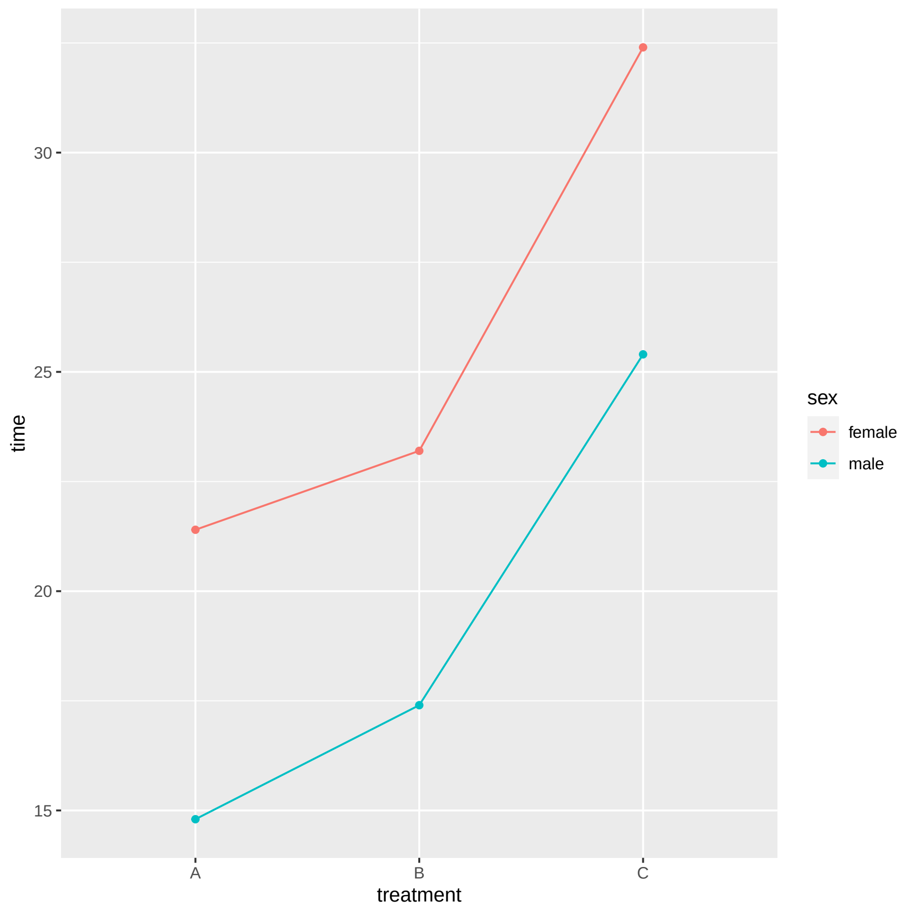

Figure 23: Pain relief interaction plot

```
## `geom_smooth()` using formula 'y ~ x'
```

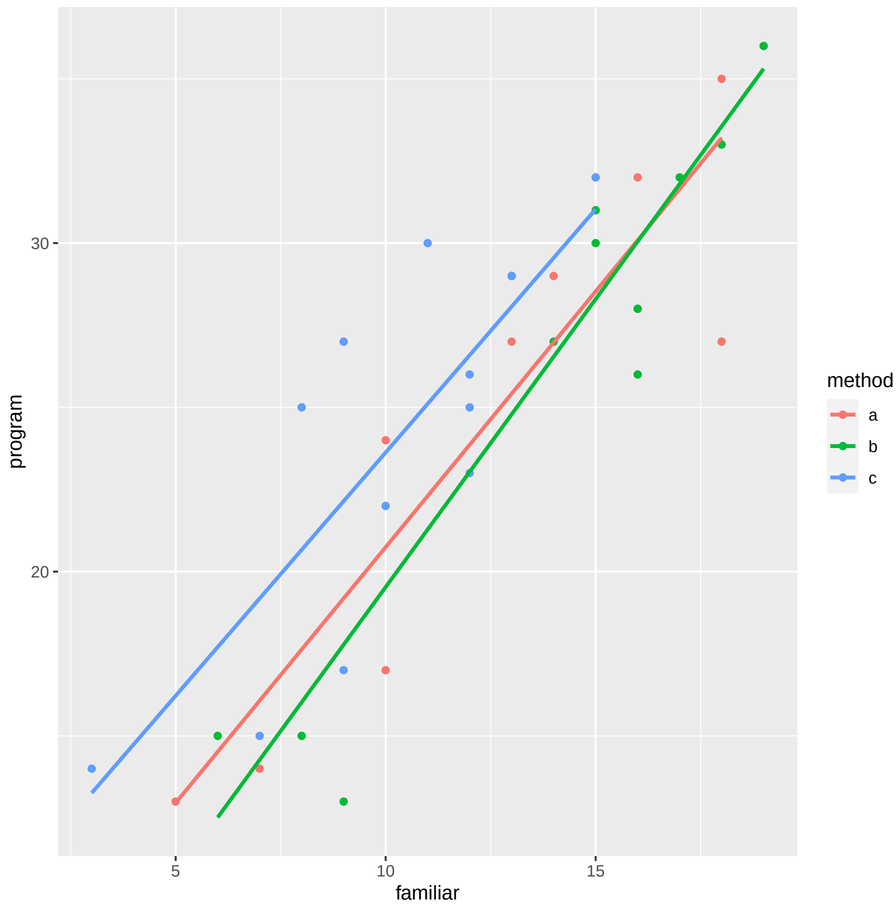

Figure 24: Programming study graph