

Solution 1

```
# barley data fits
library(SMPRACTICALS)
library(mgcv)
data(barley)
x <- barley$Location
y <- barley$y
z <- barley$Variety

#ML fit of variety as fixed effect, with GCV estimation of lambdas,
fit.gcv <- gam(y~Variety-1+s(Location,by=Block),data=barley)
summary(fit.gcv)
# For more details help(summary.gam)
# or check section 6 of
fit.plot.gcv <- plot(fit.gcv,seWithMean=TRUE)
fit.gcv$aic

#Plot residuals
par(mfrow=c(1,3))

linearShift <- 4; shift <- 10

plot(x[1:75],y[1:75],ylim=c(7.5,13.5+2*linearShift),type="n",ylab="Yield y", xlab="Location x",xlim=c(0,75))# ,yaxt="n")

lines(x[1:75],y[1:75],type="s")
lines(x[76:150],y[76:150]+linearShift,type="s")
lines(x[151:225],y[151:225]+2*linearShift,type="s")

plot(x[1:75],y[1:75],ylim=c(7.5,13.5+2*4),type="n",ylab="Yield y",
      xlab="Location x",xlim=c(0,75))# ,yaxt="n")
lines(fit.plot.gcv[[1]]$x,shift+fit.plot.gcv[[1]]$fit)
lines(fit.plot.gcv[[1]]$x,shift+fit.plot.gcv[[1]]$fit+2*fit.plot.gcv[[1]]$se,lty=2)
lines(fit.plot.gcv[[1]]$x,shift+fit.plot.gcv[[1]]$fit-2*fit.plot.gcv[[1]]$se,lty=2)

lines(fit.plot.gcv[[2]]$x,shift+fit.plot.gcv[[2]]$fit+linearShift)
lines(fit.plot.gcv[[2]]$x,shift+fit.plot.gcv[[2]]$fit+2*fit.plot.gcv[[2]]$se+linearShift,lty=2)
lines(fit.plot.gcv[[2]]$x,shift+fit.plot.gcv[[2]]$fit-2*fit.plot.gcv[[2]]$se+linearShift,lty=2)

lines(fit.plot.gcv[[3]]$x,shift+fit.plot.gcv[[3]]$fit+2*linearShift)
lines(fit.plot.gcv[[3]]$x,shift+fit.plot.gcv[[3]]$fit+2*fit.plot.gcv[[3]]$se+2*linearShift,lty=2)
lines(fit.plot.gcv[[3]]$x,shift+fit.plot.gcv[[3]]$fit-2*fit.plot.gcv[[3]]$se+2*linearShift,lty=2)

res <- residuals(fit.gcv)
res <- c(res[1:187],NA,res[188:224])

plot(x[1:75],y[1:75],ylim=c(7.5,13.5+2*4),type="n",ylab="Yield y",
      xlab="Location x",xlim=c(0,75))# ,yaxt="n")
# abline(h=8:13,lwd=0.1,col=2)
lines(1:75,shift+res[1:75],type="s")
lines(1:75,shift+res[75+c(1:75)]+linearShift,type="s")
lines(1:75,shift+res[150+c(1:75)]+2*linearShift,type="s")

#Include random effect on variety
fit.gcv.re <- gam(y~s(Variety,bs="re")+s(Location,by=Block),data=barley)
summary(fit.gcv.re)

gam.vcomp(fit.gcv.re)
fit.gcv.re$aic

#Plot the residuals
par(mfrow=c(1,2))

res <- residuals(fit.gcv)
res <- c(res[1:187],NA,res[188:224])

plot(x[1:75],y[1:75],ylim=c(7.5,13.5+2*4),type="n",ylab="Yield y",
      xlab="Location x",xlim=c(0,75))# ,yaxt="n")
```

```

# abline(h=8:13,lwd=0.1,col=2)
lines(1:75,shift+res[1:75],type="s")
lines(1:75,shift+res[75+c(1:75)]+linearShift,type="s")
lines(1:75,shift+res[150+c(1:75)]+2*linearShift,type="s")

res <- residuals(fit.gcv.re)
res <- c(res[1:187],NA,res[188:224])

plot(x[1:75],y[1:75],ylim=c(7.5,13.5+2*4),type="n",ylab="Yield y",
      xlab="Location x",xlim=c(0,75))# ,yaxt="n")
# abline(h=8:13,lwd=0.1,col=2)
lines(1:75,shift+res[1:75],type="s")
lines(1:75,shift+res[75+c(1:75)]+linearShift,type="s")
lines(1:75,shift+res[150+c(1:75)]+2*linearShift,type="s")

#Plot estimated effects
par(mfrow=c(1,1))
plot(fit.gcv$coefficients[1:75], fit.gcv.re$coefficients[1] + fit.gcv.re$coefficients[2:76])
abline(0,1)

#Clear plots
dev.off()

#FIT GAM USING reml
fit.reml <- gam(y~Variety-1+s(Location,by=Block),method="REML",data=barley)

fit.reml
summary(fit.reml)
plot.fit.reml <- plot(fit.reml,seWithMean=TRUE)
fit.reml$aic

par(mfrow=c(1,3))

plot(x[1:75],y[1:75],ylim=c(7.5,13.5+2*linearShift),type="n",ylab="Yield y", xlab="Location x",xlim=c(0,75))# ,yaxt="n")

lines(x[1:75],y[1:75],type="s")
lines(x[76:150],y[76:150]+linearShift,type="s")
lines(x[151:225],y[151:225]+2*linearShift,type="s")

linearShift <- 4; shift <- 10

plot(x[1:75],y[1:75],ylim=c(7.5,13.5+2*4),type="n",ylab="Yield y",
      xlab="Location x",xlim=c(0,75))# ,yaxt="n")
lines(plot.fit.reml[[1]]$x,shift+plot.fit.reml[[1]]$fit)
lines(plot.fit.reml[[1]]$x,shift+plot.fit.reml[[1]]$fit+2*plot.fit.reml[[1]]$se,lty=2)
lines(plot.fit.reml[[1]]$x,shift+plot.fit.reml[[1]]$fit-2*plot.fit.reml[[1]]$se,lty=2)

lines(plot.fit.reml[[2]]$x,shift+plot.fit.reml[[2]]$fit+linearShift)
lines(plot.fit.reml[[2]]$x,shift+plot.fit.reml[[2]]$fit+2*plot.fit.reml[[2]]$se+linearShift,lty=2)
lines(plot.fit.reml[[2]]$x,shift+plot.fit.reml[[2]]$fit-2*plot.fit.reml[[2]]$se+linearShift,lty=2)

lines(plot.fit.reml[[3]]$x,shift+plot.fit.reml[[3]]$fit+2*linearShift)
lines(plot.fit.reml[[3]]$x,shift+plot.fit.reml[[3]]$fit+2*plot.fit.reml[[3]]$se+2*linearShift,lty=2)
lines(plot.fit.reml[[3]]$x,shift+plot.fit.reml[[3]]$fit-2*plot.fit.reml[[3]]$se+2*linearShift,lty=2)

res <- residuals(fit.reml)
res <- c(res[1:187],NA,res[188:224])

plot(x[1:75],y[1:75],ylim=c(7.5,13.5+2*4),type="n",ylab="Yield y",
      xlab="Location x",xlim=c(0,75))# ,yaxt="n")
# abline(h=8:13,lwd=0.1,col=2)
lines(1:75,shift+res[1:75],type="s")
lines(1:75,shift+res[75+c(1:75)]+linearShift,type="s")
lines(1:75,shift+res[150+c(1:75)]+2*linearShift,type="s")

#RANDOM YIELD SELECTION

#Include random effect on variety
fit.reml.re <- gam(y~s(Variety,bs="re")+s(Location,by=Block),data=barley, method="REML")
fit.reml.re$aic

#Plot the residuals
par(mfrow=c(1,2))

res <- residuals(fit.reml)

```

```

res <- c(res[1:187],NA,res[188:224])

plot(x[1:75],y[1:75],ylim=c(7.5,13.5+2*4),type="n",ylab="Yield y",
      xlab="Location x",xlim=c(0,75))# ,yaxt="n")
# abline(h=8:13,lwd=0.1,col=2)
lines(1:75,shift+res[1:75],type="s")
lines(1:75,shift+res[75+c(1:75)]+linearShift,type="s")
lines(1:75,shift+res[150+c(1:75)]+2*linearShift,type="s")

res <- residuals(fit.reml.re)
res <- c(res[1:187],NA,res[188:224])

plot(x[1:75],y[1:75],ylim=c(7.5,13.5+2*4),type="n",ylab="Yield y",
      xlab="Location x",xlim=c(0,75))# ,yaxt="n")
# abline(h=8:13,lwd=0.1,col=2)
lines(1:75,shift+res[1:75],type="s")
lines(1:75,shift+res[75+c(1:75)]+linearShift,type="s")
lines(1:75,shift+res[150+c(1:75)]+2*linearShift,type="s")

#Plot estimated effects
par(mfrow=c(1,1))
plot(fit.reml$coefficients[1:75], fit.reml.re$coefficients[1] + fit.reml.re$coefficients[2:76])
abline(0,1)

#Clear plots
dev.off()

```