

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
```

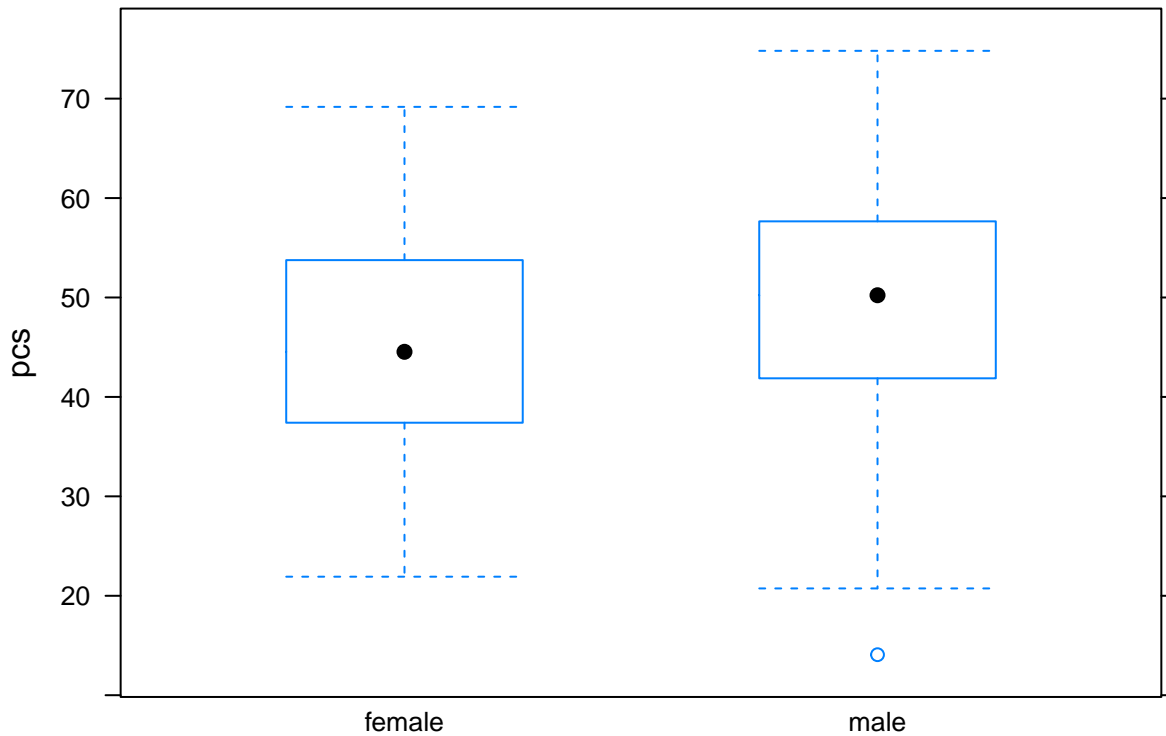
```
tally(~ sex, format="percent", data=HELPrct)
```

```
##
## 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
```

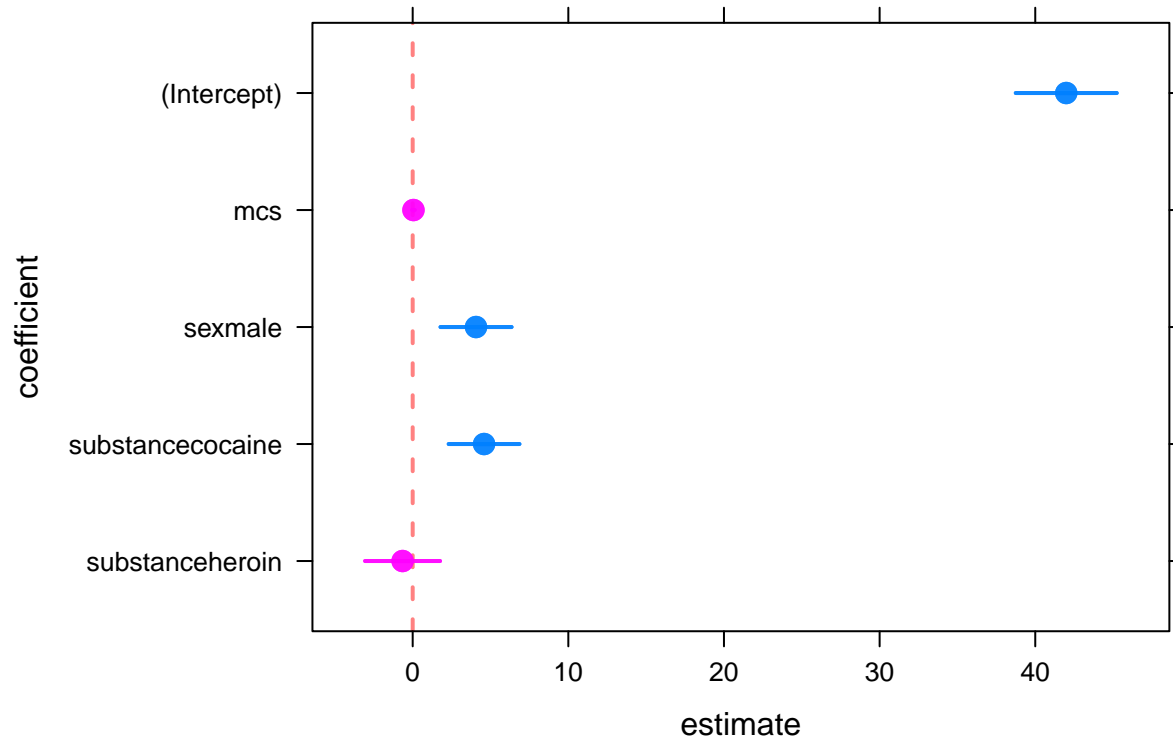
```
msummary(mlrmaineffect)
```

```
##      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
```

```
mplot(mlrmaineffect, which=7)
```

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

95% confidence intervals

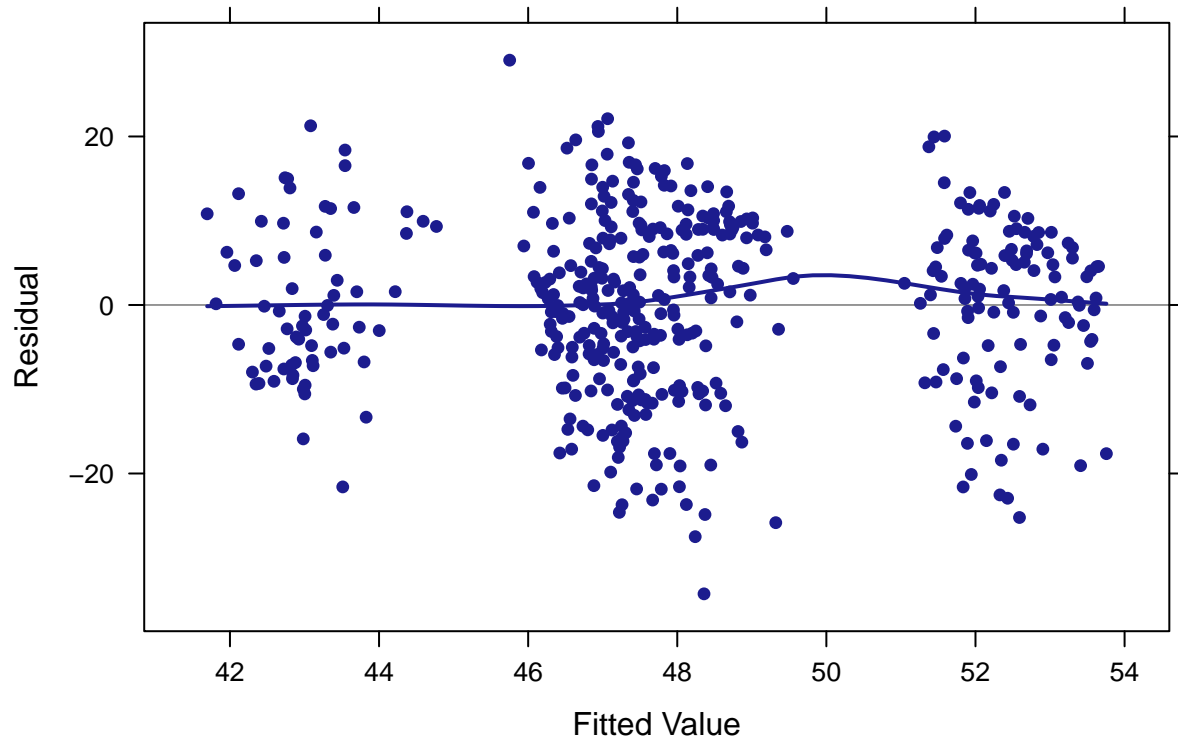


Let's do some model diagnostics.

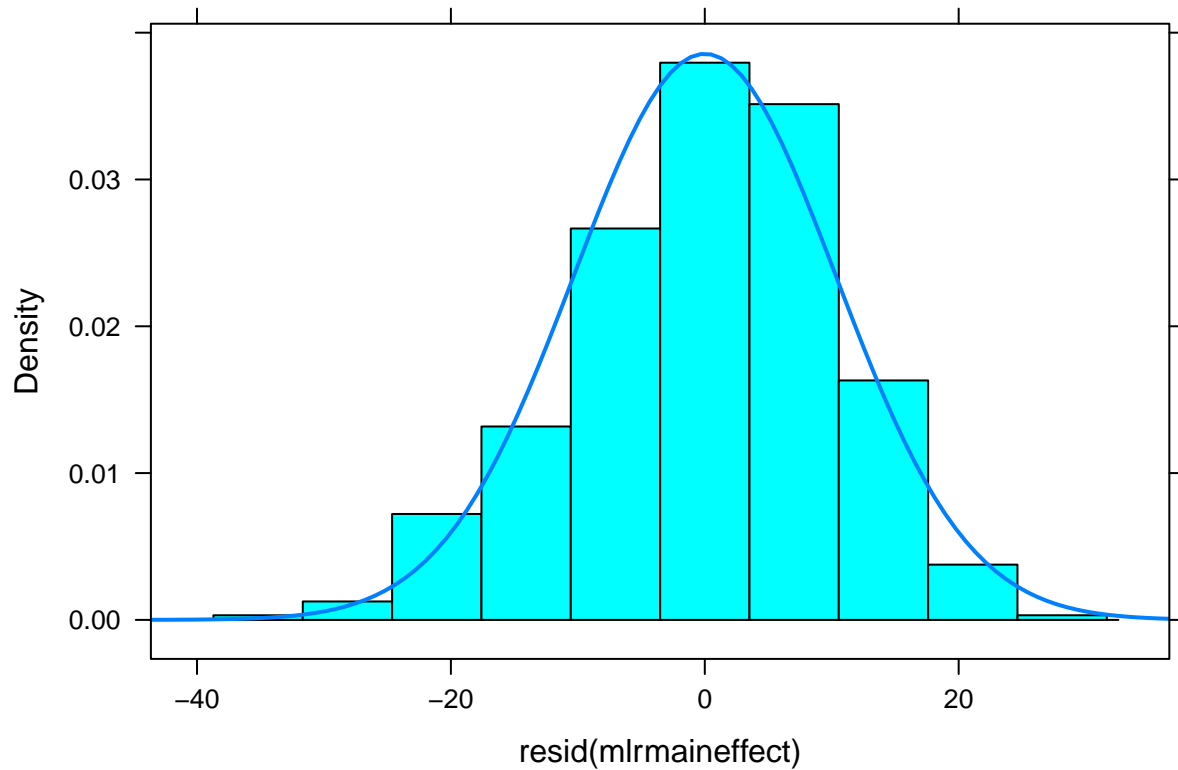
```
mpplot(mlrmaineffect, which=1)
```

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

Residuals vs Fitted



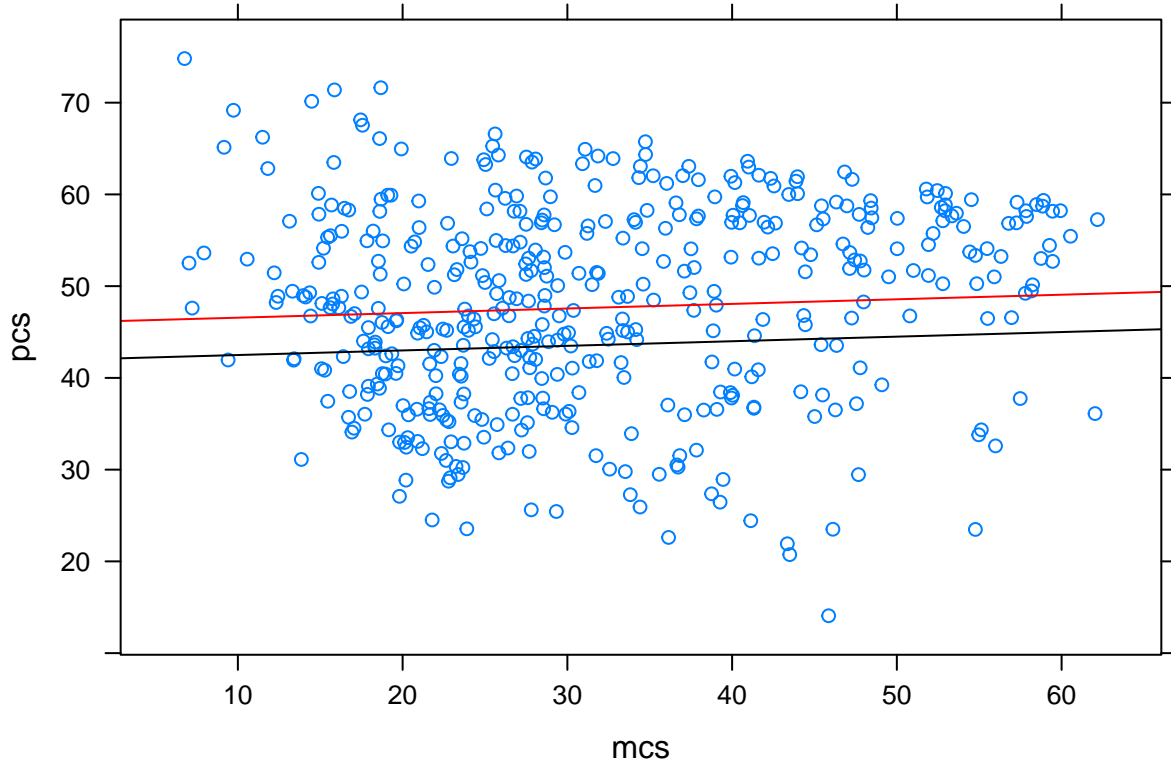
```
histogram(~ resid(mlrmaineffect), fit='normal')
```



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

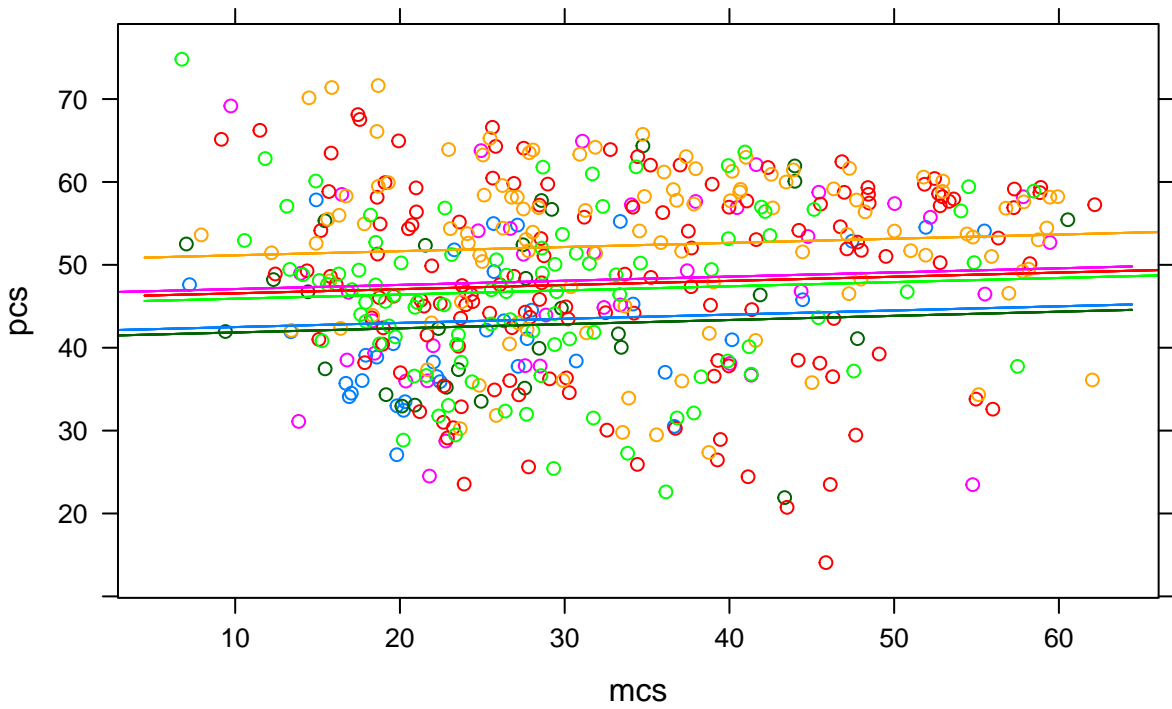
and MCS score.

```
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")
```



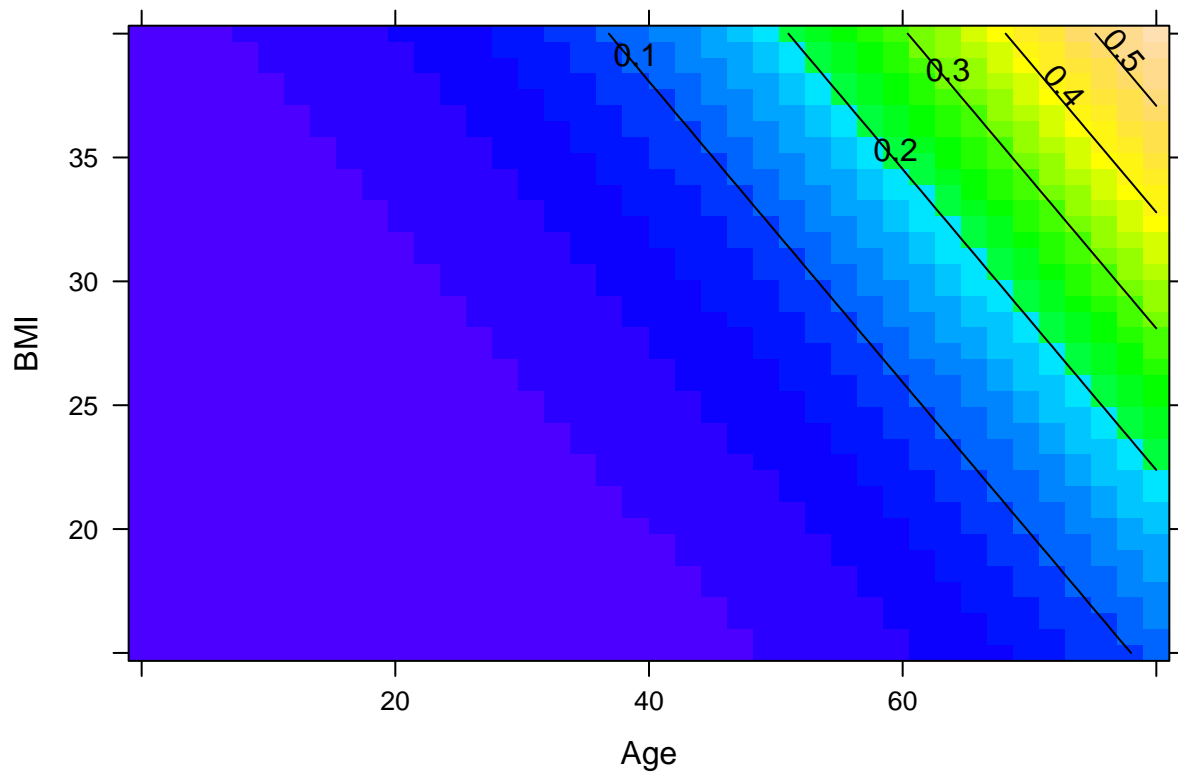
```
plotModel(mlrmaineffect)
```

female:alcohol ○ ——— female:heroin ○ ——— male:cocaine ○ ———
 female:cocaine ○ ——— male:alcohol ○ ——— male:heroin ○ ———



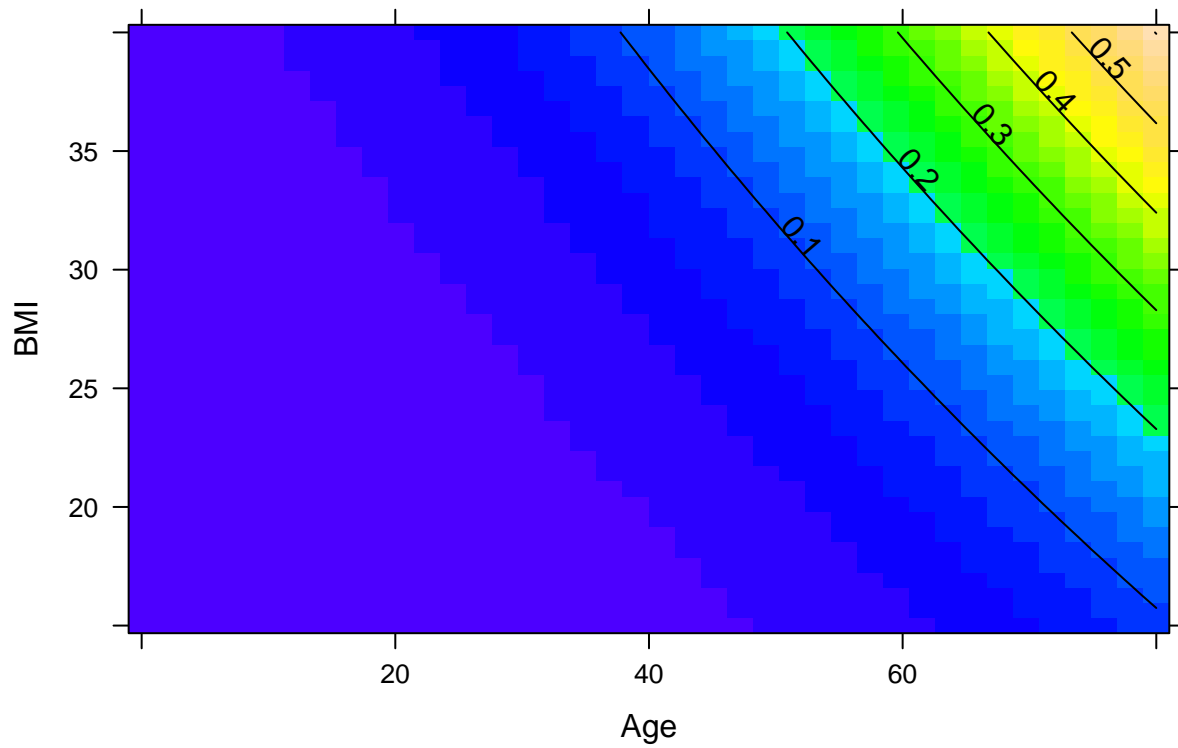
But for something even more interesting

```
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)")
```

Difference in predicted probabilities of diabetes (Interaction Model)



Contour plots...

```
plotFun((diabfun2(Age=Age, BMI=BMI) - diabfun1(Age=Age, BMI=BMI)) ~ Age + BMI, xlim=c(0, 80), ylim=c(15, 35))
```


Difference in predicted probabilities of diabetes (Interaction – Main Effect)

