```
data("Caterpillars")
FGP<- subset(Caterpillars, Fgp=="Y")
plot(Nassim~Mass, data=FGP)

#Different transformations
plot(LogNassim~LogMass, data=FGP)
plot(LogNassim^2~LogMass, data=FGP)
plot(sqrt(Nassim)~sqrt(Mass), data=FGP)

#It's not obvious that any of these is "best".
#Fit the different models to look at residuals:
log.lm <-lm(LogNassim~LogMass, data=FGP)
plot(log.lm)

log2.lm <-lm(LogNassim^2~LogMass, data=FGP)
plot(log2.lm)

FGP$sqrtNassim <- sqrt(FGP$Nassim); FGP$sqrtMass <- sqrt(FGP$Mass)
sqrt.lm <- lm(sqrtNassim~sqrtMass, data=FGP)
plot(sqrt.lm)

#Inference with the sqrt model:
cor.test(sqrtNassim~ sqrtMass, data=FGP)
summary(sqrt.lm)
anova(sqrt.lm)

#Intervals:
#confidence interval for the slope:
confint(sqrt.lm)
#confidence interval for mean response when Mass=2:
predict.lm(sqrt.lm, data.frame("sqrtMass"=sqrt(2)),interval="confidence")
#prediction interval for a particular response when Mass=2:
predict.lm(sqrt.lm, data.frame("sqrtMass"=sqrt(2)),interval="prediction")


#R code for re-making Figure 2.2: confidence and prediction interval bands for a simple
linear regression

#Model (notice that you must call for the X matrix)
model.lm <- lm(sqrtNassim~sqrtMass, data=FGP, x=TRUE)

#Calculate confidence and prediction bounds
ConfBounds <- predict.lm(model.lm, interval="confidence")
PredBounds <- predict.lm(model.lm, interval="prediction")

#Plot data
plot(sqrtNassim~sqrtMass, data=FGP)

#Put everything in order by X:
Bounds <- cbind(model.lm$x[,2], PredBounds, ConfBounds)
Bounds <- Bounds[sort.list(Bounds[,1]),]

#Plot bounds
lines(Bounds[,1], Bounds[,3],col=2, lty=2) #lwr PredBounds
lines(Bounds[,1], Bounds[,4],col=2, lty=2) #upr PredBounds
lines(Bounds[,1], Bounds[,6],col=3, lty=3) #lwr ConfBounds
lines(Bounds[,1], Bounds[,7],col=3, lty=3) #upr ConfBounds
```