Modeling in R

Nicholas Horton (nhorton@amherst.edu)

June 23, 2016

This document describes ways to fit a variety of models using the `mosaic` package. See https://github.com/ProjectMOSAIC/LittleBooks/blob/master/README.md for a link to the `Student Guide to R` that provides more details about linear regression modeling.

```
options(digits=3)
require(mosaic)
require(NHANES)
```

```
favstats(~ female, data=HELPrct)
```

```
##  min Q1 median Q3 max  mean    sd   n missing
##    0  0      0  0   1 0.236 0.425 453       0
```

```
tally(~ sex, data=HELPrct)
```

```
##
## female   male
##    107    346
```

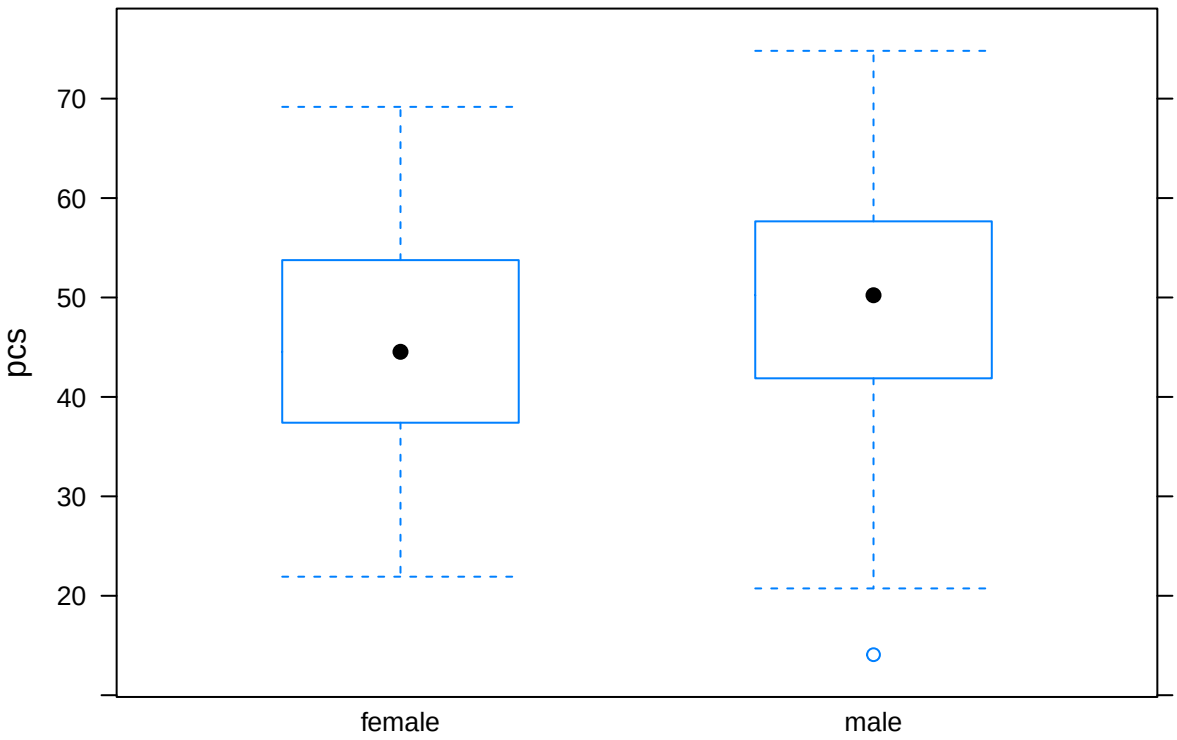
```
tally(~ sex, format="percent", data=HELPrct)
```

```
##
## female   male
##   23.6   76.4
```

```
mean(pcs ~ sex, data=HELPrct)
```

```
## female   male
##     45     49
```

```
bwplot(pcs ~ sex, data=HELPrct)
```



Now let's fit a multiple regression model for PCS (physical component scores) that includes mcs, sex, and substance (3 levels).

```
mlrmaineffect <- lm(pcs ~ mcs + sex + substance, data=HELPrct)
coef(mlrmaineffect)
```

```
##      (Intercept)              mcs          sexmale substancecocaine
##          41.9915           0.0501           4.0708           4.5872
##  substanceheroin
##          -0.6492
```

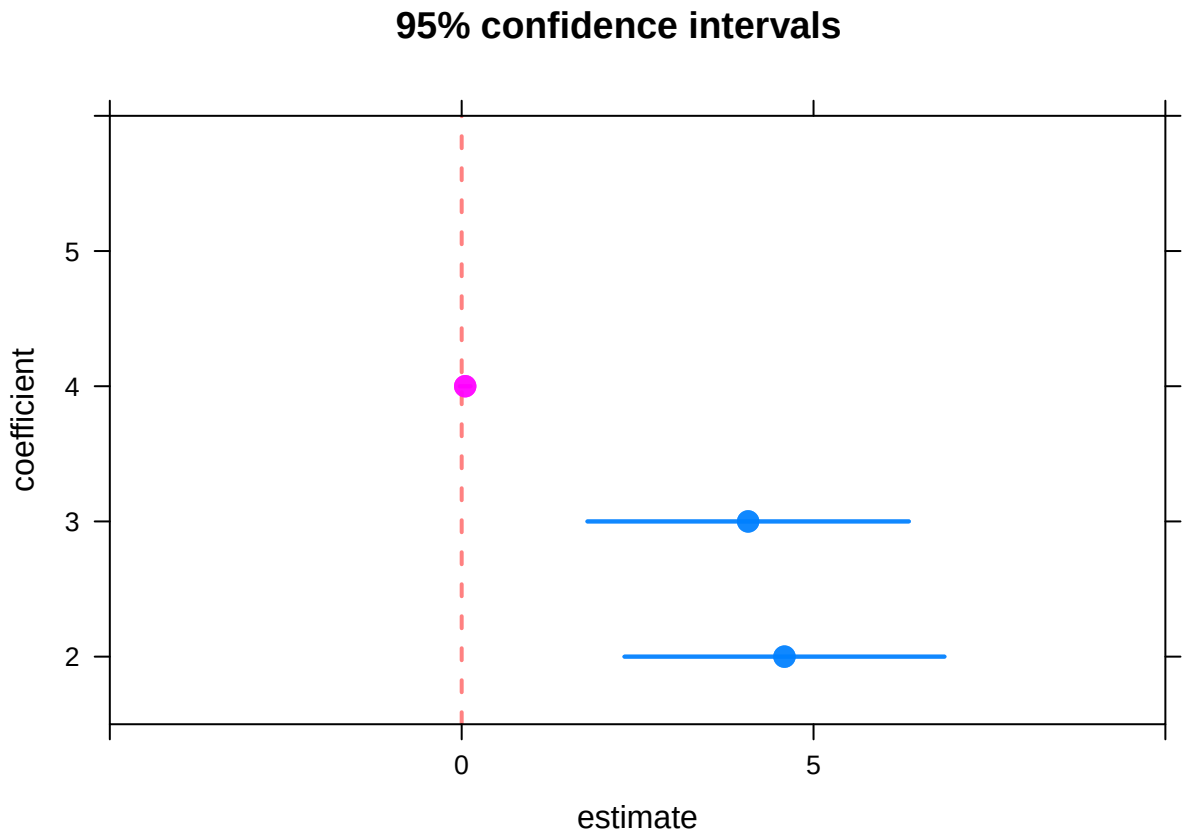
```
msummary(mlrmaineffect)
```

```
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       41.9915     1.6512   25.43  < 2e-16 ***
## mcs                0.0501     0.0390    1.28  0.19993
## sexmale            4.0708     1.1624    3.50  0.00051 ***
## substancecocaine   4.5872     1.1572    3.96  8.6e-05 ***
## substanceheroin   -0.6492     1.2260   -0.53  0.59669
##
## Residual standard error: 10.4 on 448 degrees of freedom
## Multiple R-squared:  0.0777, Adjusted R-squared:  0.0695
## F-statistic: 9.43 on 4 and 448 DF,  p-value: 2.5e-07
```

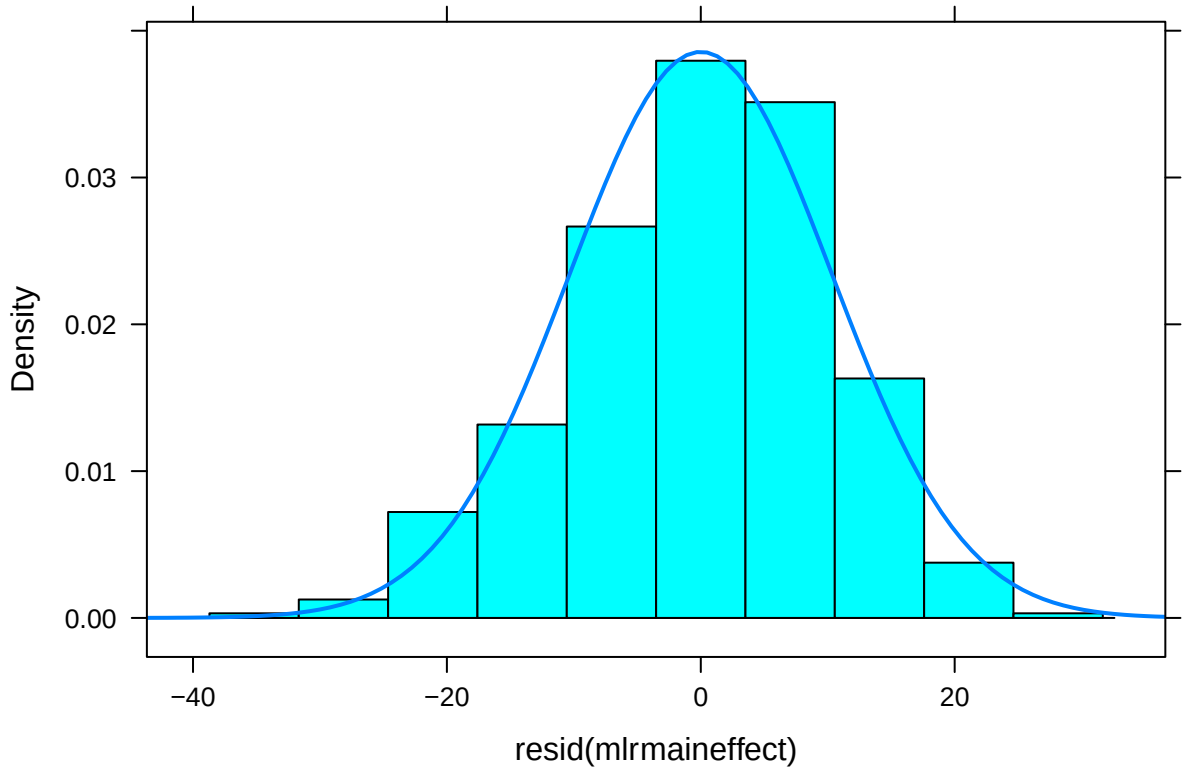
Let's do some model diagnostics.

```
mplot(mlrmaineffect, which=7, xlim=c(-5, 10), ylim=c(1.5, 6))
```

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




```
histogram(~ resid(mlrmaineffect), fit='normal')
```

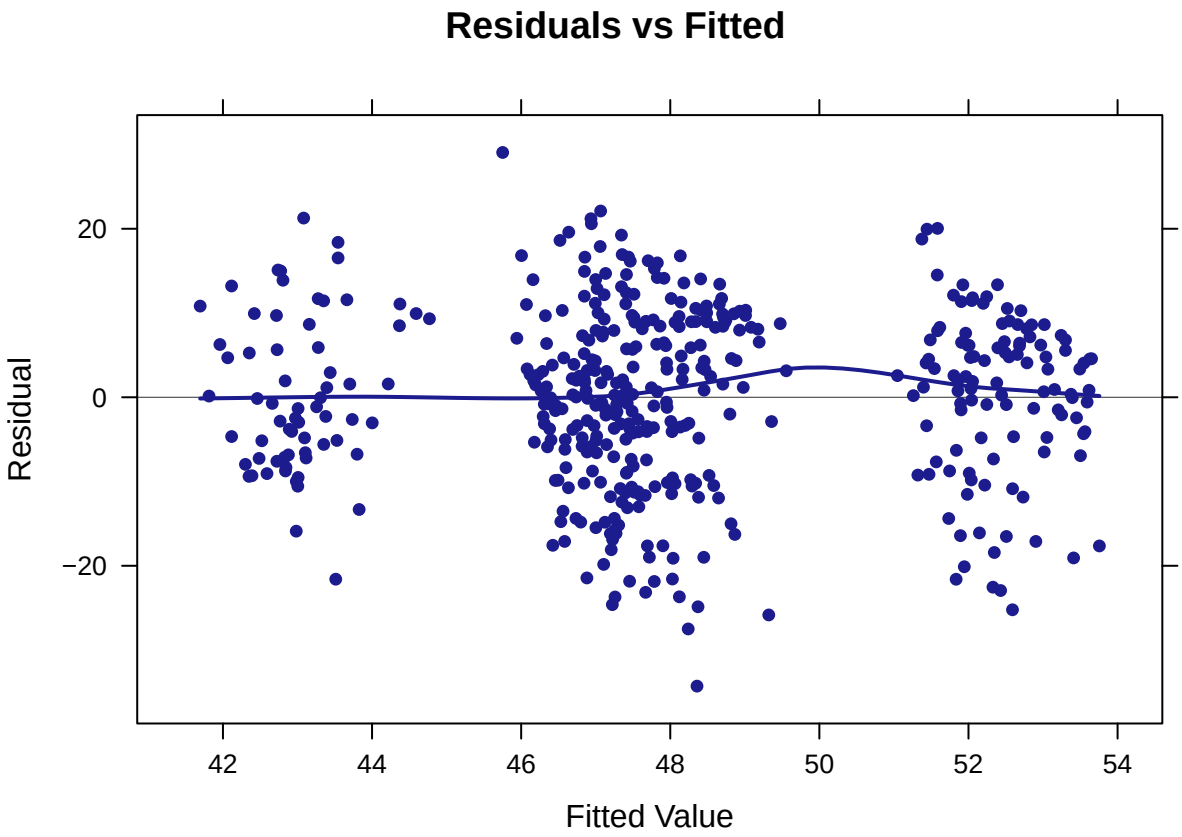


Let's plot some predicted values, say for a alcohol involved subject, as a function of being male vs. female

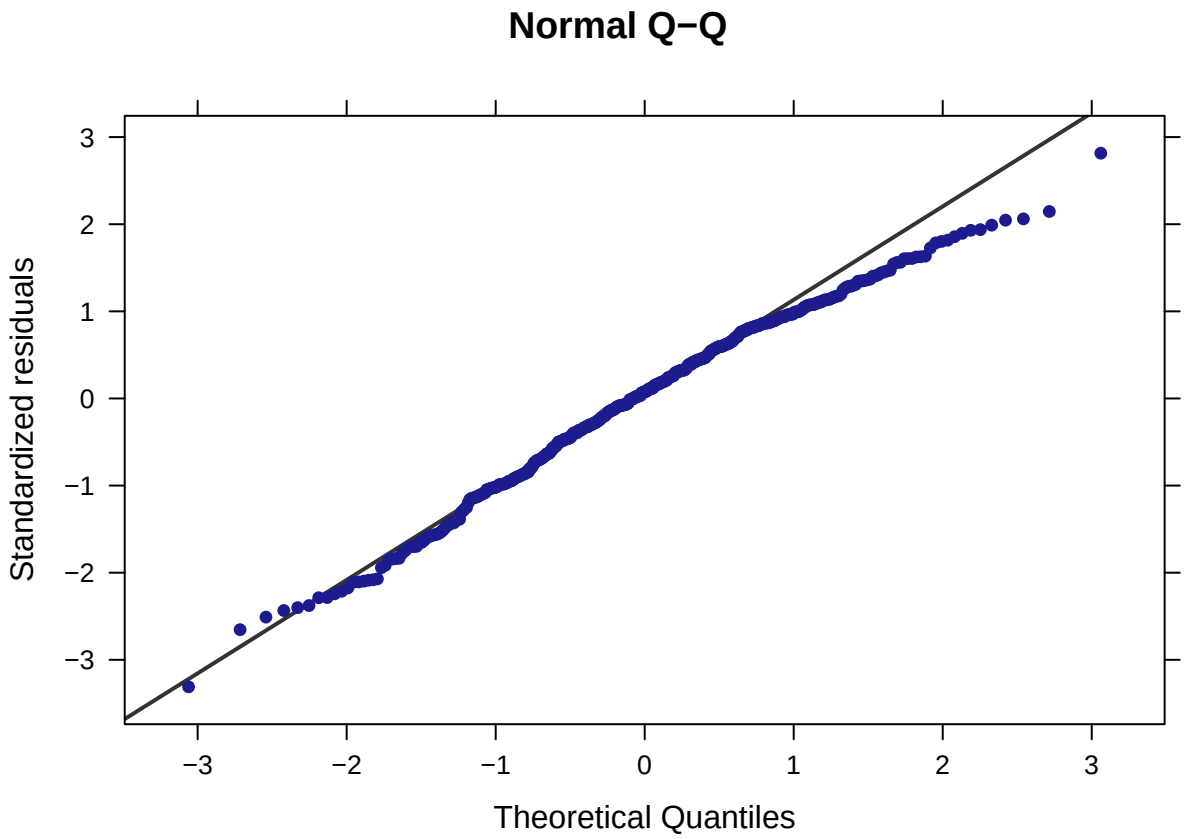
and MCS score.

```
mplot(mlrmaineffect)
```

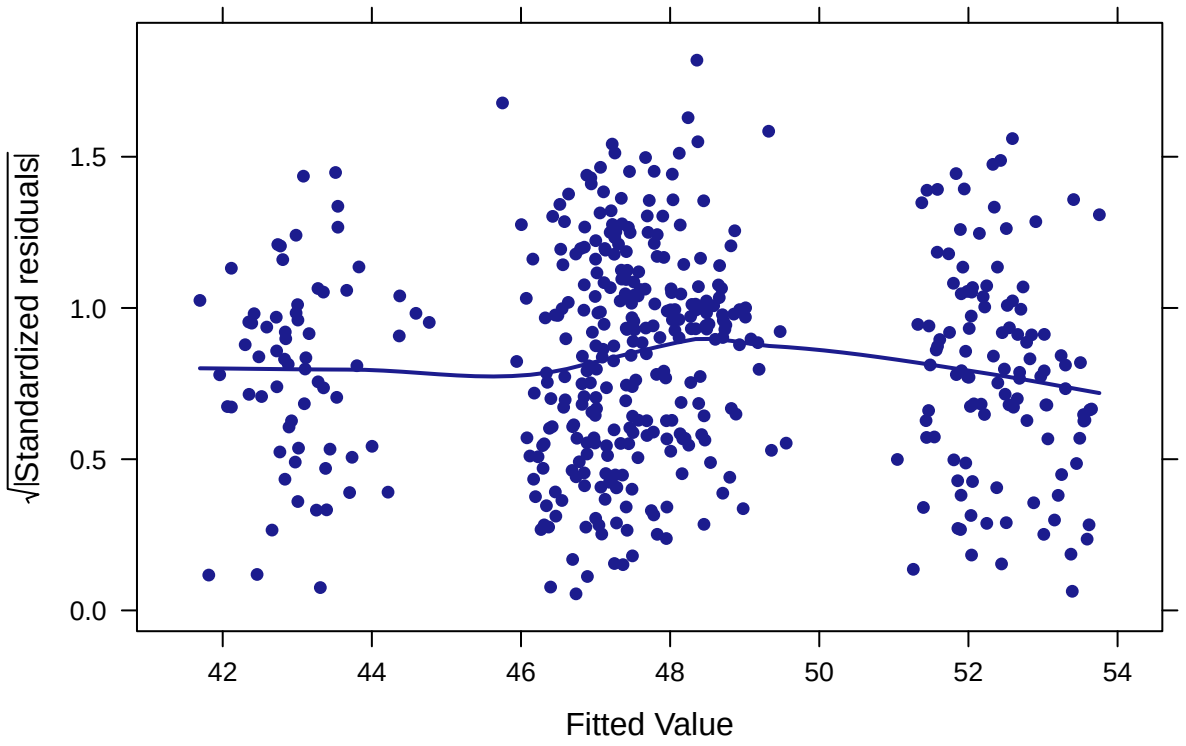
```
## [[1]]
```



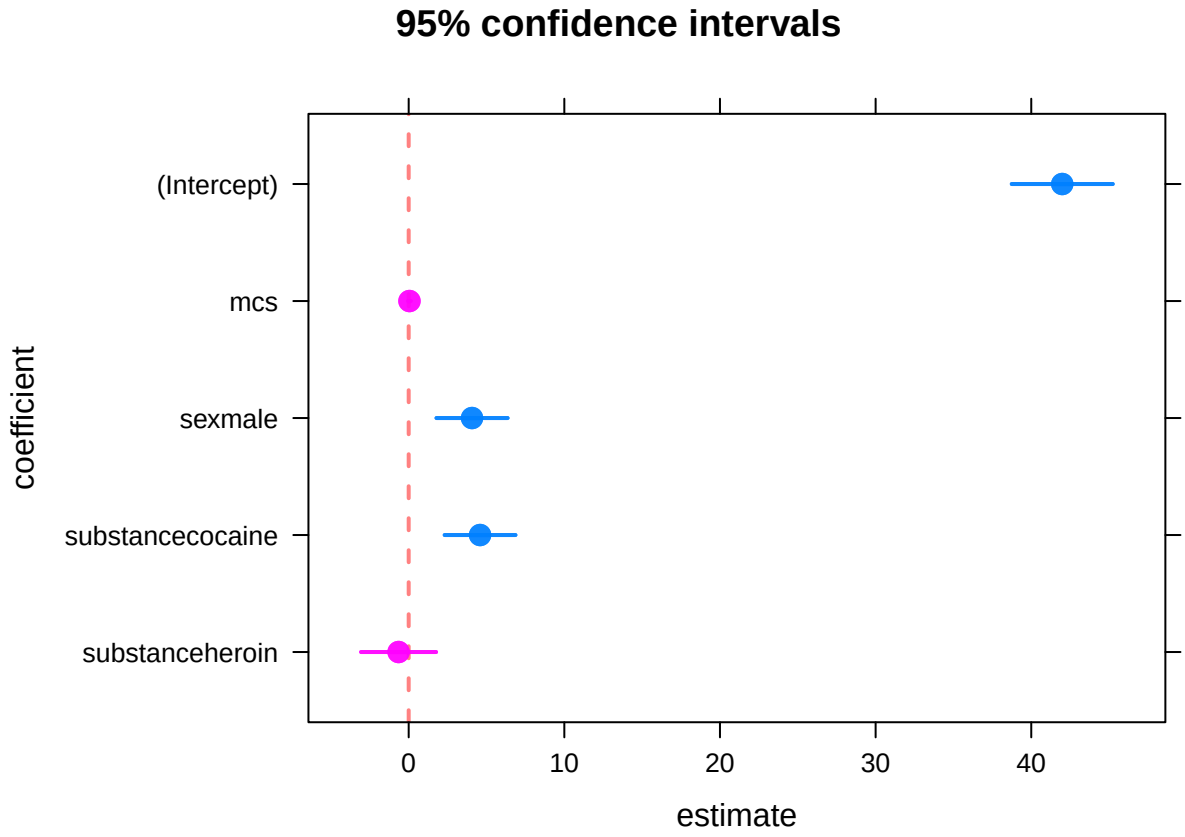

```
##
## [[2]]
```


Normal Q−Q
Standardized residuals
Theoretical Quantiles
3
2
1
0
−1
−2
−3
−3
−2
−1
0
1
2
3

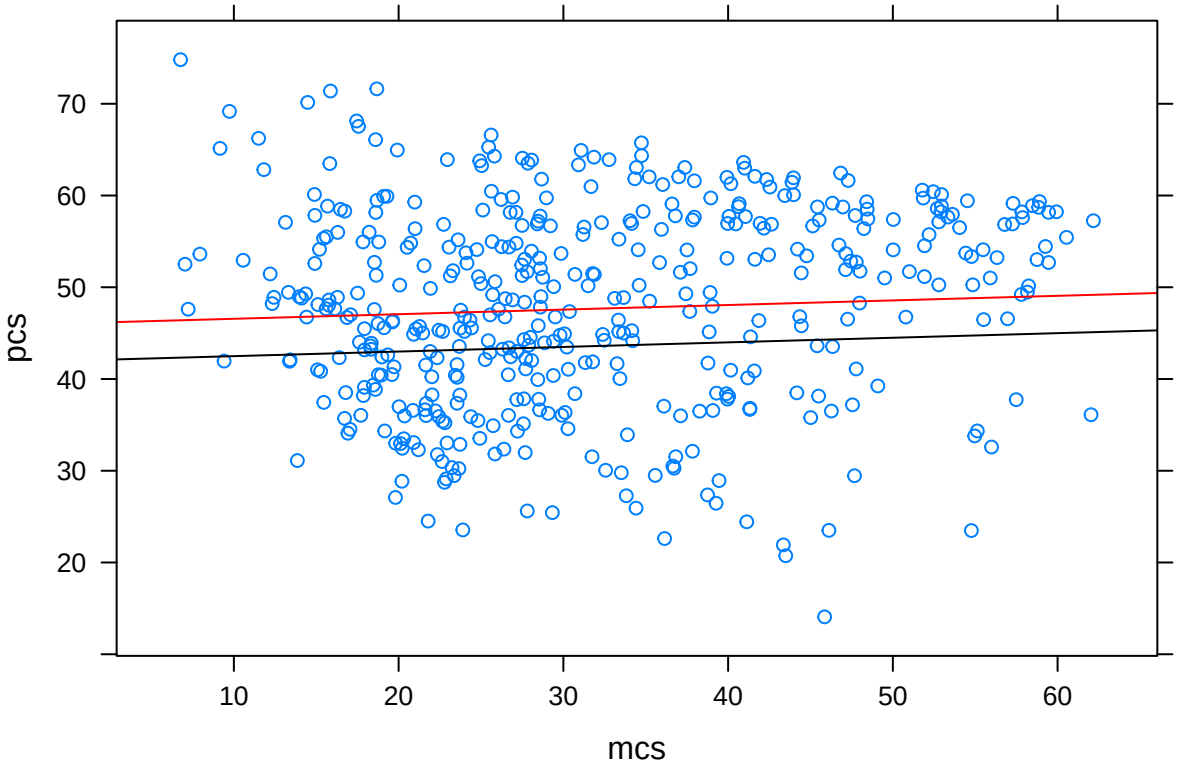
```
##
## [[3]]
```


Scale−Location
√|Standardized residuals|
1.5
1.0
0.5
0.0
42
44
46
48
50
52
54
Fitted Value

```
##
## [[4]]
```

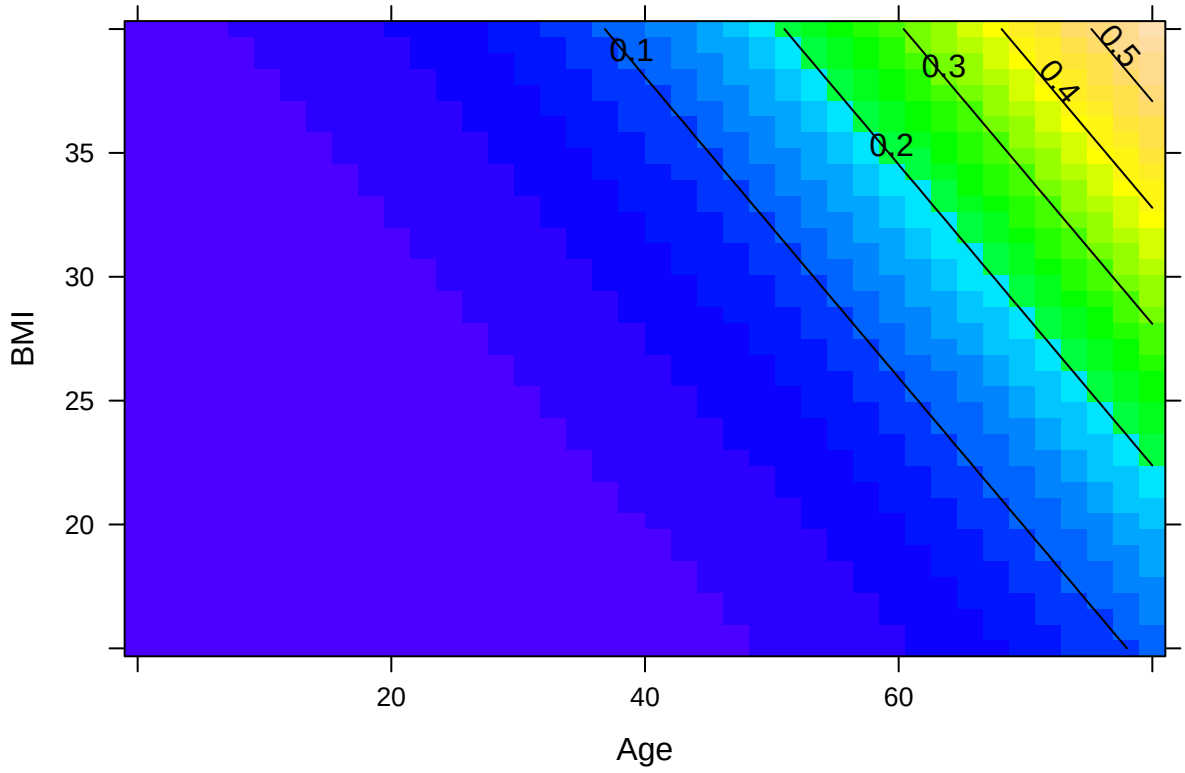



```
mlrmefun <- makeFun(mlrmaineffect)
xyplot(pcs ~ mcs, data=HELPrct)
plotFun(mlrmefun(mcs, sex="female", substance="alcohol") ~ mcs, add=TRUE, col="black")
plotFun(mlrmefun(mcs, sex="male", substance="alcohol") ~ mcs, add=TRUE, col="red")
```

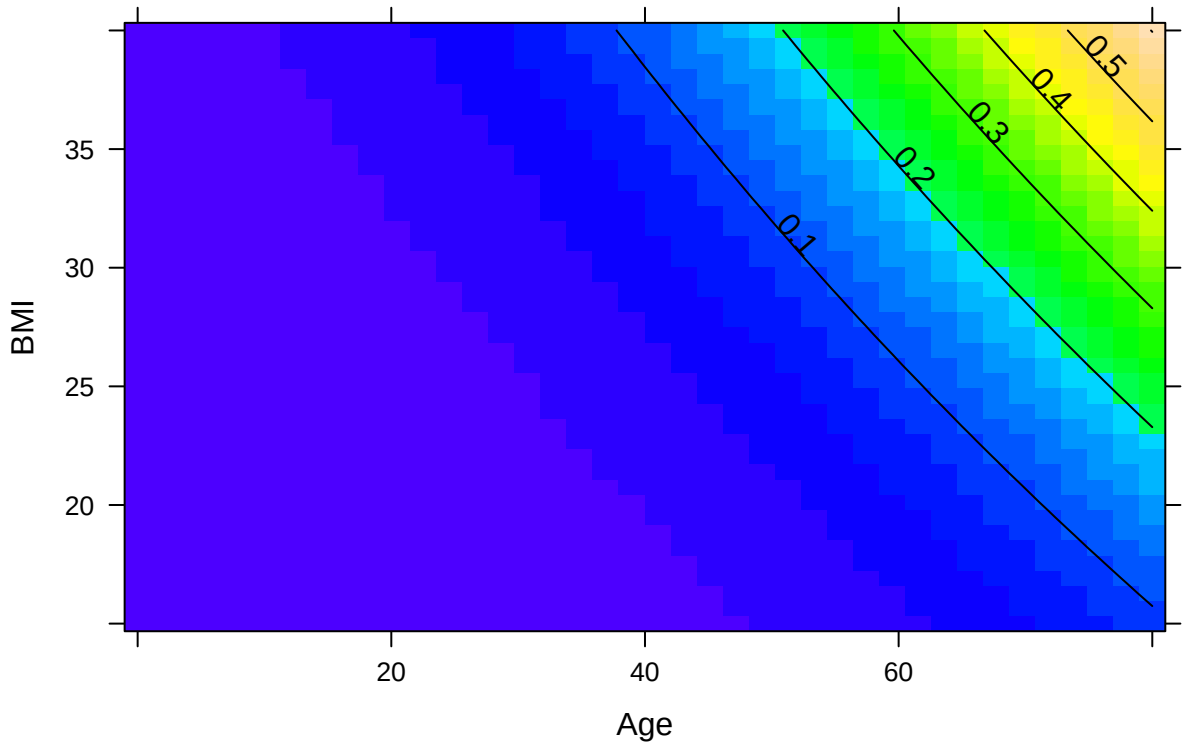


But for something even more interesting

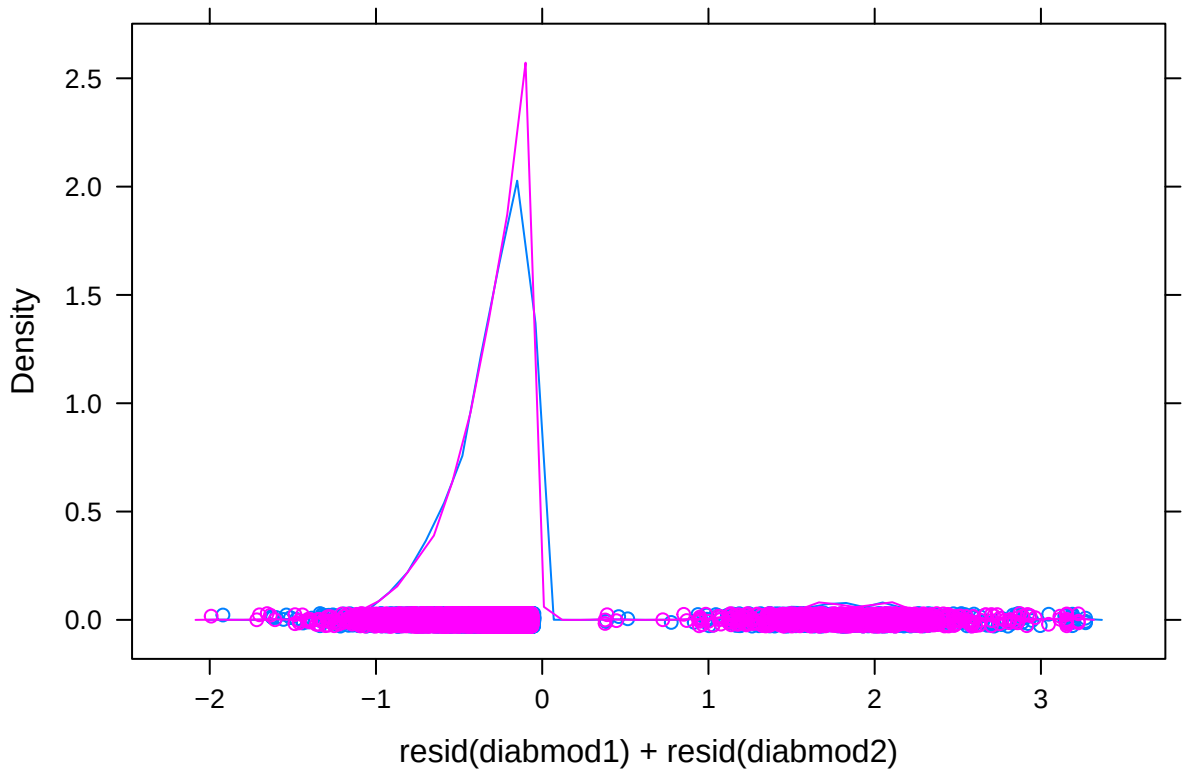
```
diabmod1 <- glm(Diabetes ~ Age + BMI, family="binomial", data=NHANES)
diabfun1 <- makeFun(diabmod1)
plotFun(diabfun1(Age=Age, BMI=BMI) ~ Age + BMI, xlim=c(0, 80), ylim=c(15, 40))
```



```
diabmod2 <- glm(Diabetes ~ Age + BMI + Age*BMI, family="binomial",
  data=NHANES)
diabfun2 <- makeFun(diabmod2)
plotFun(diabfun2(Age=Age, BMI=BMI) ~ Age + BMI, xlim=c(0, 80), ylim=c(15, 40),
        main="Difference in predicted probabilities of diabetes (Interaction Model)")
```

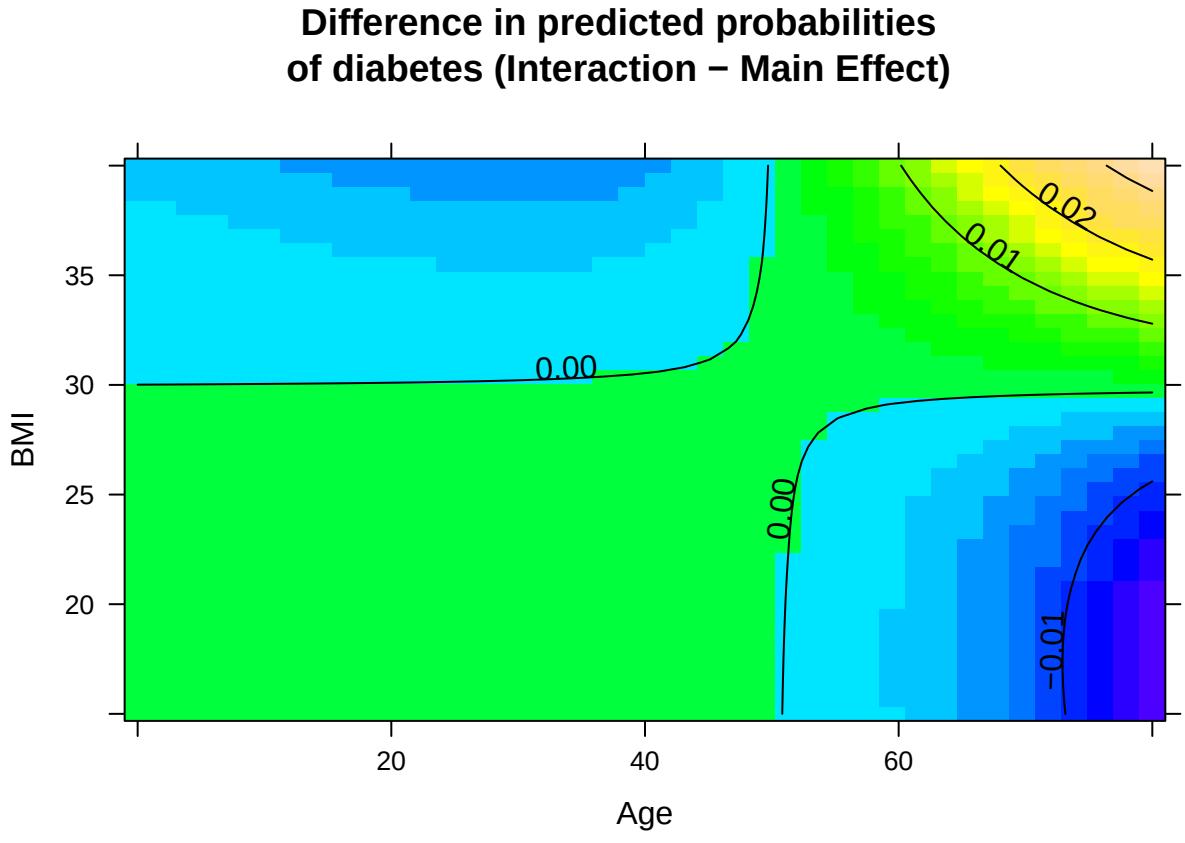



```
densityplot(~ resid(diabmod1) + resid(diabmod2))
```



Contour plots...

```
plotFun((diabfun2(Age=Age, BMI=BMI) - diabfun1(Age=Age, BMI=BMI)) ~ Age + BMI, xlim=c(0, 80), ylim=c(15
```



```r
smallNHANES <- sample(NHANES, 500)
```