

# Exploration of power and treatment effect precision using an adjusted analysis for a randomised controlled trial (Bayesian & frequentist)

*Eamonn O'Brien*

*01 August, 2016*

## 1 Introduction

Many randomised controlled trials (RCTs) are analysed in a simple manner using only the randomised treatment as the independent variable. But when the response outcome is continuous, precision of the treatment effect estimate is improved when adjusting for baseline covariates in a randomised controlled trial. We do not expect covariates to be related to the treatment assignment because of randomisation, but they may be related to the outcome, they are therefore not considered to be confounding. However, differences between the outcome which can be attributed to differences in the covariates can be removed, this results in a more precise estimate of treatment effect. This should be considered more often as sample sizes can be reduced. This is not the case for logistic models (principle of the non collapsibility of the odds ratio).

Here, we perform an investigation of adjusted and unadjusted power in the setting of a RCT. We then choose a sample size and simulate a single RCT. The RCT is then analysed using a frequentist approach and a Bayesian approach.

## 2 Prepare for stochastic simulations

```
alpha <- 0.05
n.sims <- 200                                     # simulations
N <- seq(from=100, to=1000, by=50)                # trial sizes to explore
pow.adj1 <- pow.adj2 <- pow <- rep(NA, length(N)) # object to capture summary power estimates
bias <- array(NA, dim=c(length(N),6))             # object to capture summary coeff estimates
```

## 3 Function to investigate power for a RCT contingent on the analyses performed (adjusted and unadjusted)

```
for (j in 1:length(N)){
  n <- N[j]

  #objects to collect simulations for each sample size before summary stats are calculated
  significant.unadjusted <- significant.adj1 <- significant.adj2 <- rep(NA, n.sims)
  est <- array(NA, dim=c(n.sims,6))

  for (i in 1: n.sims){
```

```

# Hyp the effect of age on outcome, a unit change in outcome over the whole age range,
# so ~0.02 effect on outcome for each increase in age of 1 year
effect.of.age <- 1/(65-18)
effect.of.sex <- 5 # Hyp the effect of sex on outcome
effect.of.treatment <- 1 # Hyp the treatment effect on outcome

# Hypothesize baseline version of response on outcome,
# a unit increase results in 0.5 increase in outcome,
# strong predictor so 5 over the baseline range
effect.of.baseline <- 1/2
intercept <- 100
noise <- 5 # random error

#collect the true parmaeters to see how well we recover them
covp <- c(intercept, effect.of.baseline, effect.of.sex,
          effect.of.age, effect.of.treatment, effect.of.treatment)

# covariate effects
baseline <- runif(n,100,110) # baseline effect
treat <- 1*(runif(n)<0.5) # randomised treatment effect
sex <- c(rep("F", n/2), rep("M", n/2)) # Generate sex covariate
age <- sample(x=18:65, size=n, replace=TRUE) # Generate age covariate

# create the outcome, with random error
Y <- effect.of.sex*(sex=="M") + effect.of.age*age + effect.of.baseline*baseline +
  effect.of.treatment*treat + rnorm(n, intercept, noise)

# prepare for analysis
d <- data.frame(Y=Y, baseline=baseline, treat=treat, sex=sex, age=age)
dd <- datadist(d, data=d); options(datadist="dd")

# Unadjusted analysis, extract p-value treatment effect
f0 <- ols(Y ~ treat, d)
fa <- anova(f0)
p.value <- as.matrix(fa)[1,5]

# Adjusted analysis, extract p-value treatment effect adjusted for 1 covariates
fa <- anova(ols(Y ~ baseline + treat, d))
p.value.adj1 <- as.matrix(fa)[2,5]

# Adjusted analysis, extract p-value treatment effect adjusted for 3 covariates
f <- ols(Y ~ baseline + sex + age + treat, d)
fa <- anova(f)
p.value.adj2 <- as.matrix(fa)[4,5]

# collect p values
significant.unadjusted[i] <- p.value <= alpha
significant.adj1[i] <- p.value.adj1 <= alpha
significant.adj2[i] <- p.value.adj2 <= alpha

#get the simulations coefficient estimates for sample size
est[i,]<- c(f$coefficients, f0$coefficients[2][[1]])

```

```

}

# summarise the simulations, calculate power
pow[j]      <- mean(significant.unadjusted)
pow.adj1[j] <- mean(significant.adj1)
pow.adj2[j] <- mean(significant.adj2)

# mean the simulations for the parameter estimates
x <- apply(est, 2, mean)

# calculate relative bias and store
bias[j,] <- 100*(x-covp)/covp
}

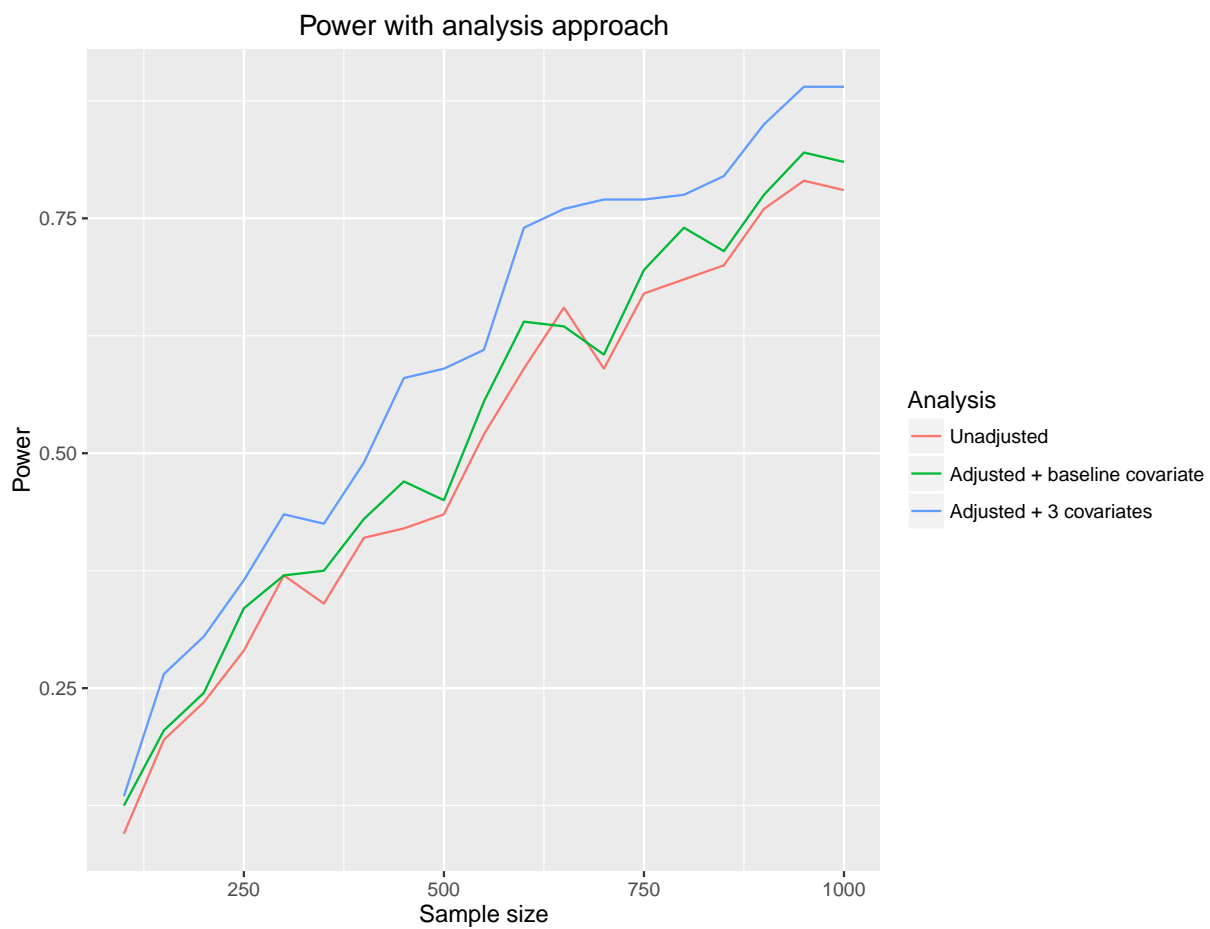
```

## 4 Plot the power with sample size relationships

```
df <- data.frame(N, pow, pow.adj1, pow.adj2)
df2 <- melt(data = df, id.vars = "N") # melt the data to a long format and plot

g <- ggplot(data = df2, aes(x = N, y = value, color= factor(variable,
labels = c("Unadjusted", "Adjusted + baseline covariate" , "Adjusted + 3 covariates")))) +
  geom_line()
g <- g + ylab("Power") + xlab("Sample size") + labs(color = "Analysis") +
  ggtitle("Power with analysis approach")

print(g)
```



## 5 Plot the percent relative bias with sample size relationships

```

namez <- c("intercept","effect.of.baseline","effect.of.sex","effect.of.age",
           "effect.of.treatment", "unadj.effect.of.treatment")
bias <- data.frame(bias)
names(bias) <- namez

df <- data.frame(cbind(N=N, bias))
df2 <- melt(data = df, id.vars = "N") # melt the data to a long format and plot

g <- ggplot(data = df2, aes(x = N, y = value, color= factor(variable, labels = namez))) +
  geom_line()
g <- g + ylab("% Relative bias") + xlab("Sample size") + labs(color = "Parameter") +
  ggtitle("% Relative Bias in coefficients") +
  geom_hline(aes(yintercept=0), color="black", linetype="dashed")
g <- g + facet_grid(. ~ variable) + guides(colour=FALSE)

print(g)

```

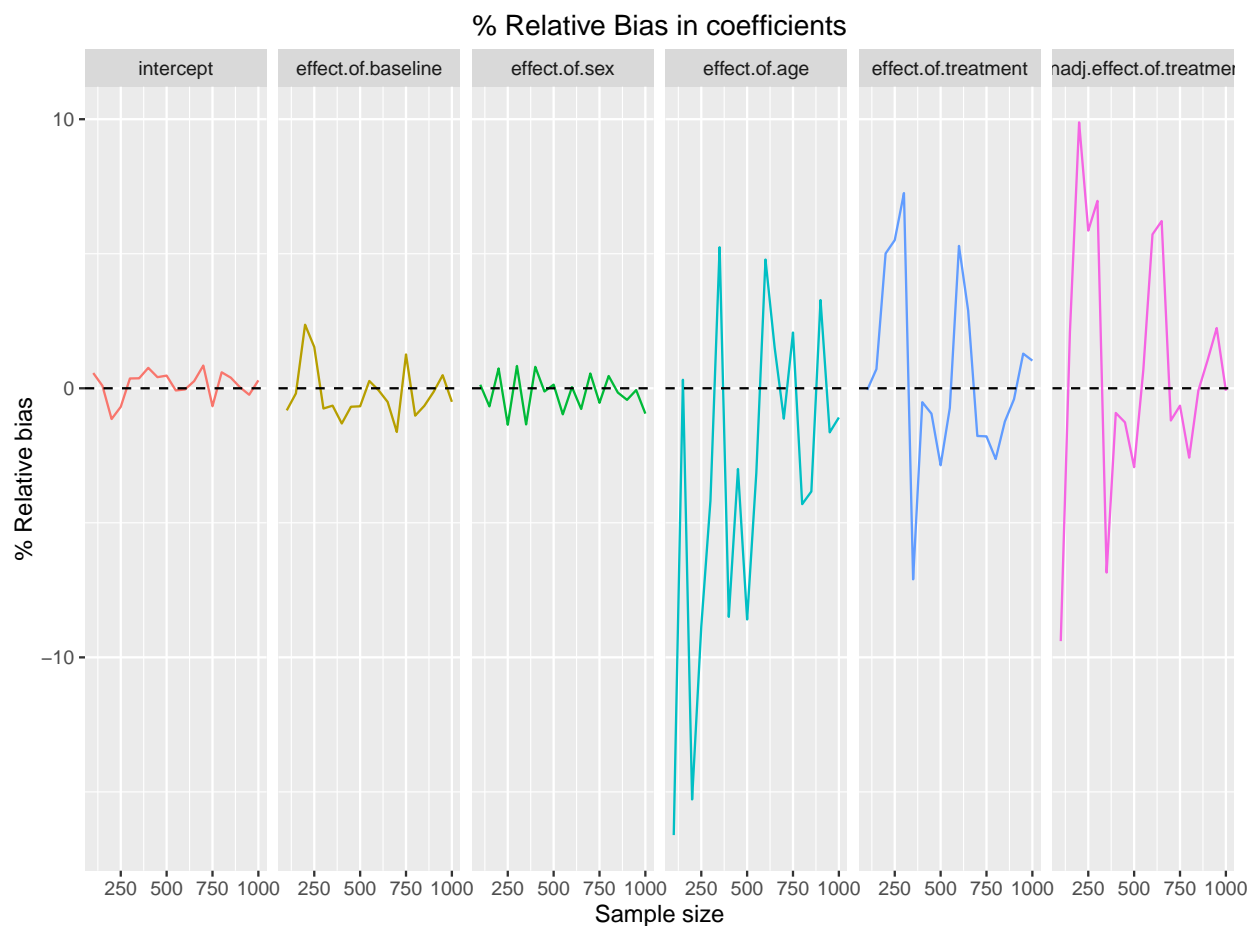


Figure 1: Profiles of percentage relative bias with sample size

## 6 Show the % relative bias with sample size relationships

```
x.c <- bias # format all numerical variables
x.c[] <- lapply(x.c, function(.col){
  if (is.numeric(.col)) return(sprintf("%.1f", .col))
  else return(.col)
})

print(x.c)
```

	intercept	effect.of.baseline	effect.of.sex	effect.of.age	effect.of.treatment	unadj.effect.of.treatment	
1	0.6	-0.8	0.1	-16.6	-0.1		-9
2	0.1	-0.2	-0.7	0.3	0.7		2
3	-1.1	2.4	0.7	-15.3	5.0		9
4	-0.7	1.5	-1.4	-8.9	5.5		5
5	0.4	-0.8	0.8	-4.2	7.3		7
6	0.4	-0.6	-1.3	5.2	-7.1		-6
7	0.8	-1.3	0.8	-8.5	-0.5		-0
8	0.4	-0.7	-0.1	-3.0	-0.9		-1
9	0.5	-0.7	0.1	-8.6	-2.9		-2
10	-0.1	0.3	-1.0	-3.2	-0.7		0
11	-0.0	-0.1	0.0	4.8	5.3		5
12	0.3	-0.5	-0.8	1.5	2.9		6
13	0.8	-1.6	0.5	-1.1	-1.8		-1
14	-0.7	1.3	-0.5	2.1	-1.8		-0
15	0.6	-1.0	0.5	-4.3	-2.6		-2
16	0.4	-0.7	-0.2	-3.8	-1.2		-0
17	0.0	-0.1	-0.4	3.3	-0.4		1
18	-0.2	0.5	-0.1	-1.6	1.3		2
19	0.3	-0.5	-0.9	-1.1	1.0		-0

## 7 Based on power estimates of adjusted analyses a trial of 800 patients takes place

```
set.seed(87564)
n <- 800 # sample size

effect.of.age <- 1/(65-18) # Hypothesize the effect of age on outcome
effect.of.sex <- 5 # Hypothesize the effect of sex on outcome
effect.of.treatment <- 1 # Hypothesize the treatment effect on outcome
effect.of.baseline <- 1/2 # Hypothesize the baseline effect on outcome
intercept <- 100
noise <- 5 # the standard deviation for the simulated data

baseline <- runif(n,100,110) # Generate baseline covariate
treat <- 1*(runif(n)<0.5) # randomised treatment effect
sex <- c(rep("F", n/2), rep("M", n/2)) # Generate sex covariate
age <- sample(x=18:65, size=n, replace=TRUE) # Generate age covariate
```

```

# Create the response outcome, with random error
Y <- effect.of.sex*(sex=="M") + effect.of.age*age + effect.of.baseline*baseline +
effect.of.treatment*treat + rnorm(n, intercept, noise)

#~~~~~
# alternative data creation
dat <- data.frame( baseline=baseline, treat=treat, sex=sex, age=age)

X <- model.matrix(~baseline + treat + sex + age, dat)

#the regression slopes
betas <-c(intercept, effect.of.baseline, effect.of.treatment, effect.of.sex, effect.of.age)

# the simulated data
y_norm <- rnorm(800, X%*%betas, noise)
#~~~~~

lm( Y ~      baseline + sex + age + treat)

```

Call:

```
lm(formula = Y ~ baseline + sex + age + treat)
```

Coefficients:

(Intercept)	baseline	sexM	age	treat
96.326964	0.535192	5.183220	0.009781	1.706045

```
lm( y_norm ~ baseline + sex + age + treat)
```

Call:

```
lm(formula = y_norm ~ baseline + sex + age + treat)
```

Coefficients:

(Intercept)	baseline	sexM	age	treat
94.1915	0.5508	5.8003	0.0232	1.1279

```
#~~~~~
```

## 8 Create a data frame

```

d <- data.frame(Y=Y, baseline=baseline, treat=treat, sex=sex, age= age )
dd <- datadist(d, data=d) ; options(datadist="dd")

```

## 9 Bayesian analysis using STAN, the model statement

```

d$sex2 <- as.numeric(ifelse(d$sex=="M", 1,0) )

m0 <- map( alist( Y ~ dnorm(mu, sigma),

                 mu <- b0 + b4*treat,

                 b0 ~ dnorm(0, 100),
                 b4 ~ dnorm(0, 10),
                 sigma ~ dunif(0, 10) ),
           data=d )

m1 <- map( alist( Y ~ dnorm(mu, sigma),

                 mu <- b0 + b1*baseline + b2*sex2 + b3*age +
                 b4*treat,

                 b0 ~ dnorm(0, 100),
                 c(b1, b2, b3, b4) ~ dnorm(0, 10),
                 sigma ~ dunif(0, 10) ),
           data=d )

# construct centered predictors
# d$baseline <- d$baseline - mean(d$baseline)
# d$age <- d$age - mean(d$age)

m1 <- map2stan( alist( Y ~ dnorm(mu, sigma),

                     mu <- b0 + b1*baseline + b2*sex2 + b3*age +
                     b4*treat,

                     b0 ~ dnorm(0, 100),
                     c(b1, b2, b3, b4) ~ dnorm(0, 10),
                     sigma ~ dunif(0, 10) ),
               data=d , chains=3, iter=5000 , warmup=1000 )

```

## 10 Bayesian analysis using STAN, frequentist and Bayesian results together

	Mean	StdDev	2.5%	97.5%
b0	155.57	0.29	155.00	156.14
b4	1.61	0.42	0.80	2.42
sigma	5.88	0.15	5.59	6.17

	Mean	StdDev	lower 0.95	upper 0.95	n_eff	Rhat
b0	95.90	6.64	83.14	109.16	4212	1
b1	0.54	0.06	0.42	0.66	4244	1
b2	5.18	0.35	4.49	5.88	8222	1
b3	0.01	0.01	-0.01	0.04	10026	1
b4	1.70	0.35	1.02	2.40	8191	1
sigma	5.05	0.13	4.82	5.31	7900	1



## 11 Model assumption check

```
pairs(m1)
```

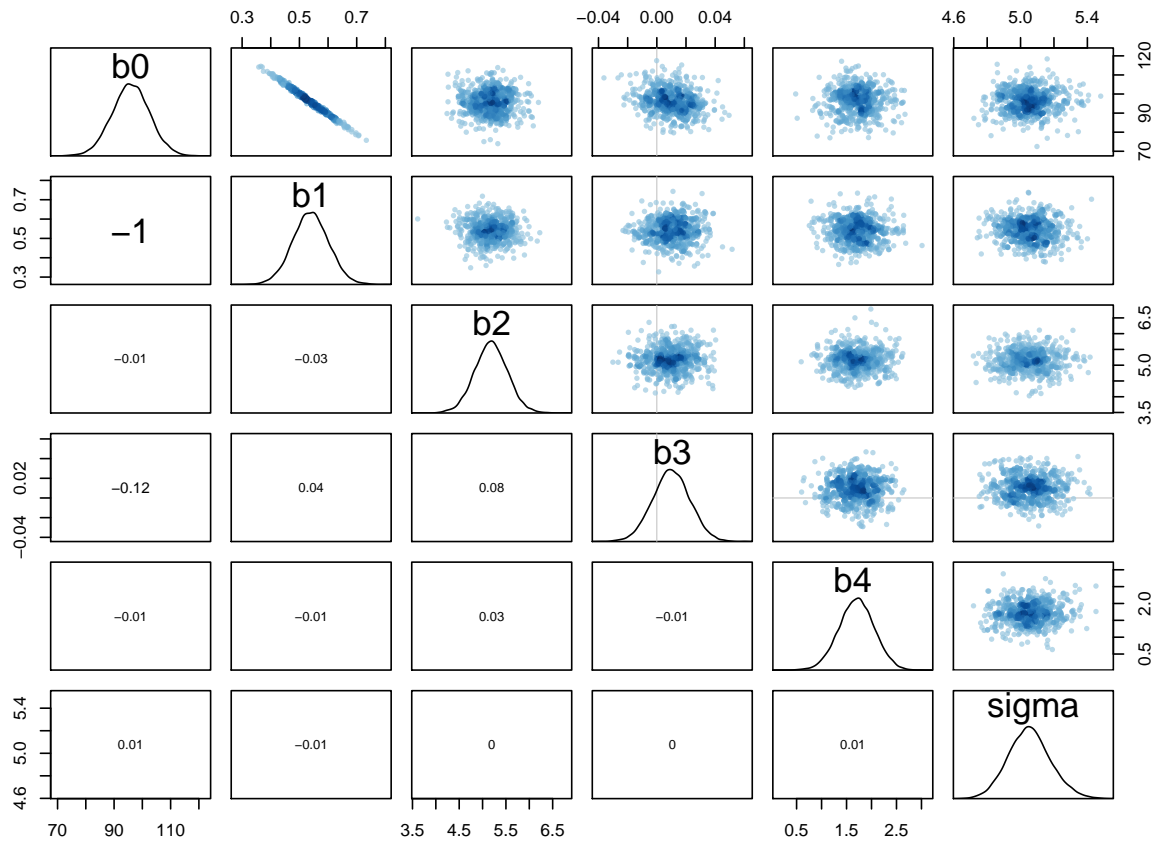
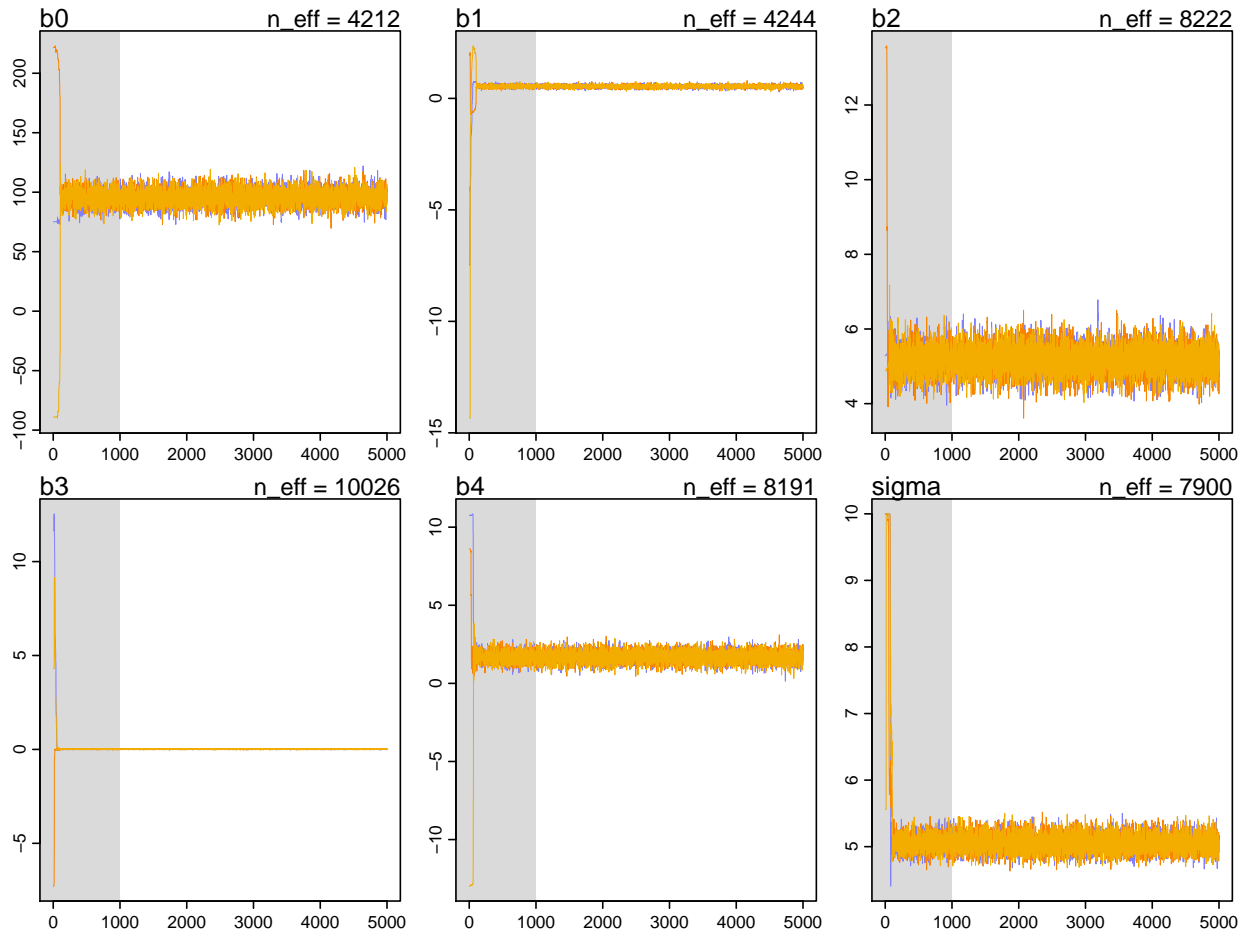


Figure 2: Check chains are mixing

```
post <- extract.samples(m1)
# lapply(post, densityplot)
# par(mfrow=c(3,3))
# for ( i in 1:6 ) { plot(post[i] , type='l', main=names(post)[i]) }
# par(mfrow=c(1,1))

plot(m1)
```



# Plot the predicted effects, counterfactual samples of 100 patients are also simulated and plotted.  
# Treatment effect

```
nn <- 100 # just plot this number of patient rather than 800
post0 <- extract.samples(m0)

namez <- "Treatment"
x <- c(rep(0,nn/2) , rep(1, nn/2))

pred.data1 <- data.frame(                                # Make predictions for this dataset
  treat= x
)

# compute counterfactual mean response (mu1)
mu1 <- link( m0 , data=pred.data1 )
mu.mean <- apply( mu1 , 2 , mean )
mu.PI <- apply( mu1 , 2 , PI )

# simulate counterfactual response outcomes
R.sim1 <- sim( m0 , data=pred.data1 , n=1e4 ) # n < posterior samples or error?
R.PI <- apply( R.sim1 , 2 , PI )

# Need Y to be the same length as x, these values are not plotted
Y <- seq(from=min( R.sim1, na.rm=T), max(R.sim1, na.rm=T), length.out=nn )
```

```

par(mfrow=c(1,2))

# mean and response outcome plot
plot( Y ~ x , type="n", xlab=paste(namez), ylab="Response", xlim=c(-0.2,1.2 ),
ylim=c(min( R.sim1,na.rm=T)*1, max( R.sim1, na.rm=T)*1), xaxt = "n") # get appropriate
axis(side = 1, at = c(0,1), labels = c(0,1), tck=-.05) # better ticks on x axis
# plot this info
lines( x , mu.mean ) # plot this info
shade( mu.PI , x , col= col.alpha("blue", 0.15))
shade( R.PI , x , col= col.alpha("purple", 0.15))
points(R.sim1[1,] ~ jitter(x)) # pick one of the many simulations to plot

# mean effect
plot(Y ~ x, type="n", xlab=paste(namez), ylab="Response" , ylim=c(155,158), xaxt = "n")
axis(side = 1, at = c(0,1), labels = c(0,1), tck=-.05) # better axes
lines( x , mu.mean )
shade( mu.PI , x , col= col.alpha("blue", 0.15))
grid(nx = NULL, ny = NULL)

# title
mtext(line=-3, text= "Unadjusted treatment effect estimate", outer=T, cex=.7)

# title
foo<-post0$b4 # collect the appropriate mcmc samples
mtext( line= -2,
text=paste(namez , " mean effect ", p2(mean(foo)),", SD=", p3(sd(foo)) ,", 95% CI (",
p2(quantile(foo, c(.025))), " , ", p2(quantile(foo, c(0.975))),"), p>0 = ",
mean(foo>0), sep="") ,outer = TRUE, cex = 1)

```

## 12 Adjusted analysis

```

nn<-100 ## just plot this number of patient rather than 800

par(mfrow=c(1,1))
P.baseline <- median(d$baseline) # Hold baseline
P.treat <- c(rep(0,nn/2) , rep(1, nn/2)) # vary
P.age <- median(d$age) # Hold age at median
P.sex <- 0 # Hold sex at female 0

namez <- "Treatment"
x <- P.treat # define the variable here, so less typing later perhaps

pred.data1 <- data.frame( # Make predictions for this dataset
  baseline=P.baseline,
  sex2=P.sex,
  age=P.age,
  treat=x
)

# compute counterfactual mean response (mu1)

```

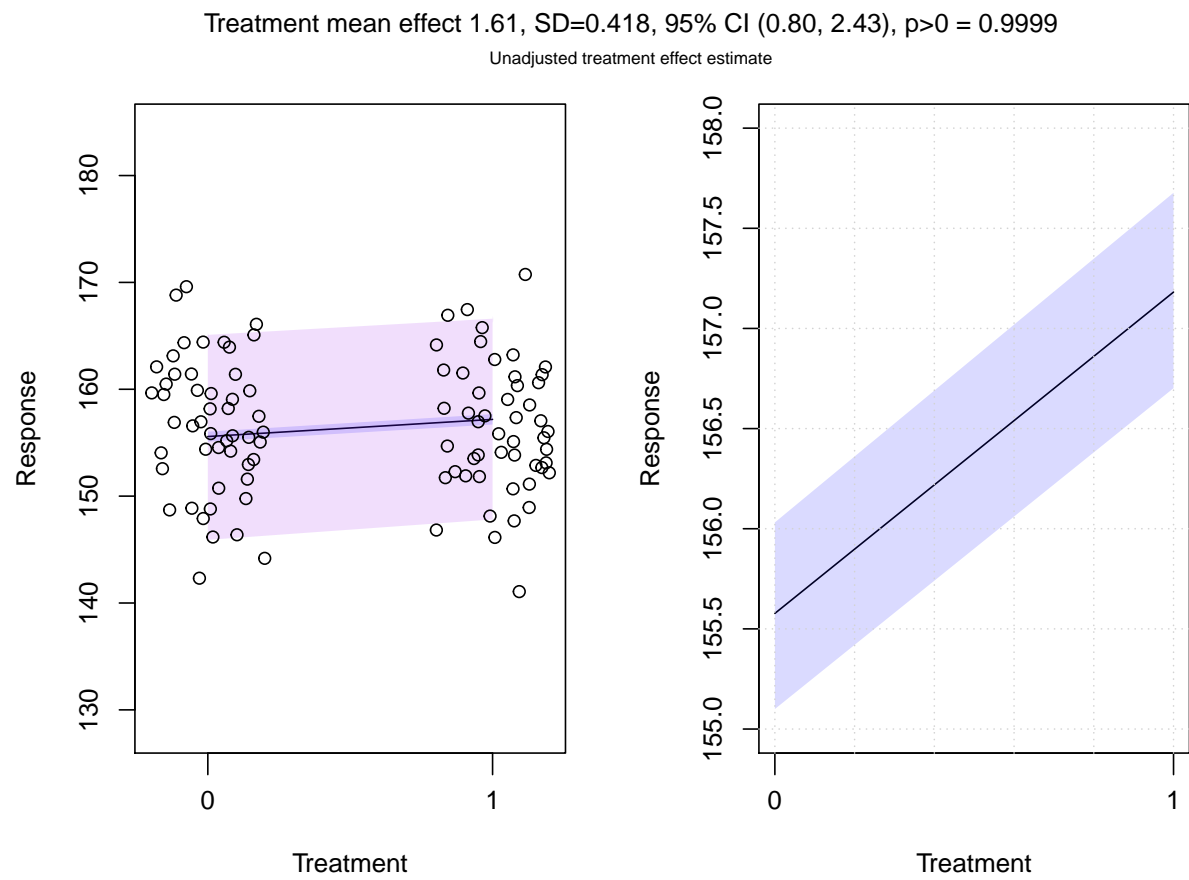


Figure 3: Effects of treatment on outcome, simulated response outcomes left panel, mean response right panel

```

mu1 <- link( m1 , data=pred.data1 )
mu.mean <- apply( mu1 , 2 , mean )
mu.PI <- apply( mu1 , 2 , PI )

# simulate counterfactual response outcomes
R.sim1 <- sim( m1 , data=pred.data1 , n=1e4 )
R.PI <- apply( R.sim1 , 2 , PI )

# Need Y to be the same length as x, these values are not plotted
Y <- seq(from=min( R.sim1, na.rm=T), max(R.sim1, na.rm=T), length.out=nn )

par(mfrow=c(1,2))

# mean and response outcome plot
plot( Y ~ x , type="n", xlab=paste(namez), ylab="Response" , xlim=c(-0.2,1.2 ) ,
      ylim=c(min( R.sim1, na.rm=T)*1, max( R.sim1, na.rm=T)*1), xaxt = "n") # get appropriate
axis(side = 1, at = c(0,1), labels = c(0,1), tck=-.05) # better ticks on x axis
# plot this info
lines( x , mu.mean ) # plot this info
shade( mu.PI , x , col= col.alpha("blue", 0.15))
shade( R.PI , x , col= col.alpha("purple", 0.15))
points(R.sim1[1,] ~ jitter(x)) # pick one of the many simulations to plot

# mean effect
plot(Y ~ x, type="n", xlab=paste(namez), ylab="Response" , ylim=c(152,156), xaxt = "n")
axis(side = 1, at = c(0,1), labels = c(0,1), tck=-.05) # better axes
lines( x , mu.mean )
shade( mu.PI , x , col= col.alpha("blue", 0.15))
grid(nx = NULL, ny = NULL)

# title
mtext( line=-3, text=paste(namez , "effect adjusted to age",
P.age ,"; baseline", p2(P.baseline),"; sex coded", P.sex, sep=" "), outer=T, cex=.7)

# title
foo<-post$b4 # collect the appropriate mcmc samples

mtext( line= -2,
text=paste(namez , " mean effect ", p2(mean(foo , na.rm=T)),", SD=", p3(sd(foo,na.rm=T)) ,", 95%
p2(quantile(foo, c(.025), na.rm=T)), " , ", p2(quantile(foo, c(0.975), na.rm=T)),"), p>0 = ",
mean(foo>0), sep="")
,outer = TRUE, cex = 1)

```

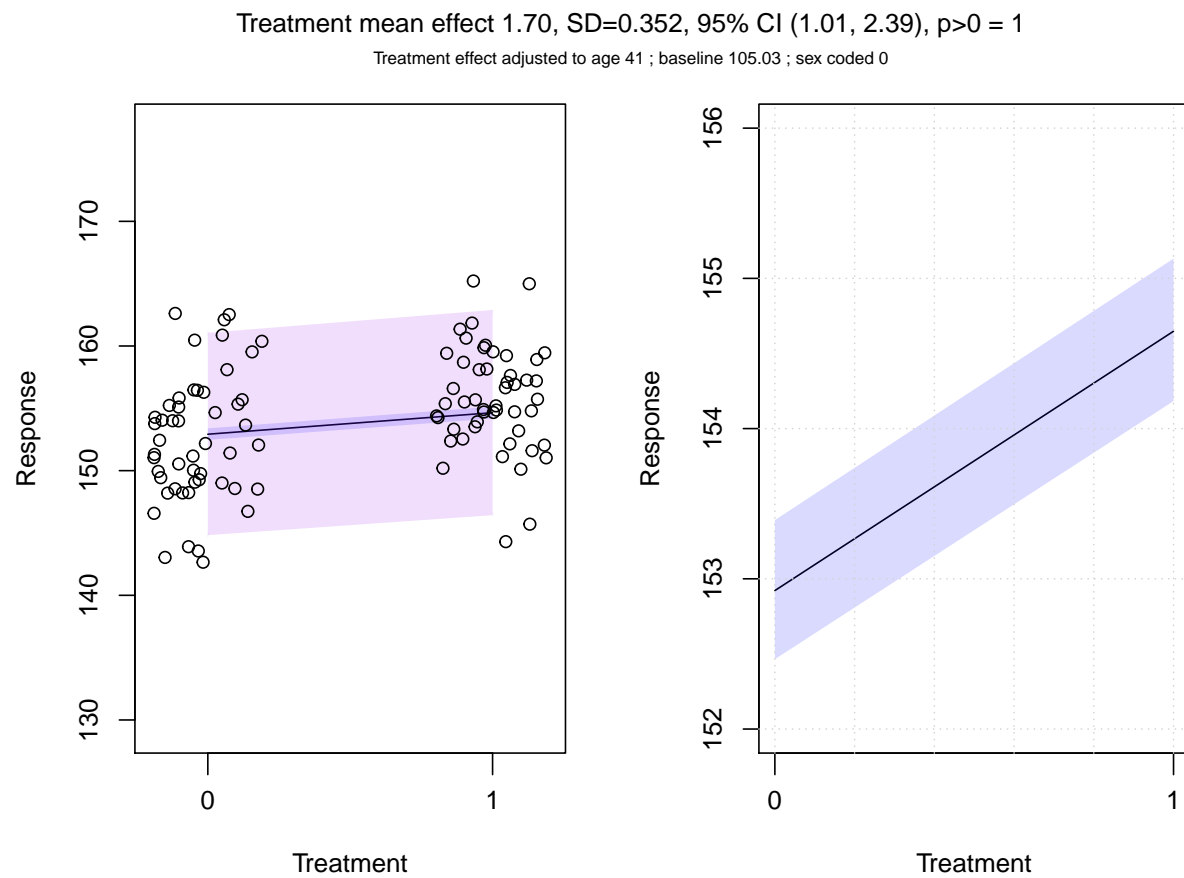


Figure 4: Effects of treatment on outcome, simulated response outcomes left panel, mean response right panel

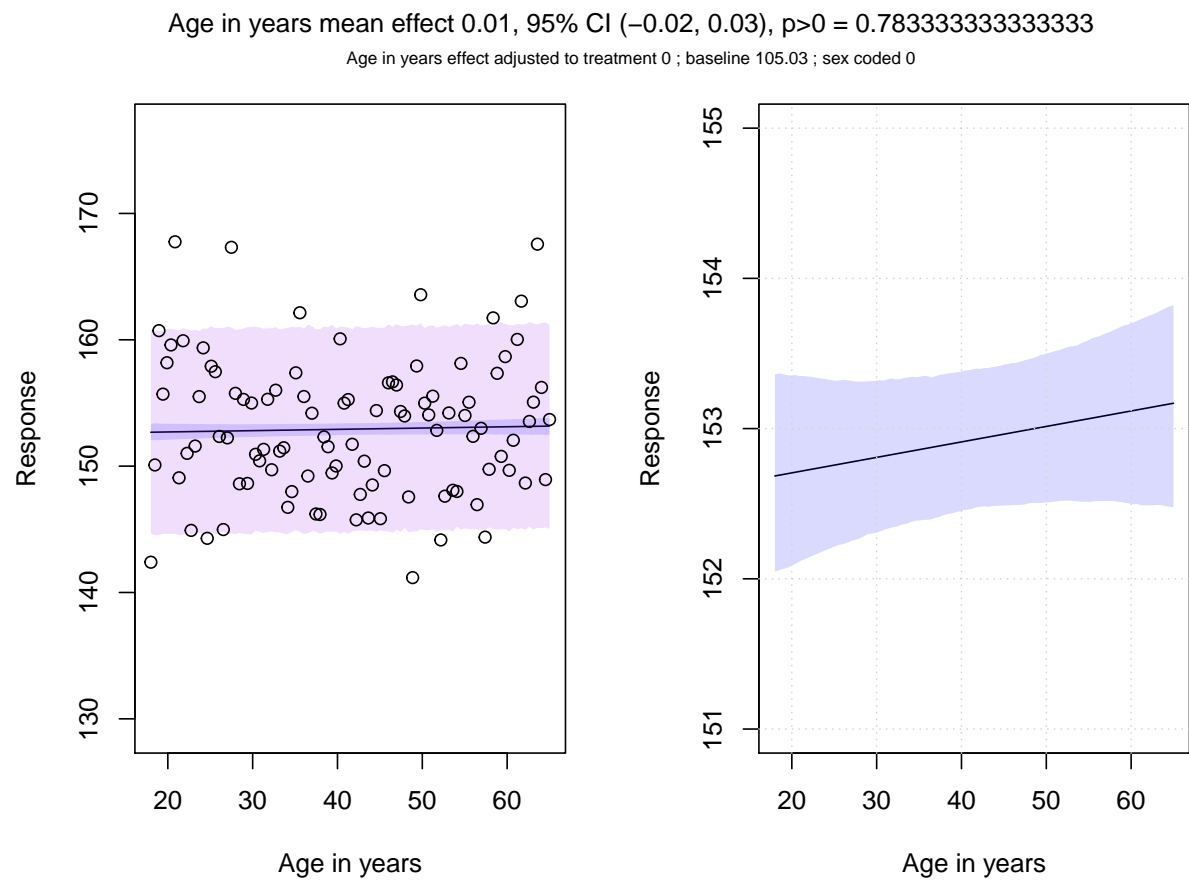


Figure 5: Effects of age on outcome, simulated response outcomes left panel, mean response right panel

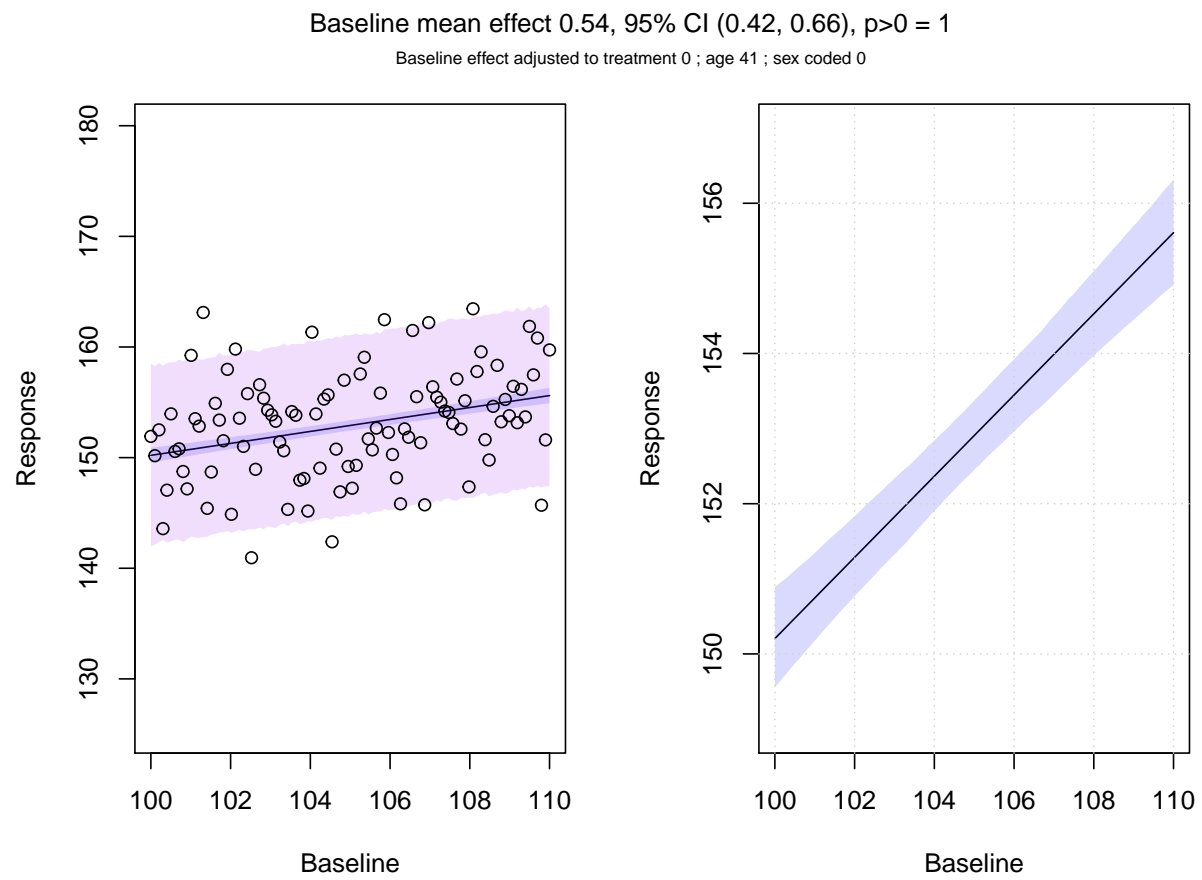
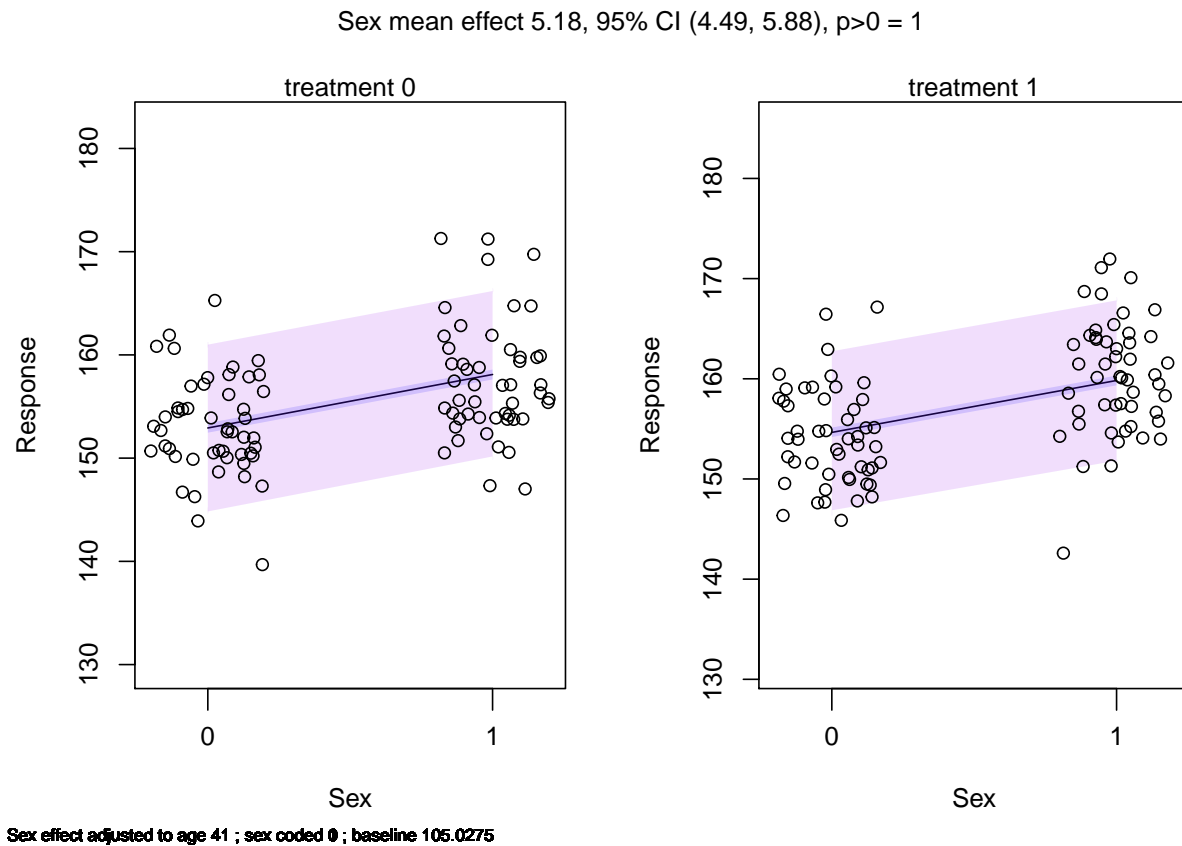
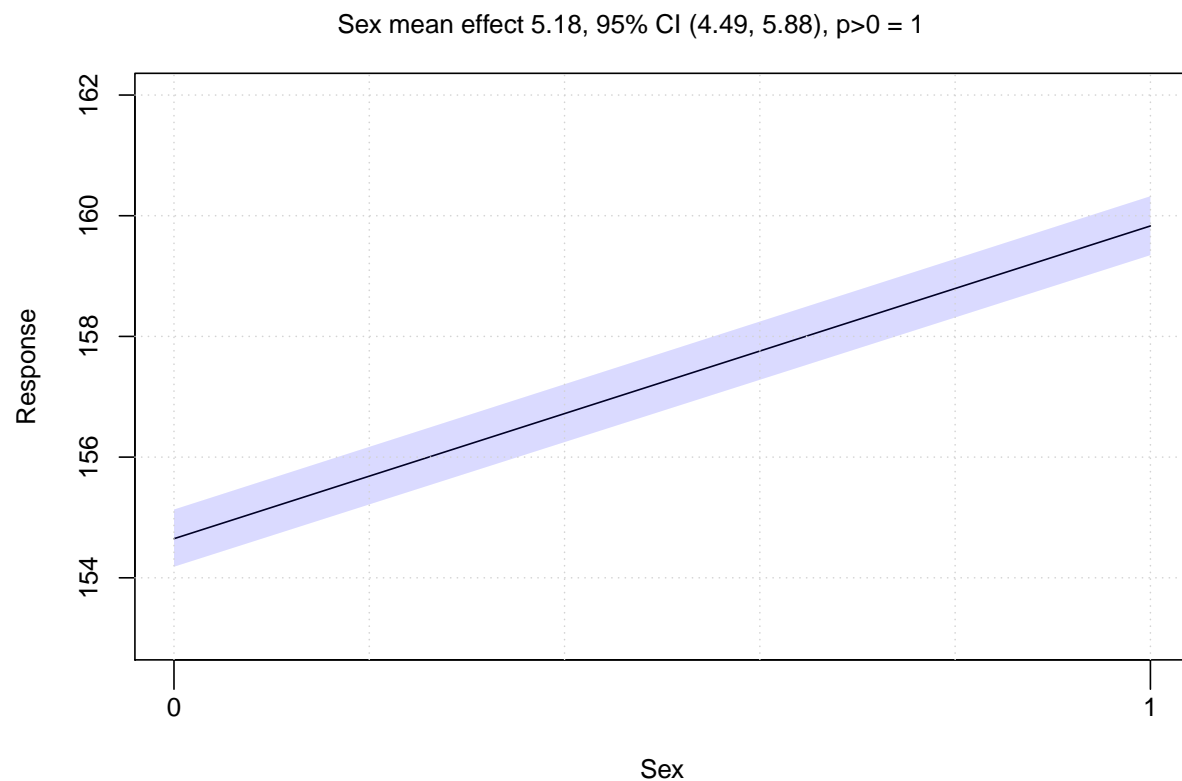


Figure 6: Effects of baseline on outcome, simulated response outcomes left panel, mean response right panel



- 13 For completion the effects of the other covariates on outcome; age
- 14 Baseline effect on outcome
- 15 Effect on outcome; sex. There is no real need to show treatment as well, as no interaction was modelled.





Sex effect adjusted to age 41 ; sex coded 0 ; baseline 105.0275

## 16 Frequentist analysis, print the regression table (also print Bayesian)

```
precis(m1)
```

	Mean	StdDev	lower 0.89	upper 0.89	n_eff	Rhat
b0	95.90	6.64	85.31	106.44	4212	1
b1	0.54	0.06	0.44	0.64	4244	1
b2	5.18	0.35	4.60	5.72	8222	1
b3	0.01	0.01	-0.01	0.03	10026	1
b4	1.70	0.35	1.16	2.28	8191	1
sigma	5.05	0.13	4.85	5.26	7900	1

```
f<- ols( Y ~ baseline + sex + age + treat , d, x=TRUE, y=TRUE)
print(f)
```

Linear Regression Model

```
ols(formula = Y ~ baseline + sex + age + treat, data = d, x = TRUE,
     y = TRUE)
```

		Model Likelihood		Discrimination	
		Ratio Test		Indexes	
Obs	800	LR chi2	263.60	R2	0.281
sigma	5.0481	d.f.	4	R2 adj	0.277
d.f.	795	Pr(> chi2)	0.0000	g	3.626

Residuals

	Min	1Q	Median	3Q	Max
	-15.8808	-3.3450	-0.1489	3.4544	15.8508

	Coef	S.E.	t	Pr(> t )
Intercept	96.3270	6.5606	14.68	<0.0001
baseline	0.5352	0.0621	8.62	<0.0001
sex=M	5.1832	0.3576	14.49	<0.0001
age	0.0098	0.0128	0.76	0.4458
treat	1.7060	0.3572	4.78	<0.0001

## 17 Compute analysis of variance table for the fitted model object with $\chi^2$ statistics.

```
an <- anova(f, main.effect=FALSE, test=c('Chisq')) # try anova(f, test=c('F') ) # F tests
print(an, 'subscripts')
```

			Wald Statistics		Response: Y	
Factor	d.f.	Partial SS	MS	Chi-Square	P	Tested
baseline	1	1891.44228	1891.44228	74.22	<.0001	1
sex	1	5353.62495	5353.62495	210.08	<.0001	2
age	1	14.83008	14.83008	0.58	0.4455	3
treat	1	581.34701	581.34701	22.81	<.0001	4
REGRESSION	4	7906.66760	1976.66690	310.27	<.0001	1-4
ERROR	795	20259.11554	25.48316			

Subscripts correspond to:

```
[1] baseline sex=M age treat
```

## 18 Plot the partial effect of each variable.

```
par(mfrow=c(3,1))
plot(an,
      what=c("proportion chisq"),
      xlab=NULL, pch=16,
      rm.totals=TRUE, rm.ia=FALSE, rm.other=NULL,
      sort=c("descending"), margin=NULL, pl=TRUE,
      trans=NULL, ntrans=40 )

plot(an,
      what=c("chisqminusdf"))

plot(an,
      what=c('proportion R2'))
```

```
par(mfrow=c(1,1))
```

## 19 Compute predicted values and confidence limits

```
plot(Predict(f), anova=an, pval=TRUE)
```

## 20 The key result, computed predicted values and confidence limits for the treatment effect on outcome. Notice the larger SE for the unadjusted analysis.

```
summary(f, treat=c(0,1), est.all=FALSE)
```

	Effects			Response : Y			
Factor	Low	High	Diff.	Effect	S.E.	Lower 0.95	Upper 0.95
treat	0	1	1	1.706	0.35719	1.0049	2.4072

```
ggplot(Predict(f, treat), anova=an, pval=TRUE)
```

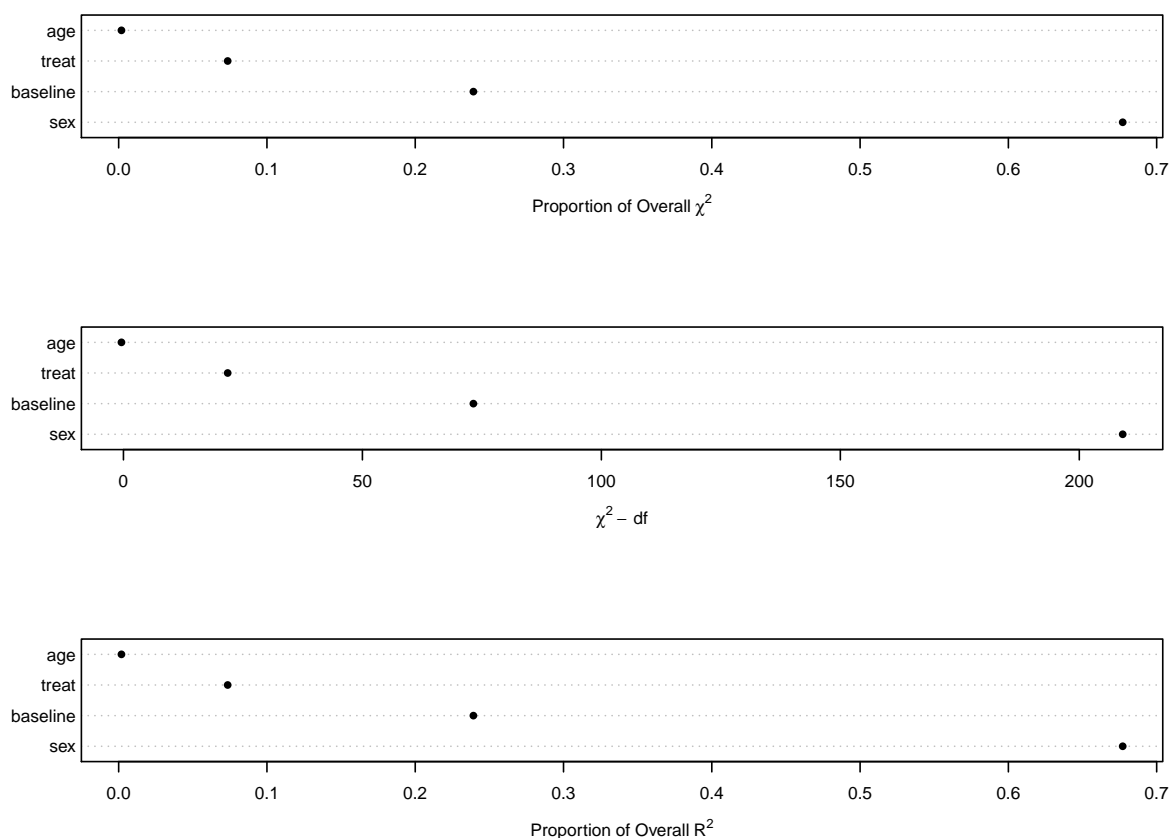


Figure 7: The default for `plot(anova())` is to display the Wald Chi2 statistic minus its degrees of freedom for assessing the partial effect of each variable. Even though this is not scaled  $[0,1]$  it is probably the best method in general because it penalizes a variable requiring a large number of parameters to achieve the Chi2. If a predictor interacts with any other predictor(s), the Chi2 and partial  $R^2$  measures combine the appropriate interaction effects with main effects. For example if the model was  $y \sim \text{biomarker} * \text{treatment}$  the statistic for treatment is the combined effects of treatment as a main effect plus the effect modification that treatment provides for the biomarker effect. This is an assessment of whether there is a difference between the treatment for any biomarker score.

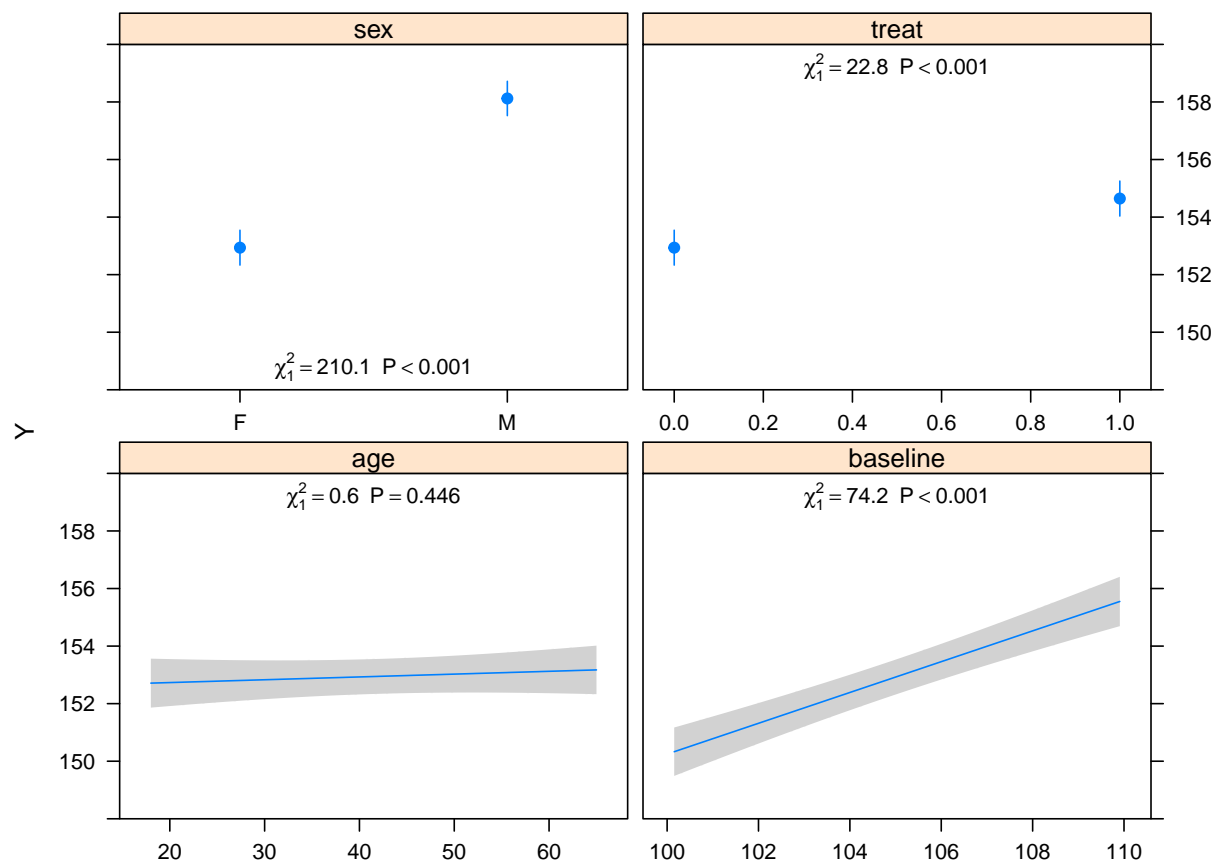
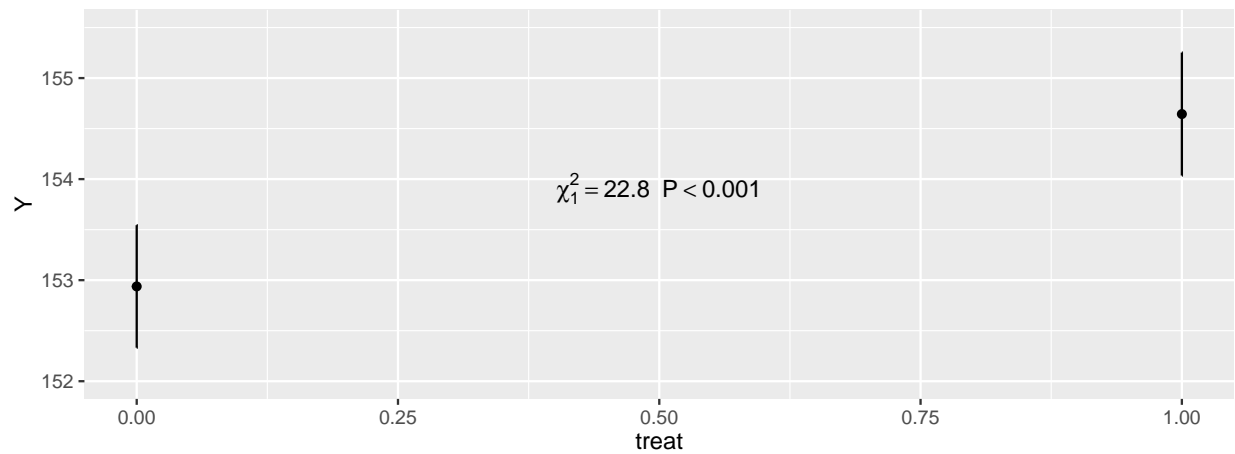


Figure 8: Show predicted values and confidence bands

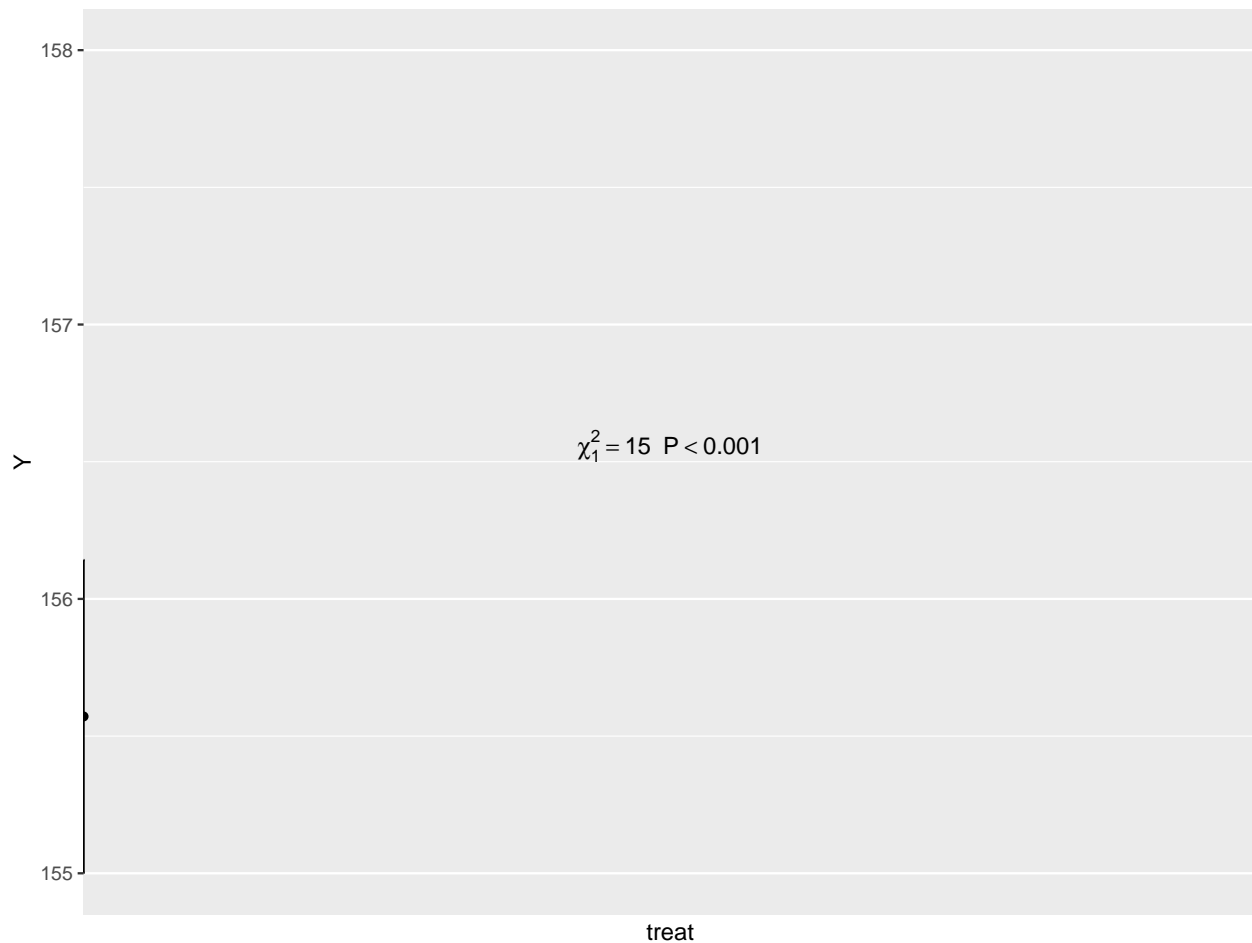


Adjusted to:baseline=105 sex=F age=41

```
#univariate
f0 <- ols( Y ~ treat , d, x=TRUE, y=TRUE)
summary(f0, treat=c(0,1))
```

Effects		Response : Y					
Factor	Low	High	Diff.	Effect	S.E.	Lower 0.95	Upper 0.95
treat	0	1	1	1.6114	0.41636	0.79413	2.4287

```
an0 <- anova(f0, main.effect=FALSE, test=c('Chisq'))
ggplot(Predict(f0), anova=an0, pval=TRUE)
```



**21 Plot the other covariate effects for information. Sex main effect on outcome.**

```
summary(f, sex=c("M", "F"), est.all=FALSE)
```

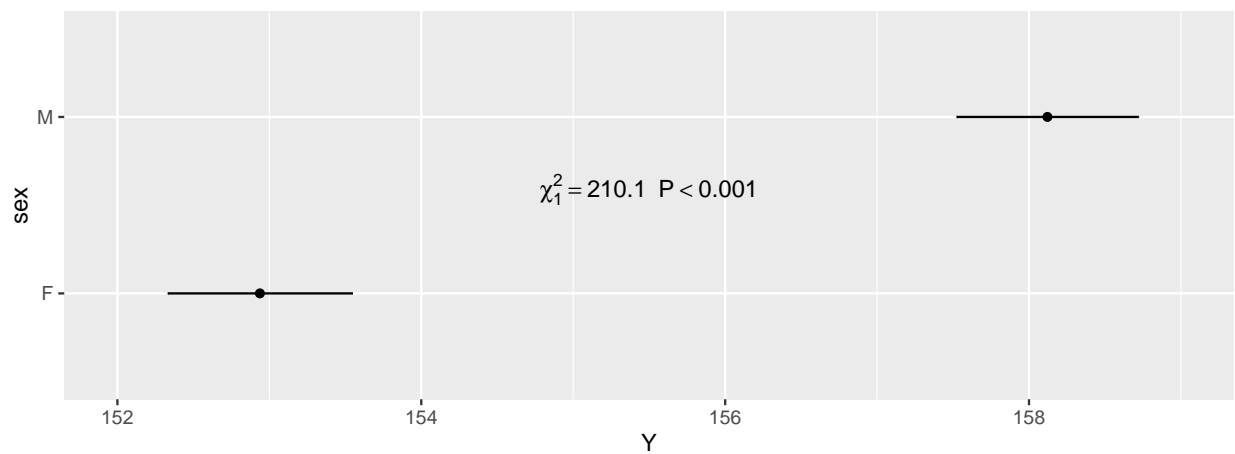
Effects		Response : Y					
Factor	Low	High	Diff.	Effect	S.E.	Lower 0.95	Upper 0.95
sex - M:F	1	2	NA	5.1832	0.3576	4.4813	5.8852

```
ggplot(Predict(f, sex), anova=an, pval=TRUE)
```

**22 Show age main effect on outcome.**

```
summary(f, age=c(18, 65), est.all=FALSE)
```



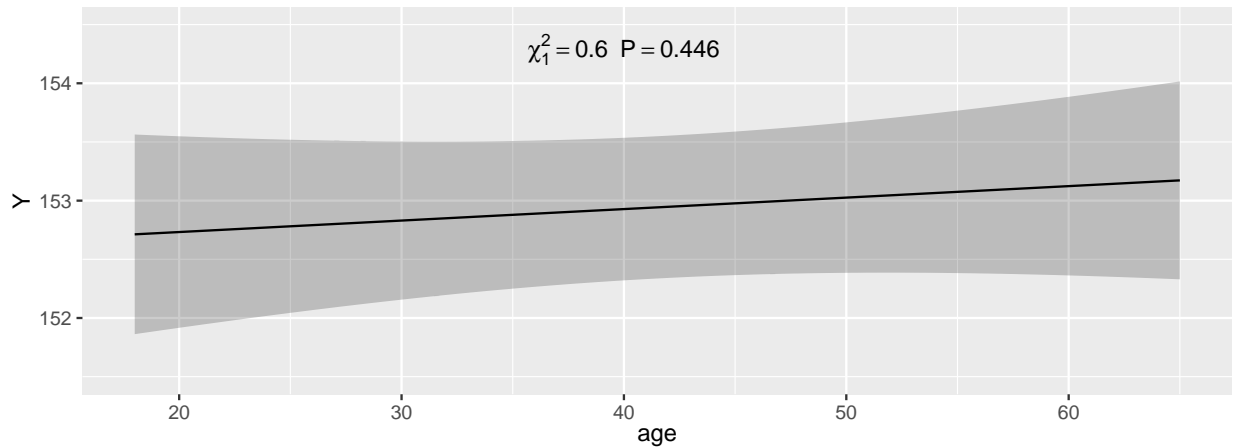


Adjusted to:baseline=105 age=41 treat=0

Figure 9: Sex main effect on outcome

Effects				Response : Y		
Factor	Low	High	Diff.	Effect	S.E.	Lower 0.95 Upper 0.95
age	18	65	47	0.45971	0.60262	-0.7232 1.6426

```
ggplot(Predict(f, age), anova=an, pval=TRUE)
```



Adjusted to:baseline=105 sex=F treat=0

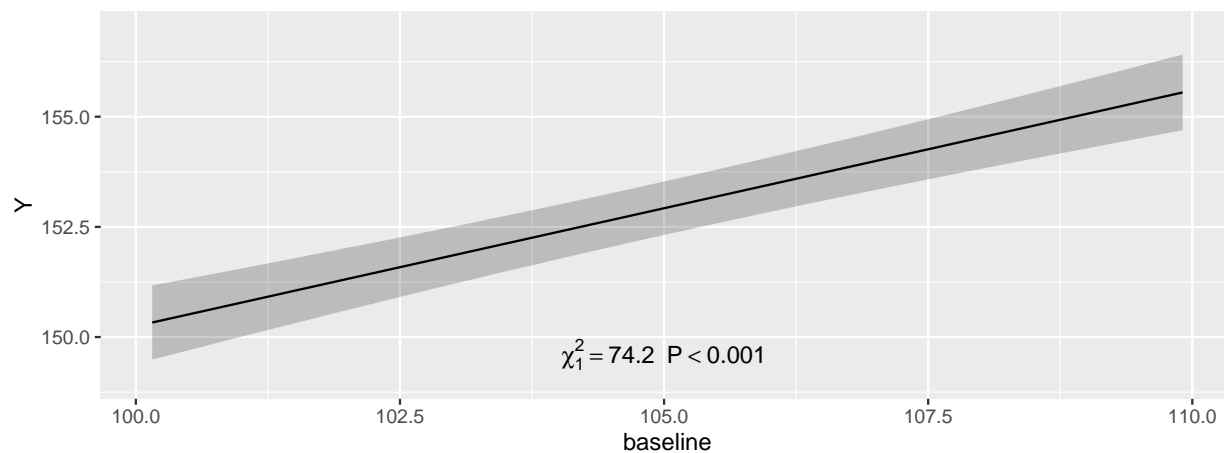
Figure 10: Age main effect on outcome

## 23 Show baseline main effect on outcome.

```
summary(f, baseline=c(-1,1), est.all=FALSE)
```

Effects				Response : Y		
Factor	Low	High	Diff.	Effect	S.E.	Lower 0.95 Upper 0.95
baseline	-1	1	2	1.0704	0.12424	0.8265 1.3143

```
ggplot(Predict(f, baseline), anova=an, pval=TRUE)
```



Adjusted to:sex=F age=41 treat=0

Figure 11: Baseline main effect on outcome

## 24 Reproduce the regression table using the summary function

```
summary(f, baseline=c(0,1), sex=c("M","F") , age=c(0,1), treat=c(0,1) )
```

Effects				Response : Y			
Factor	Low	High	Diff.	Effect	S.E.	Lower 0.95	Upper 0.95
baseline	0	1	1	0.5351900	0.062121	0.413250	0.65713
age	0	1	1	0.0097812	0.012822	-0.015387	0.03495
treat	0	1	1	1.7060000	0.357190	1.004900	2.40720
sex - M:F	1	2	NA	5.1832000	0.357600	4.481300	5.88520

## 25 The contrast function, match the regression table for treatment effect

```
contrast(f,list(treat=1, age=0, baseline=0, sex=c("F")),
         list(treat=0, age=0, baseline=0, sex=c("F")))
```

	age	baseline	sex	Contrast	S.E.	Lower	Upper	t	Pr(> t )
1	0		0 F	1.706045	0.35719	1.004898	2.407192	4.78	0

Error d.f.= 795

Confidence intervals are 0.95 individual intervals

## 26 The contrast function, treatment effect when covariates held at median, however this does not matter as there is no interaction in the model

```
contrast(f,list(treat=1,
               age=median(age), baseline=median(baseline), sex="F"),
         list(treat=0,
               age=median(age), baseline=median(baseline), sex="F"))
```

	age	baseline	sex	Contrast	S.E.	Lower	Upper	t	Pr(> t )
1	41	105.0275	F	1.706045	0.35719	1.004898	2.407192	4.78	0

Error d.f.= 795

Confidence intervals are 0.95 individual intervals

```
contrast(f, list(treat=1),list(treat=0))
```

	baseline	sex	age	Contrast	S.E.	Lower	Upper	t	Pr(> t )
1	105.0275	F	41	1.706045	0.35719	1.004898	2.407192	4.78	0

Error d.f.= 795

Confidence intervals are 0.95 individual intervals

## 27 Assumption check

```
r <- residuals(f)
plot(r); abline(h=0)
```

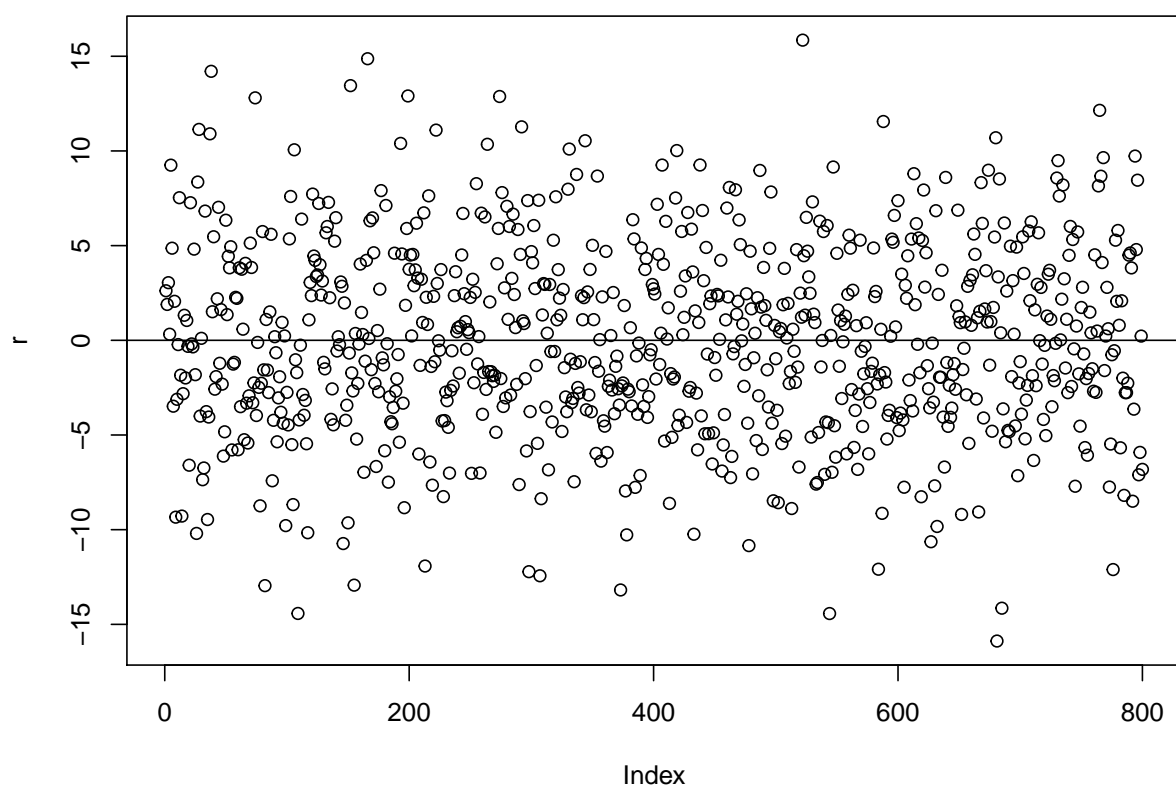


Figure 12: Check OLS assumptions

```
qqnorm(r) ; qqline(r)
```

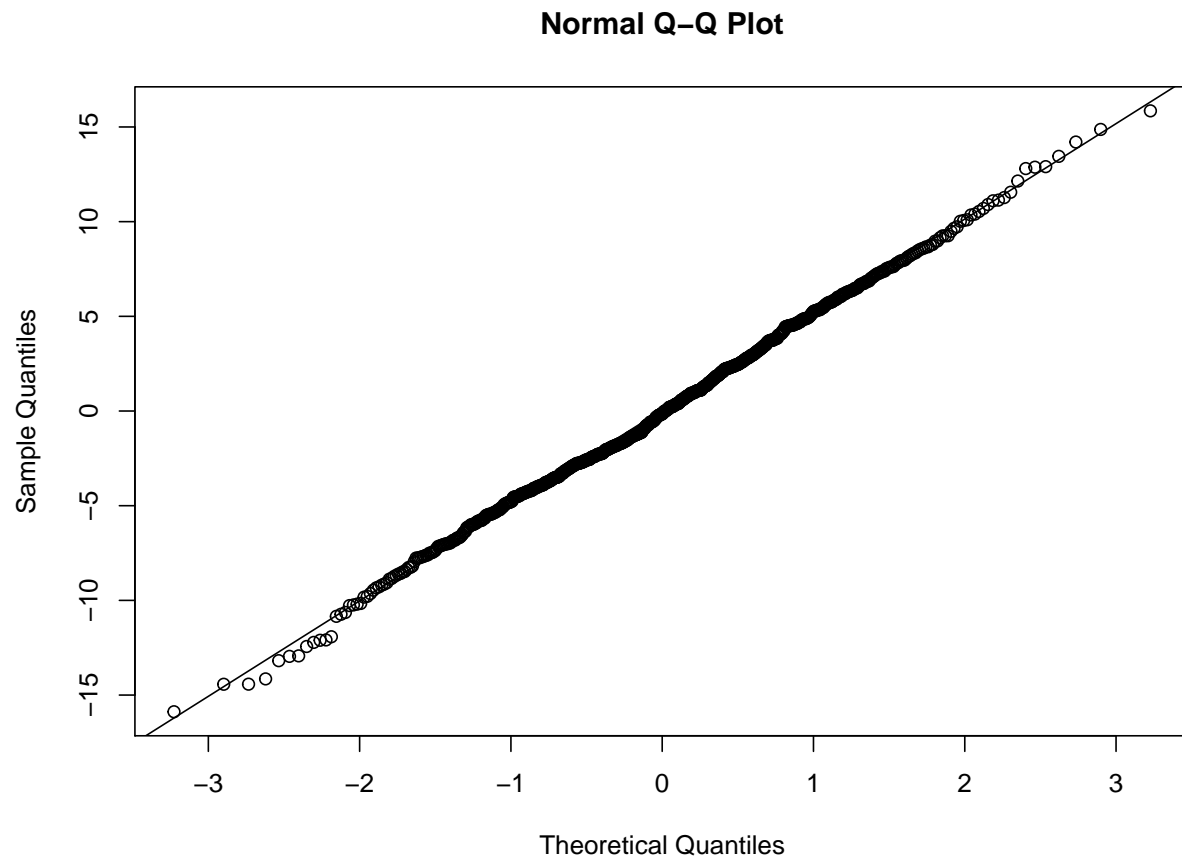


Figure 13: Check OLS assumptions

```
# resid(f, "dfbetas")  
# which.influence(f)
```

## 28 References

Statistical Rethinking 1st Edition, Richard McElreath post treatment bias p151, WAIC p199 and simulate p129

<http://thstatsgeek.com/2014/02/01/adjusting-for-baseline-covariates-in-randomized-controlled-trials/>  
<http://egap.org/content/power-analysis-simulations-r>

<http://stackoverflow.com/questions/14554558/simulate-a-linear-model-100-times>

<http://stats.stackexchange.com/questions/155246/which-variable-relative-importance-method-to-use>

## 29 Computing Environment

R version 3.2.2 (2015-08-14)

Platform: x86\_64-w64-mingw32/x64 (64-bit)

Running under: Windows 8 x64 (build 9200)

locale:

[1] LC\_COLLATE=English\_United Kingdom.1252 LC\_CTYPE=English\_United Kingdom.1252

[3] LC\_MONETARY=English\_United Kingdom.1252 LC\_NUMERIC=C

[5] LC\_TIME=English\_United Kingdom.1252

attached base packages:

[1] grid parallel stats graphics grDevices utils datasets methods base

other attached packages:

[1] VennDiagram\_1.6.17 futile.logger\_1.4.3 rethinking\_1.58 rstan\_2.10.1 StanHeaders\_2.1.0

[6] reshape\_0.8.5 rms\_4.5-0 SparseM\_1.7 Hmisc\_3.17-4 ggplot2\_2.1.0

[11] Formula\_1.2-1 survival\_2.39-5 lattice\_0.20-33 knitr\_1.13

loaded via a namespace (and not attached):

[1] zoo\_1.7-13 reshape2\_1.4.1 splines\_3.2.2 colorspace\_1.2-6 htmltools\_0.3.5

[6] stats4\_3.2.2 loo\_0.1.6 yaml\_2.1.13 chron\_2.3-47 foreign\_0.8-66

[11] RColorBrewer\_1.1-2 lambda.r\_1.1.9 matrixStats\_0.50.2 multcomp\_1.4-6 plyr\_1.8.4

[16] stringr\_1.0.0 MatrixModels\_0.4-1 munsell\_0.4.3 gtable\_0.2.0 mvtnorm\_1.0-5

[21] codetools\_0.2-14 coda\_0.18-1 evaluate\_0.9 labeling\_0.3 latticeExtra\_0.15-1

[26] inline\_0.3.14 quantreg\_5.26 TH.data\_1.0-7 Rcpp\_0.12.6 KernSmooth\_2.23-1

[31] acepack\_1.3-3.3 scales\_0.4.0 formatR\_1.4 gridExtra\_2.2.1 digest\_0.6.9

[36] stringi\_1.1.1 polspline\_1.1.12 tools\_3.2.2 sandwich\_2.3-4 magrittr\_1.5

[41] futile.options\_1.0.0 cluster\_2.0.3 MASS\_7.3-45 Matrix\_1.2-2 data.table\_1.9.6

[46] rmarkdown\_1.0 rpart\_4.1-10 nnet\_7.3-12 nlme\_3.1-128

[1] "~/GIT\\programs-master"

This took 744.85 seconds to execute.