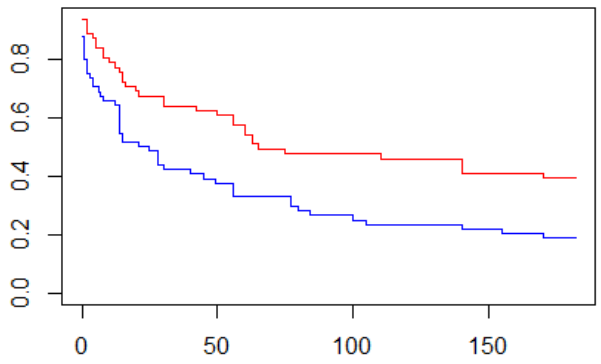
hw8

1)

a. use the method for the log-rank test to find the sample size needed. Assume that all patients who have not relapsed by 180 days will be censored at that point, and none will be censored before that point.

We need 456 people

```
library(asaur)
library(survival)
LogRankDeaths <- function(Delta, p, alpha, pwr) {
  z.alpha <- qnorm(alpha, lower.tail = FALSE)
  z.beta <- qnorm(1 - pwr, lower.tail = FALSE)
  num <- (z.alpha + z.beta) ^ 2
  denom <- p*(1-p)*(log(Delta)) ^ 2
  dd <- num / denom
  dd
}

LogRankDeaths(0.7, 0.5, 0.05, 0.9)

## [1] 269.2674

fit_survfit  = survfit(Surv(ttr, relapse) ~ grp, data = pharmacoSmoking)
summary(fit_survfit)

plot(fit_survfit, col = c("red","blue"))
```



```
table(pharmacoSmoking$grp)

##
## combination   patchOnly
##          61          64

25/61

## [1] 0.4098361

0.409^(1/0.7)*270 + 0.409*270

## [1] 185.7105

186+270

## [1] 456
```

b. use simulations to find the sample size for a Cox proportional hazards model that includes group and age (linear) as covariates with a hazard ratio of 0.7 for group and -0.02 for age. Patients will be accrued to the study uniformly over 90 days, then followed-up for an additional 90 days, there will be no other censoring other than the study ending at 180 days.

Need 99 patients in this setup.

```
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

library(survival)
smoke1 <- pharmacoSmoking %>%
  filter(grp == "combination") %>% filter(!(ttr < 180 & relapse == 0))

smoke1$censor = ifelse(smoke1$ttr >= 180, 1, 0)

survreg(Surv(ttr+0.001, relapse)~1,
        data=smoke1)

## Call:
## survreg(formula = Surv(ttr + 0.001, relapse) ~ 1, data = smoke1)
##
## Coefficients:
```

```
## (Intercept)
##    5.440526
##
## Scale= 2.51874
##
## Loglik(model)= -197.5   Loglik(intercept only)= -197.5
## n= 61
```

```
simfun3 <- function(n=100, beta_t= log(0.7),
                    beta_a=-0.02,
                    beta0=log(5.27), scale=1.93,
                    accrual=90, followup=90) {
  age <- sample(41:56, n, replace=TRUE)
  treat <- rep(0:1, each=n/2)
  eta <- exp(beta0 + beta_t*treat +
               beta_a*(age-41))

  time <- rweibull(n, scale, eta)
  cens <-  followup
  tmpdat = data.frame(time = pmin(time,cens),
             status=as.numeric(time < cens),
             treat=treat,
             age=age)
  fit1 = coxph(Surv(time,status) ~ treat+age,
              data=tmpdat)
  summary(fit1)$coef[1,5]
}


out2 = mean(replicate(1000, simfun3(n=100)) < 0.05)
out2
```

```
## [1] 0.928
```

2. Fit a Poisson approximation model/piecewise constant hazard model to the pharmacoSmoking data using only group as a predictor/covariate and using time intervals for the constant hazards of 0-15, 15-40, and 40-182 days.

```
phsmoke <- survSplit( Surv(I(ttr+0.5), relapse)~grp, data=pharmacoSmoking,
                         cut=c(15,40),
                         id="id", episode="timeblock")

phsmoke$exp <- phsmoke$tstop - phsmoke$tstart
phsmoke$timeblock = as.factor(phsmoke$timeblock)

fit.pois <- glm(relapse ~ 0 + factor(timeblock)  +
                  offset(log(exp)) +
                   grp, data=phsmoke,
                family=poisson)

summary(fit.pois)
```

```
##
## Call:
## glm(formula = relapse ~ 0 + factor(timeblock) + offset(log(exp)) +
##     grp, family = poisson, data = phsmoke)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3659  -0.9928  -0.7631   1.0837   3.0943
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## factor(timeblock)1  -3.8434     0.2024 -18.988  < 2e-16 ***
## factor(timeblock)2  -5.0911     0.2871 -17.731  < 2e-16 ***
## factor(timeblock)3  -5.6670     0.2150 -26.363  < 2e-16 ***
## grppatchOnly         0.6383     0.2160   2.955  0.00312 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 18974.59  on 272  degrees of freedom
## Residual deviance:   432.19  on 268  degrees of freedom
## AIC: 618.19
##
## Number of Fisher Scoring iterations: 7
```

3)

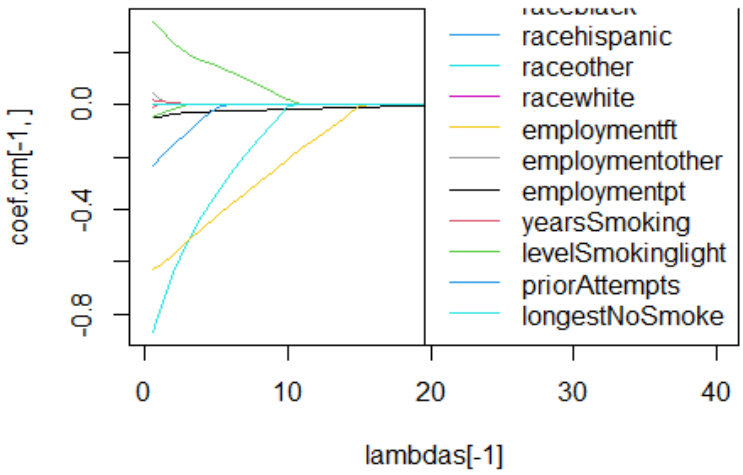
Fit a predictive model to the pharmacoSmoking data using all of the covariates other than ageGroup2 and ageGroup4 (only linear terms for the numerical covariates). Use the lasso with cross validation to find a simpler predictive model. Show how it compares in prediction to the full model and explain your choice of the penalty.

The best lambda is 3.

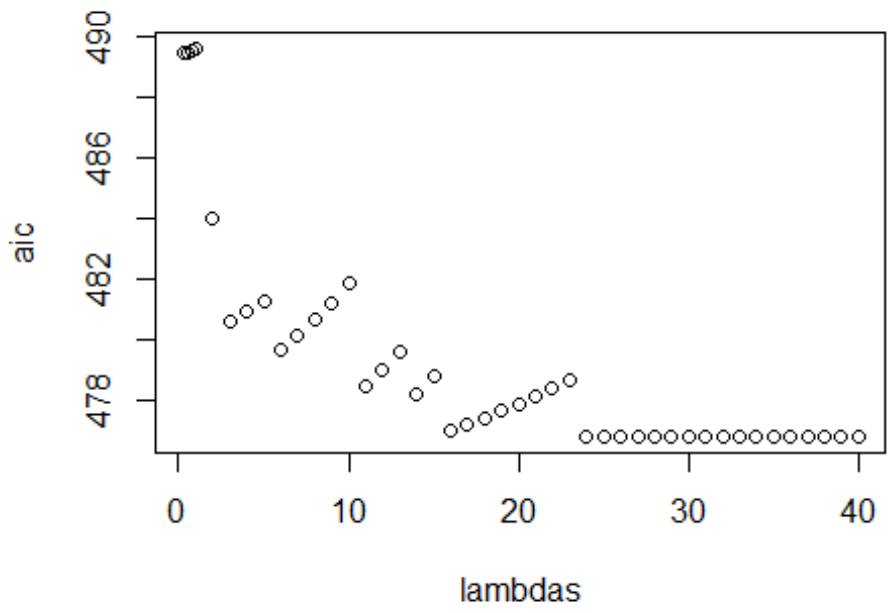
This model does better than the other

```
library(survival)


fit_3 = coxph(Surv(ttr, relapse) ~ age + gender + race + employment +
yearsSmoking + levelSmoking + priorAttempts + longestNoSmoke, data =
pharmacoSmoking)
```

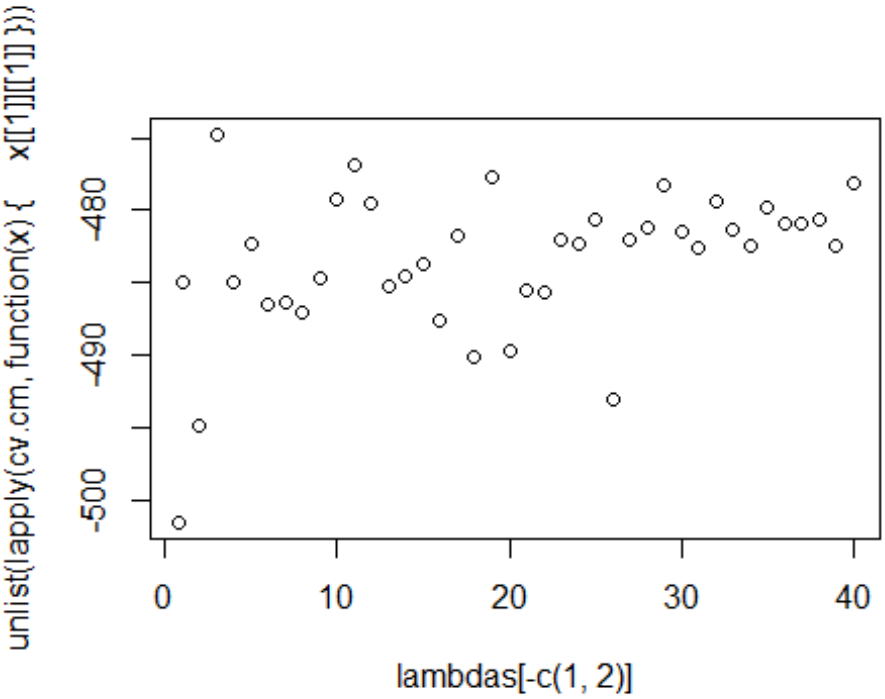
```
coef.cm <- sapply(lfit.cm, coef, which='all') |>
  t()

matplot(lambdas[-1], coef.cm[-1,], type='l', lty=1, col=1:8)
legend('bottomright', legend=colnames(coef.cm), lty=1, col=1:8)
```




```
approx_n <- sapply(lfit.cm, function(x) sum(coef(x)!=0))
ll <- sapply(lfit.cm, loglik)
aic <- -ll + 2*approx_n

plot(lambdas, aic)
```




```
set.seed(1)
folds <- sample(rep(1:10, 50))

cv.cm <- lapply(lambdas[-(1:2)], function(lambda) {
  cvl(Surv(ttr, relapse) ~ age + gender + race +
      employment + yearsSmoking + levelSmoking +
      priorAttempts + longestNoSmoke,
      data = pharmacoSmoking,
      lambda1 = lambda,
      fold = 10)
})

cv.cm[[1]][[1]]

## [1] -501.4706

lambdas[which.max(lapply(cv.cm, function(x){x[[1]][[1]]}))]

## [1] 1

plot(lambdas[-c(1,2)], unlist(lapply(cv.cm, function(x){x[[1]][[1]]})))
```




```
best_lasso = penalized(
    Surv(ttr, relapse) ~ age + gender + race +
      employment + yearsSmoking + levelSmoking +
      priorAttempts + longestNoSmoke,
    standardize=TRUE,
    data = pharmacoSmoking,
    lambda1 = 3
  )

#predict(best_lasso, newdata = pharmacoSmoking)


# pen <- penalized(
#         Surv(ttr, relapse), penalized = ~age + gender + race +
#       employment + yearsSmoking + levelSmoking +
#       priorAttempts + longestNoSmoke, lambda1 = 3,
#       unpenalized = Surv(ttr, relapse) ~ age + gender + race +
#       employment + yearsSmoking + levelSmoking +
#       priorAttempts + longestNoSmoke, data = pharmacoSmoking)
# preds = predict(pen, data = pharmacoSmoking)
# as.data.frame(preds) %>% View()
# preds_lasso = as.data.frame(preds)
# plot(preds[,6])
# plot(coxph(Surv(ttr, relapse) ~ age + gender + race +
#       employment + yearsSmoking + levelSmoking +
#       priorAttempts + longestNoSmoke, data = pharmacoSmoking))
# preds = predict(fit_3, newdata = pharmacoSmoking)
```

```
#
# plot(preds)
#
# predict(best_lasso, data = pharmacoSmoking)
#
# mean((c(preds_lasso[6,])*125 - pharmacoSmoking$ttr)^2) %>% sqrt()
# mean((preds*125 - pharmacoSmoking$ttr)^2) %>% sqrt()
```

4)

The file ChildMort2.csv Download has data on child mortality. The columns Age, Age.L, and Age.U have information on the interval censored age at death. Using the interval censored data, estimate the survival curve for the child mortality. Compare this curve to the standard Kaplan-Meier estimate using the exit and event columns in the same dataset.

```
#install.packages("BiocManager")

library(interval)

child = read.csv("ChildMort2.csv")
colnames(child)

## [1] "exit"  "event" "Age"   "Age.L" "Age.U"

interval_fit = icfit(Surv(Age.L, Age.U, type='interval2')~1,
             data=child)

plot(interval_fit, ylim = c(0.75,1))
lines(survfit(Surv(exit, event)~1, data = child), col = "red")
```