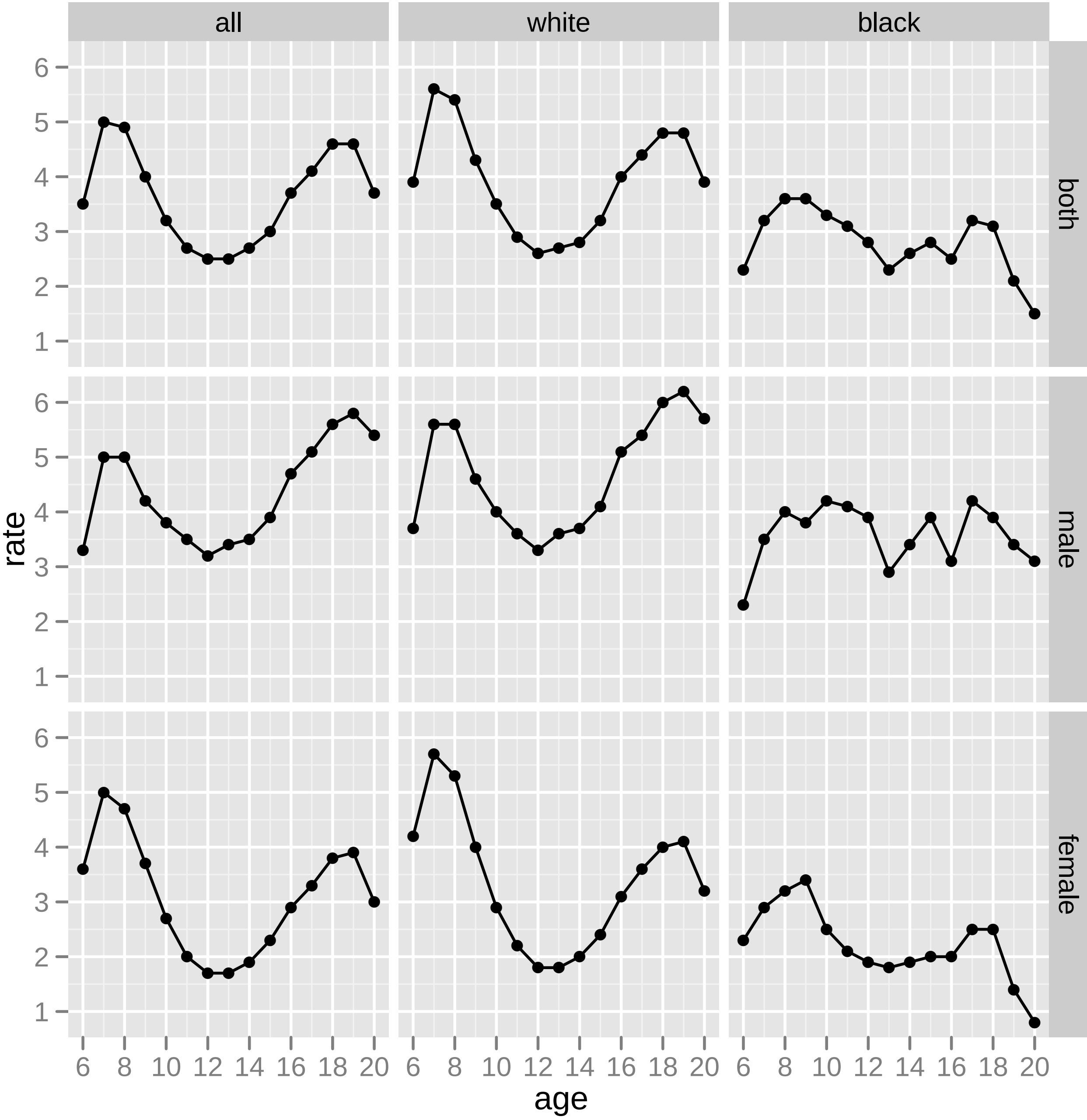
Read in data and add labels:

```
> cdat <- read.csv("cancer.dat", comment = "#", header = FALSE)
> names(cdat) <- c("age", "race", "sex", "rate")
> cdat <- transform(cdat, race = factor(race, labels = c("all",
     "white", "black")), sex = factor(sex, labels = c("both",
     "male", "female")))
```

Basic plot:

```
> library(ggplot2)
> g1 <- ggplot(cdat, aes(x = age, y = rate)) + geom_line() + geom_point() +
     facet_grid(sex ~ race)
> print(g1)
```



Function for 2-component normal mixture prediction:

```
> binormfun <- function(p, age) {
     with(as.list(p), r * (p * dnorm(age, mean = age1, sd = exp(logsd1)) +
```

```
        (1 - p) * dnorm(age, mean = age2, sd = exp(logsd2))))
 }
```

```
> binormfun2 <- function(p, age) {
     with(as.list(p), cbind(r * (p * dnorm(age, mean = age1, sd = exp(logsd1))),
         r * (1 - p) * dnorm(age, mean = age2, sd = exp(logsd2))))
 }
```

Objective function for sum of squares:

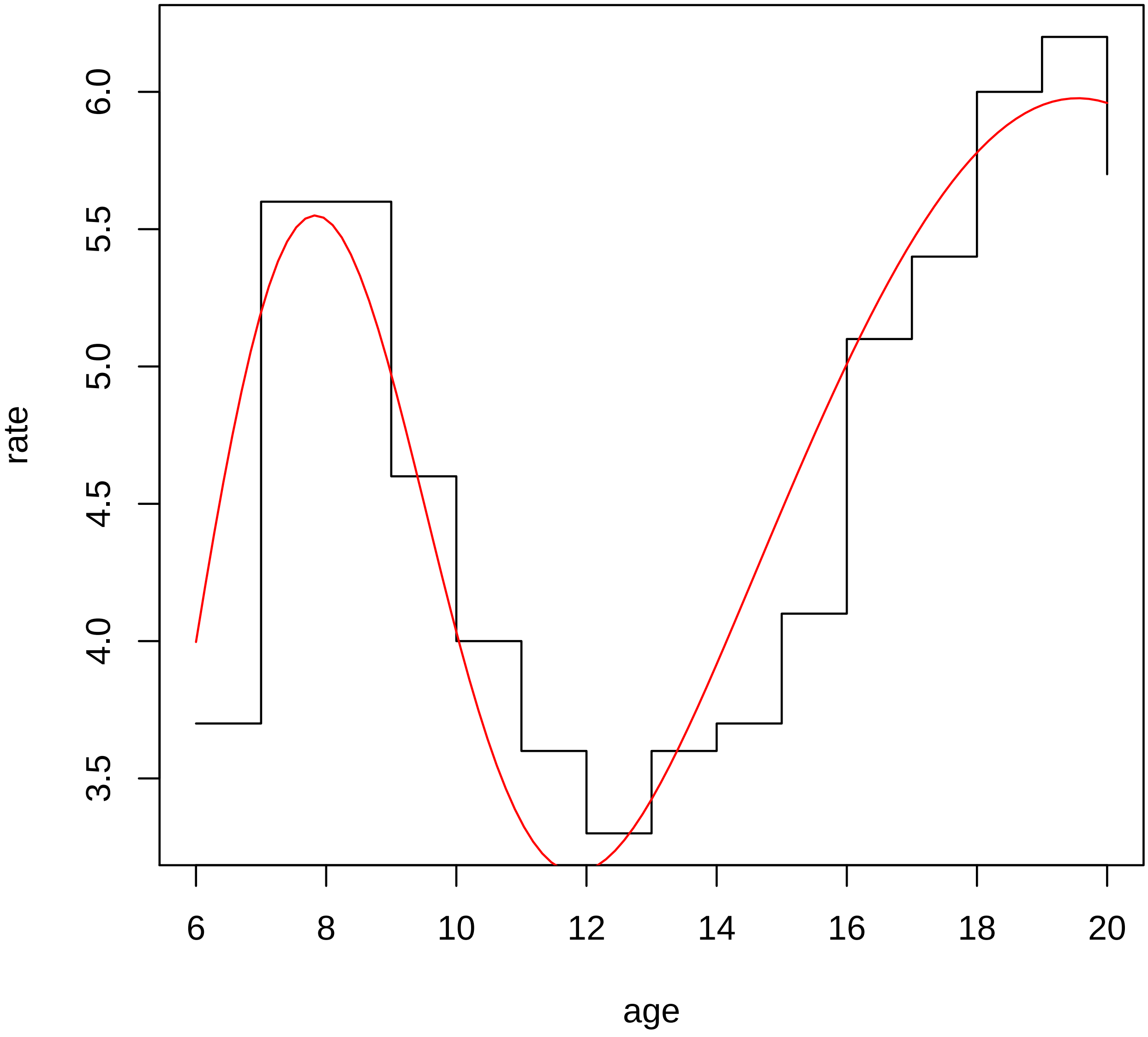
```
> ssqfun <- function(p, age, obsrate) {
     sum((obsrate - binormfun(p, age))^2)
 }
```

Try it out on the "white male" data:

```
> cdat.whitemale <- subset(cdat, sex == "male" & race == "white")
> opt1 <- optim(fn = ssqfun, par = c(age1 = 8, age2 = 18, p = 0.4,
     r = 3, logsd1 = 0, logsd2 = 0), age = cdat.whitemale$age,
     obsrate = cdat.whitemale$rate, method = "L-BFGS-B", lower = c(age1 = 5,
         age2 = 5, p = 0.01, r = 0.1, logsd1 = -2, logsd2 = -2),
     upper = c(age1 = 21, age2 = 21, p = 0.99, r = 1000, logsd1 = 2,
         logsd2 = 2))
```

Plot:

```
> with(cdat.whitemale, {
     plot(age, rate, type = "s")
 })
> agevec <- seq(6, 20, length = 101)
> lines(agevec, binormfun(opt1$par, agevec), col = 2)
```



Fit all the categories:

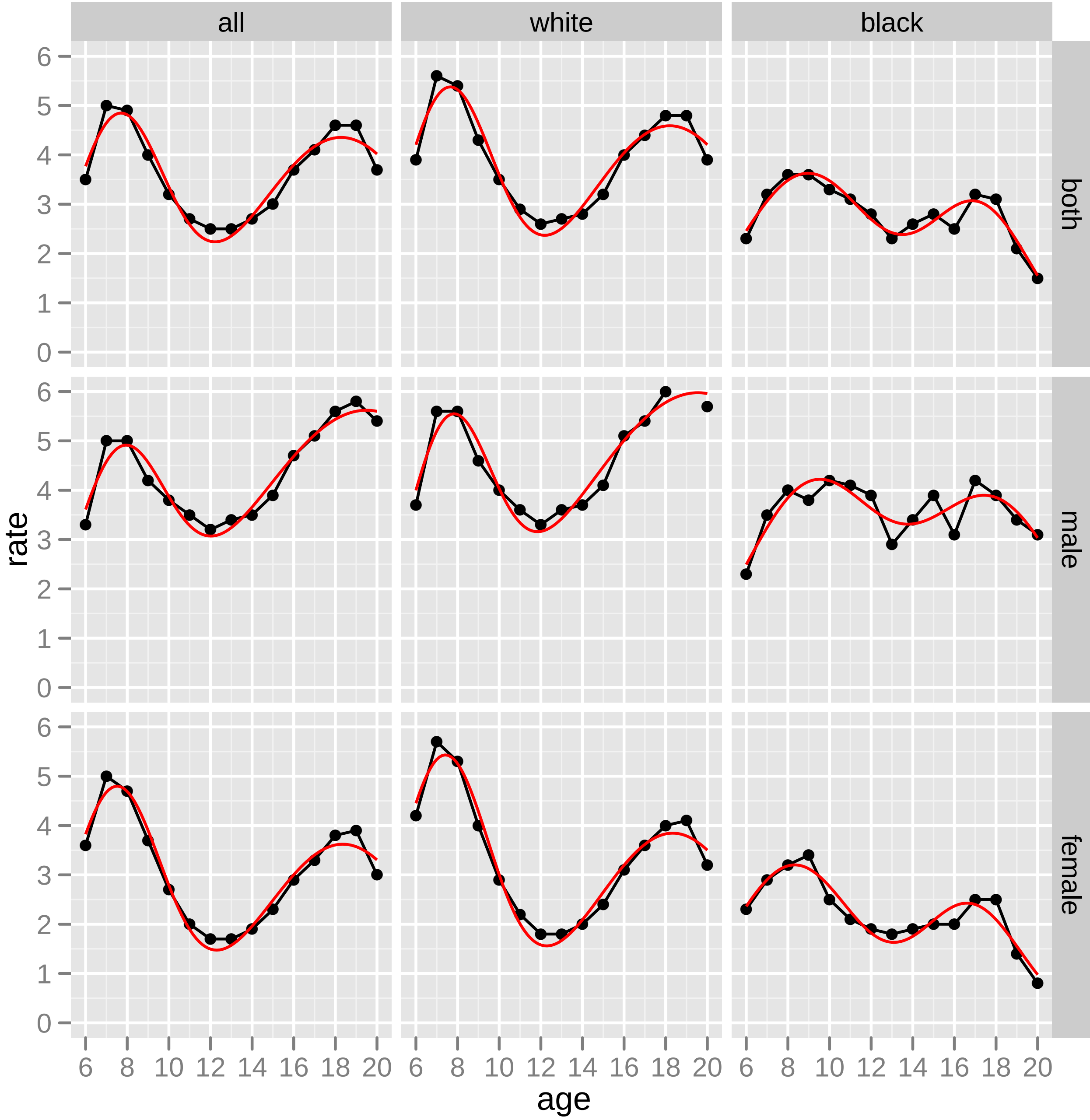
```
> fitvals <- ddply(cdat, c("race", "sex"), function(d) {
     optd <- optim(fn = ssqfun, par = c(age1 = 8, age2 = 18, p = 0.4,
```

```
         r = 3, logsd1 = 0, logsd2 = 0), age = d$age, obsrate = d$rate,
         method = "L-BFGS-B", lower = c(age1 = 5, age2 = 5, p = 0.01,
             r = 0.1, logsd1 = -1, logsd2 = -1), upper = c(age1 = 21,
             age2 = 21, p = 0.99, r = 1000, logsd1 = 2, logsd2 = 2))
     data.frame(matrix(optd$par, nrow = 1))
 })
> names(fitvals) <- c("race", "sex", names(opt1$par))
```

Predictions:

```
> predvals <- ddply(fitvals, c("race", "sex"), function(d) {
     data.frame(race = d$race, sex = d$sex, age = agevec, rate = binormfun(unlist(d)[-(1:2)]
         agevec))
 })
```

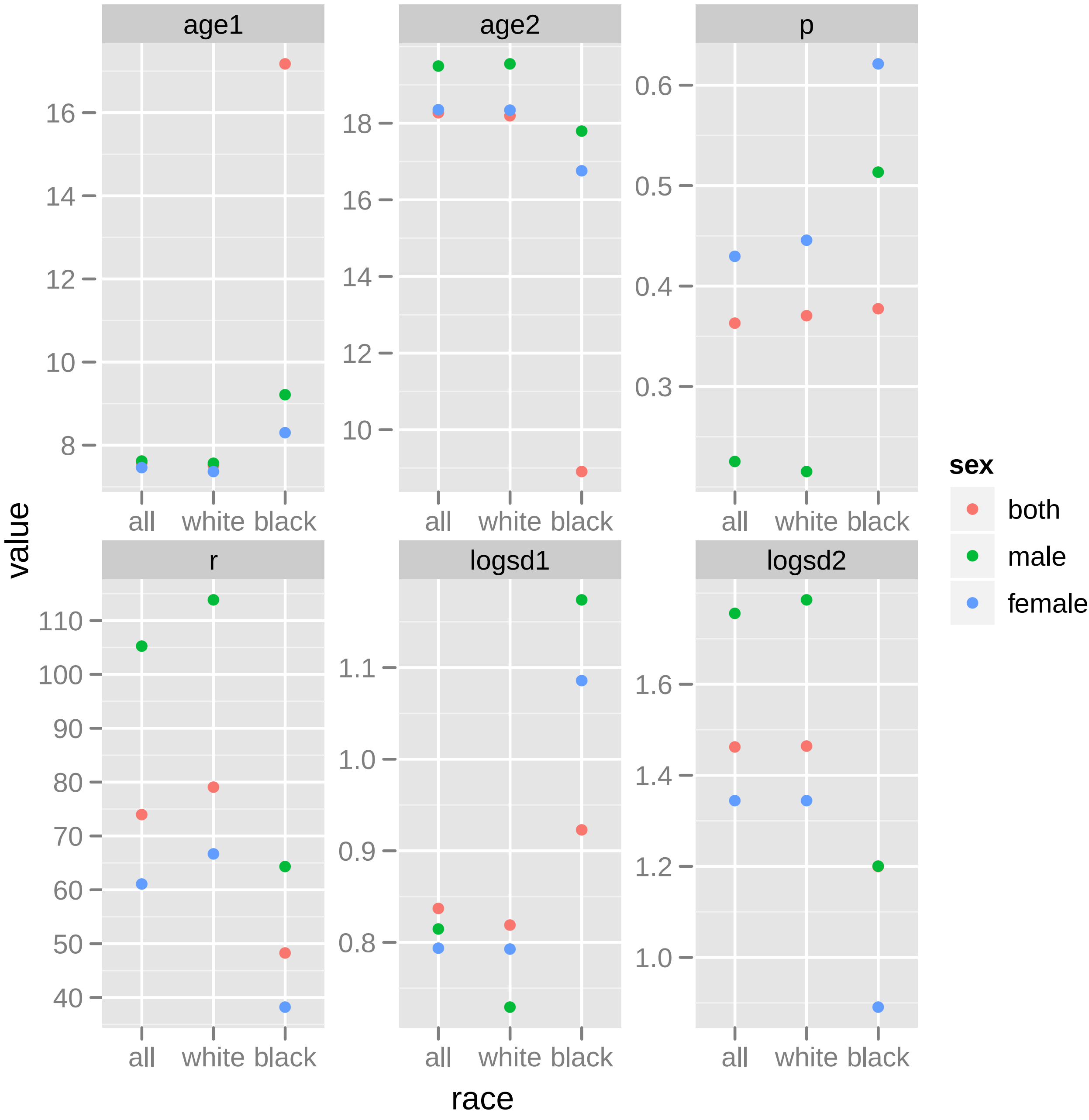
```
> print(g1 + geom_line(data = predvals, colour = "red") + ylim(0,
     6))
```



(Black/both, male/all, and white/female were bogus. with lower bound of -2 on logsd; it all looks pretty good with lower bound of -1 on logsd)

Note that there is some ambiguity in the definitions: in particular, the order of the components is reversed in the black/both case (we should swap age1/age2, logsd1/logsd2, make $p = 1 - p$)

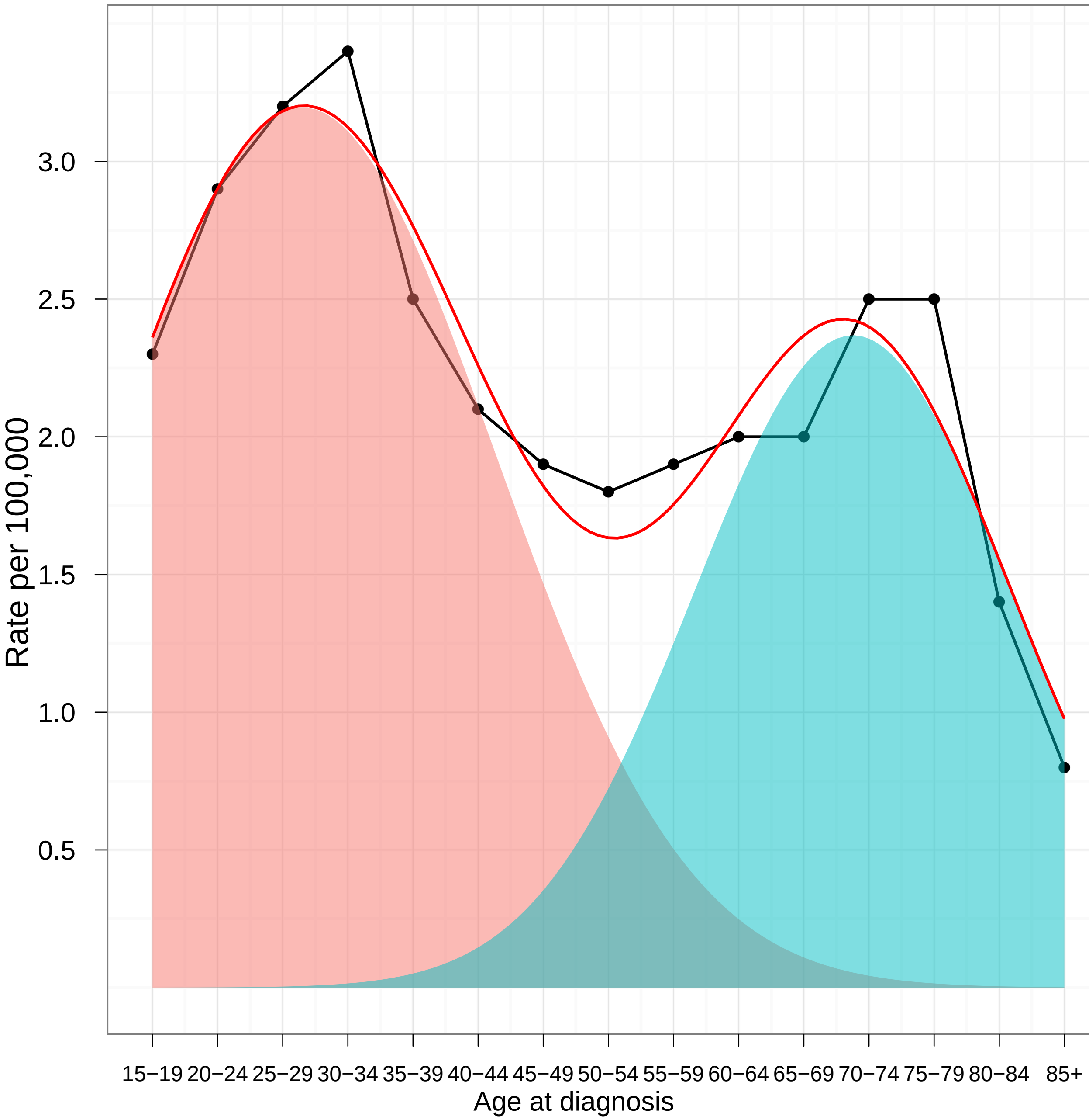
```
> fitvals

   race    sex      age1      age2         p         r    logsd1    logsd2
1   all   both  7.561498 18.271628 0.3631618  73.96827 0.8371759 1.4623695
2   all   male  7.622772 19.496572 0.2254660 105.26870 0.8149234 1.7555329
3   all female  7.462562 18.354990 0.4297399  61.06521 0.7937880 1.3442427
4 white   both  7.518462 18.194473 0.3705919  79.05838 0.8191183 1.4643501
5 white   male  7.569612 19.546954 0.2154487 113.86495 0.7292121 1.7855559
6 white female  7.369441 18.337651 0.4455924  66.70573 0.7928542 1.3444033
7 black   both 17.173957  8.912847 0.3773506  48.27721 0.9230265 1.1995284
8 black   male  9.213683 17.798859 0.5134767  64.31589 1.1740598 1.2003312
9 black female  8.305486 16.753111 0.6212328  38.19581 1.0857327 0.8905411

> print(ggplot(melt(fitvals, id.var = 1:2), aes(x = race, y = value,
     colour = sex)) + geom_point() + facet_wrap(~variable, scale = "free"))
```



Now plot overlapping histograms:

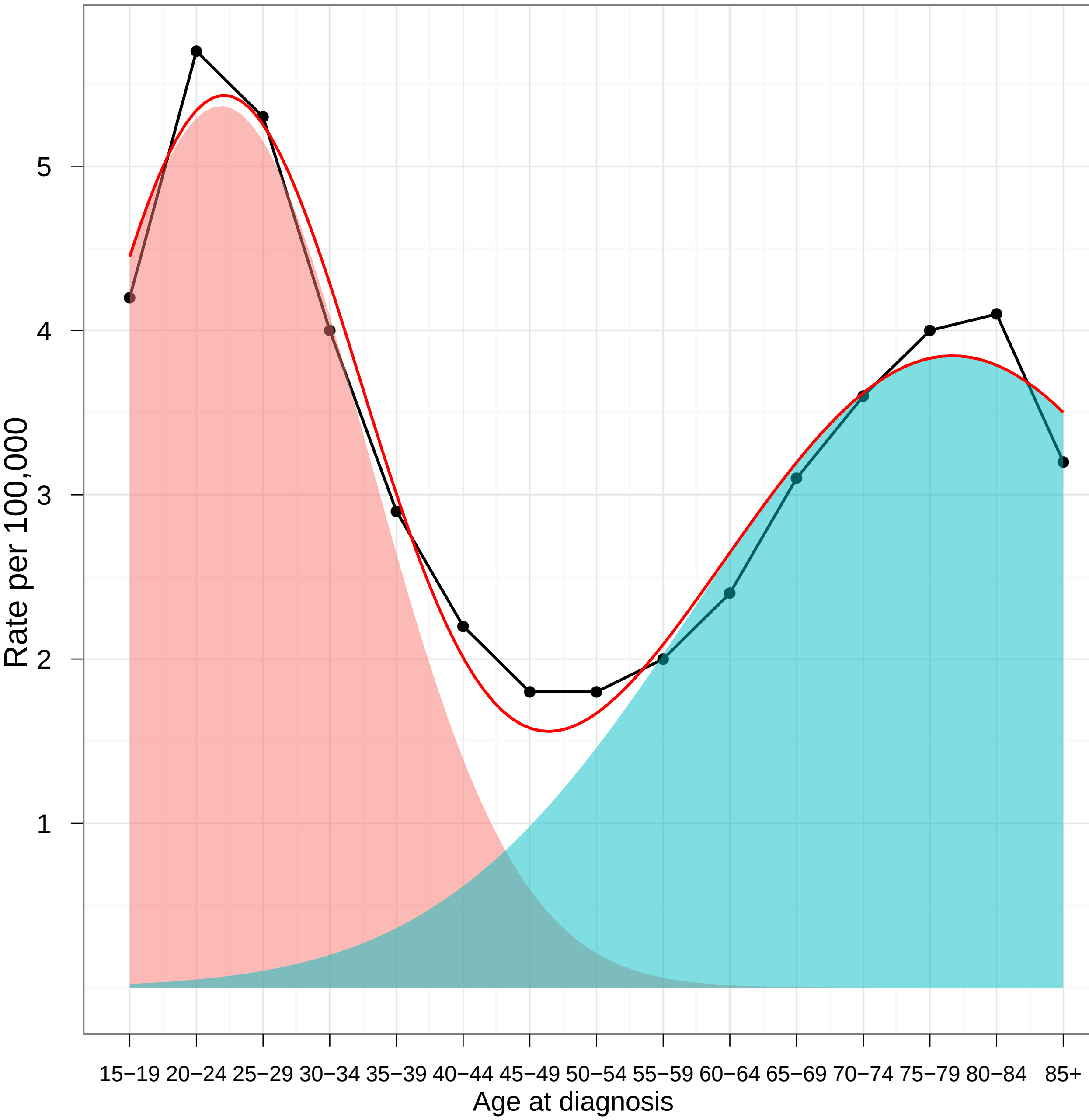
```
> predvals2 <- melt(ddply(fitvals, c("race", "sex"), function(d) {
    data.frame(race = d$race, sex = d$sex, age = agevec, rate = binormfun2(unlist(d)[-(1:2
```

```
        agevec))
 }), id.vars = 1:3)

> labs <- c(paste(seq(15, 80, by = 5), seq(19, 84, by = 5), sep = "-"),
     "85+")
> names(labs) <- unique(cdat$age)

> print(g1 + geom_ribbon(data = predvals2, aes(y = NULL, fill = variable,
     ymax = value), ymin = 0, alpha = 0.5) + theme_bw() + geom_line(data = predvals,
     colour = "red") + ylim(0, 6))

> ss <- function(x) subset(x, race == "black" & sex == "female")
> print(g1 %+% ss(cdat) + geom_ribbon(data = ss(predvals2), aes(y = NULL,
     fill = variable, ymax = value), ymin = 0, alpha = 0.5) +
     theme_bw() + geom_line(data = ss(predvals), colour = "red") +
     opts(legend.position = "none", axis.text.x = theme_text(size = 8),
         axis.title.x = theme_text(vjust = 0.5)) + labs(x = "Age at diagnosis",
     y = "Rate per 100,000") + scale_x_continuous(breaks = 6:20,
     labels = labs))
```



```
> ss <- function(x) subset(x, race == "white" & sex == "female")
> print(g1 %+% ss(cdat) + geom_ribbon(data = ss(predvals2), aes(y = NULL,
      fill = variable, ymax = value), ymin = 0, alpha = 0.5) +
```

```
theme_bw() + geom_line(data = ss(predvals), colour = "red") +
opts(legend.position = "none", axis.text.x = theme_text(size = 8),
    axis.title.x = theme_text(vjust = 0.5)) + labs(x = "Age at diagnosis",
y = "Rate per 100,000") + scale_x_continuous(breaks = 6:20,
labels = labs))
```


Rate per 100,000
5
4
3
2
1
15−19
20−24
25−29
30−34
35−39
40−44
45−49
50−54
55−59
60−64
65−69
70−74
75−79
80−84
85+
Age at diagnosis