Traditional and Modern Approaches to Modelling with R: An Advanced Course

Bill Venables,
CSIRO, Cleveland, Australia

2007

Session 01

R and Modelling overview

Statistical modelling: a standpoint and overview

- Typical modelling situation:
  - Response variable Y
  - Predictor variables $x_1, x_2, x_3, \ldots, x_p$
  - Unavoidable random variation: $Z \sim \mathrm{N}(0,1)$
  - conceptual model:

$$Y = f(\mathbf{x}, Z), \text{ where } \mathbf{x} = (x_1, x_2, \ldots, x_p)$$

- Let $\mathbf{x_0}$ be some point in the range of the data near which we would like to approximate the function.
- Assuming smoothness, we may do so by the first few terms in a power series expansion

Local approximations

- First order approximation:

$$Y_i = \beta_0 + \sum_{j=1}^{p} \beta_j (x_{ij} - x_{0j}) + \sigma(Z - 0)$$

- Second order approximation:

$$Y_i = \beta_0 + \sum_{j=1}^{p} \beta_j (x_{ij} - x_{0j}) + \sum_{j=k}^{p} \sum_{k=1}^{p} \beta_{jk} (x_{ij} - x_{0j})(x_{ik} - x_{0k})$$
$$+ \sum_{j=1}^{p} \left(\sigma_0 + \sigma_j (x_{ij} - x_{0j})\right)(Z - 0) + \delta(Z - 0)^2$$

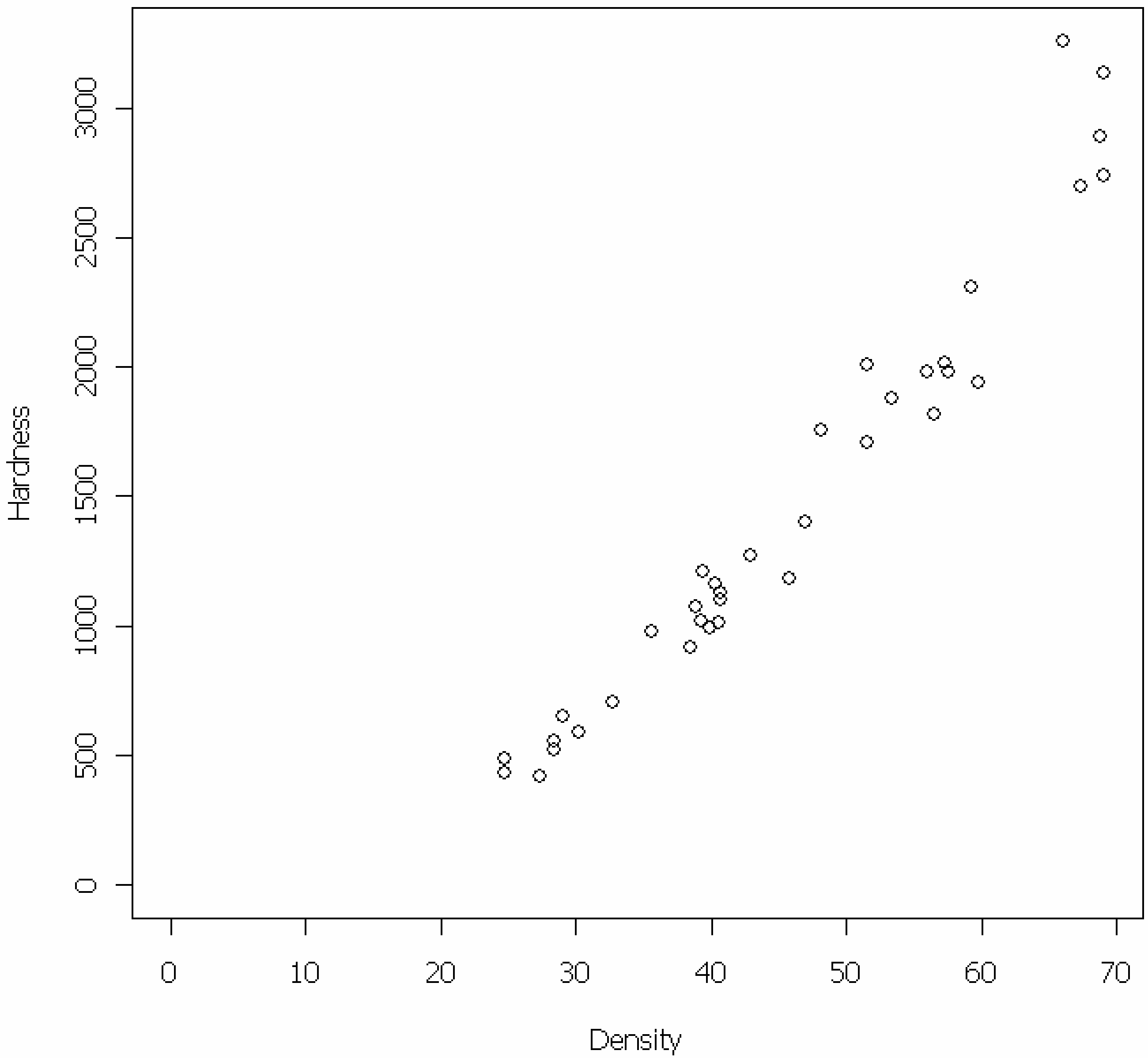
Speculative conclusions

- Regression models usually have an empirical justification as a local approximation to the response curve or surface
- They may only be useful in a close range about the observed data
- Extending the range of usefulness for the approximation may require some allowance for
  - Second (or higher) degree curvature terms
  - Interactions between variables
  - Non-normality of the error structure (skewness/kurtosis)
  - A great deal more data than is available!?

An Example: Janka hardness data

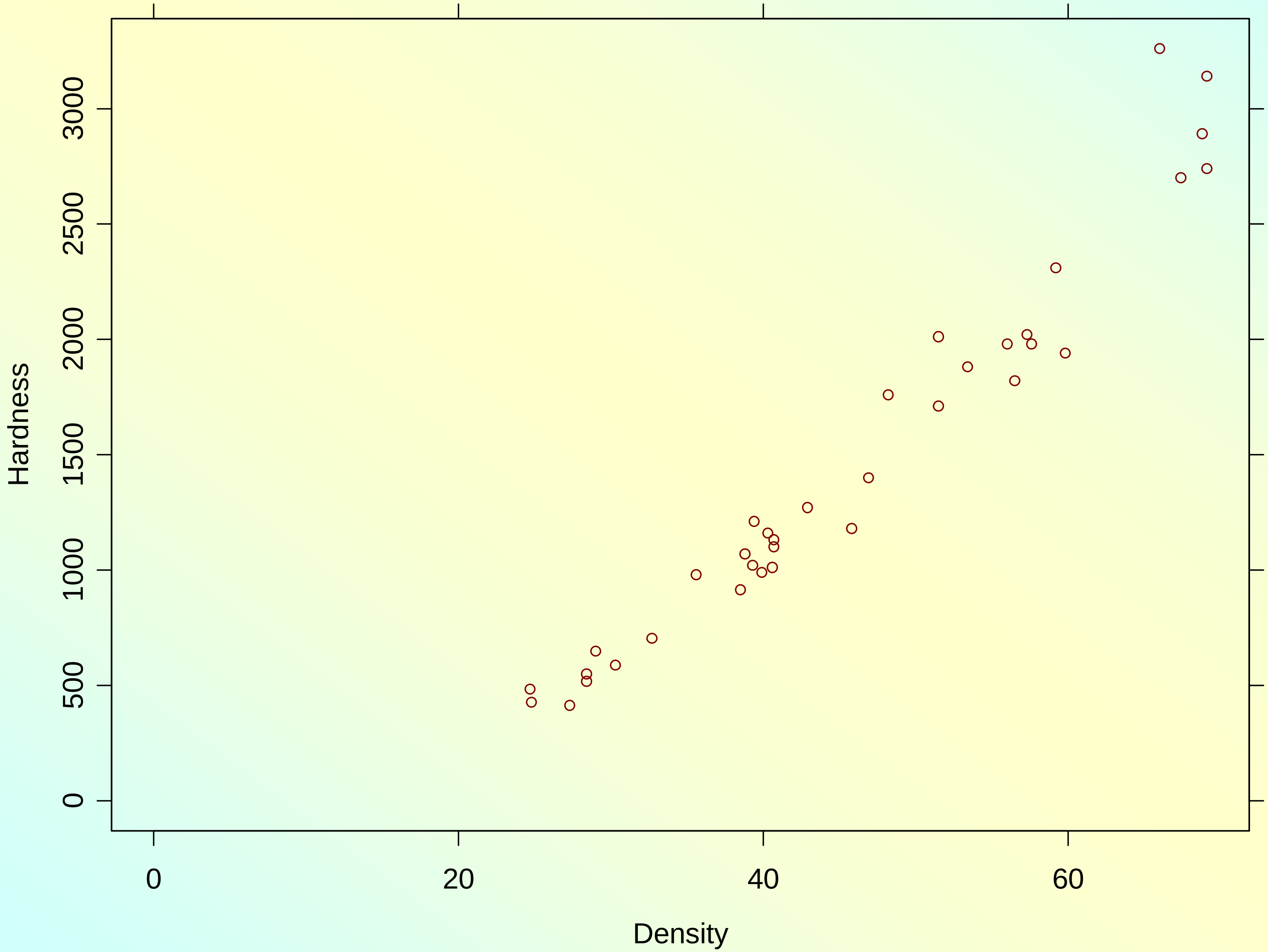
	Density	Hardness		Density	Hardness
1	24.7	484	19	42.9	1270
2	24.8	427	20	45.8	1180
3	27.3	413	21	46.9	1400
4	28.4	517	22	48.2	1760
5	28.4	549	23	51.5	1710
6	29.0	648	24	51.5	2010
7	30.3	587	25	53.4	1880
8	32.7	704	26	56.0	1980
9	35.6	979	27	56.5	1820
10	38.5	914	28	57.3	2020
11	38.8	1070	29	57.6	1980
12	39.3	1020	30	59.2	2310
13	39.4	1210	31	59.8	1940
14	39.9	989	32	66.0	3260
15	40.3	1160	33	67.4	2700
16	40.6	1010	34	68.8	2890
17	40.7	1100	35	69.1	2740
18	40.7	1130	36	69.1	3140

```
with(janka, plot(Density, Hardness,
  xlim = range(0, Density), ylim = range(0, Hardness),
  las=0))
```


Hardness
Density
0
500
1000
1500
2000
2500
3000
0
10
20
30
40
50
60
70

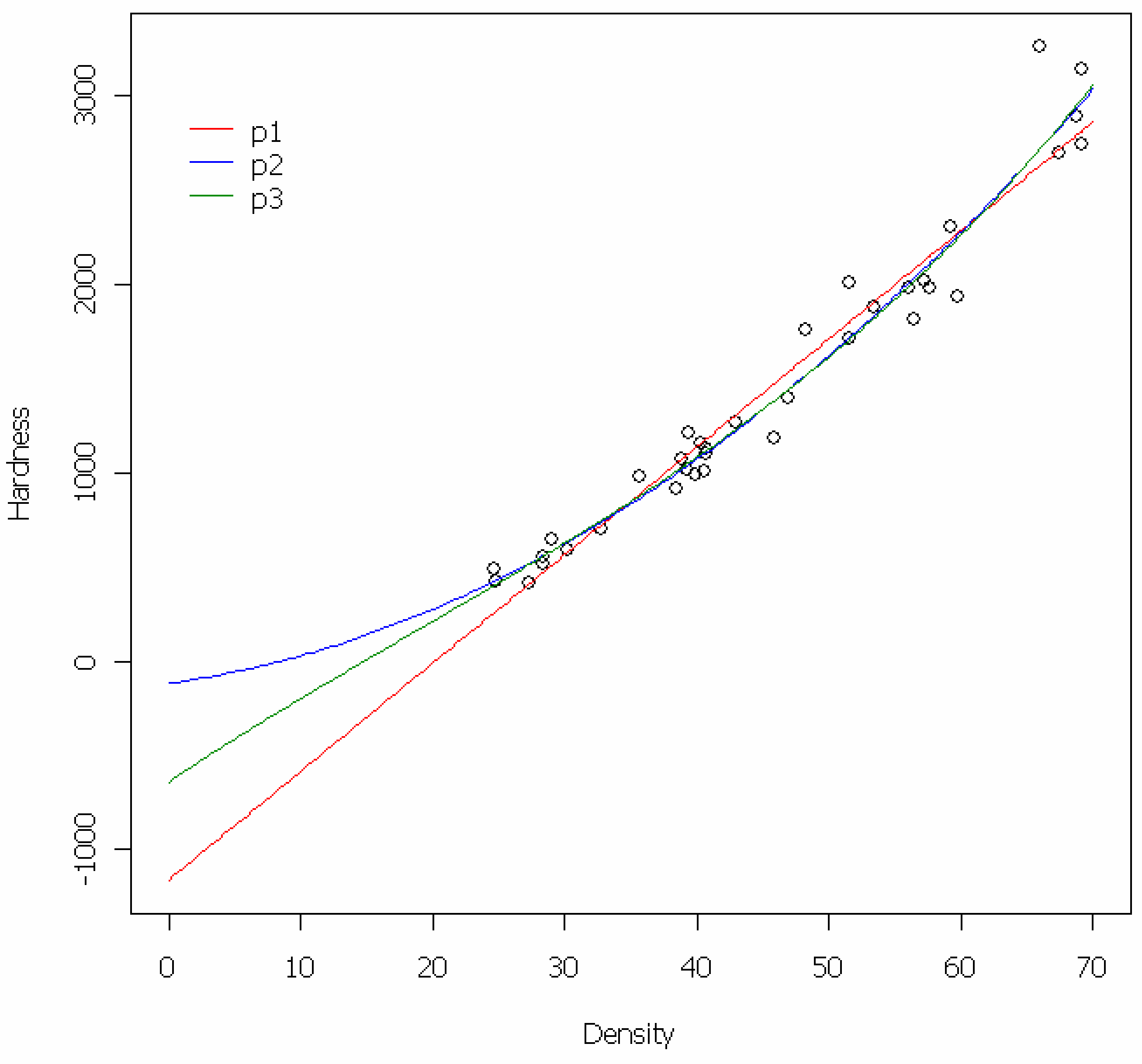
Some initial explorations

```
jank.1 <- lm(Hardness ~ Density, janka)
jank.2 <- update(jank.1, .~.+I(Density^2))
jank.3 <- update(jank.2, .~.+I(Density^3))

Janka <- rbind(janka,
   data.frame(Density = seq(0, 70, length=200),
               Hardness = NA))

Janka <- Janka[order(Janka$Density), ]  # ordered densities
Janka <- transform(Janka,
               p1 = predict(jank.1, Janka),
               p2 = predict(jank.2, Janka),
               p3 = predict(jank.3, Janka))
```


Hardness
0
500
1000
1500
2000
2500
3000
0
20
40
60
Density

Plotting the results

```
with(Janka, {
  rg <- range(Hardness, p1, p2, p3, na.rm = T)
  plot(Density, Hardness, ylim = rg)
  lines(Density, p1, col="red")
  lines(Density, p2, col="blue")
  lines(Density, p3, col="green4")
  legend(0, 3000, paste("p", 1:3, sep=""),
  lty = 1, col = c("red", "blue", "green4"),
  bty = "n")
})
```



p1
p2
p3
Hardness
3000
2000
1000
0
-1000
0
10
20
30
40
50
60
70
Density

The stability of coefficients

```
x <- c(mean(janka$Hard), coef(jank.1),
            coef(jank.2), coef(jank.3))
w <- matrix(0, 4, 4)
w[!lower.tri(w)] <- x
dimnames(w) <- list(paste("Degree", 0:3),
                  paste("Model", 0:3))
round(t(w), 5)
```

```
          Degree 0 Degree 1 Degree 2 Degree 3
Model 0  1469.4722  0.00000  0.00000  0.00000
Model 1 -1160.4997 57.50667  0.00000  0.00000
Model 2  -118.0074  9.43402  0.50908  0.00000
Model 3  -641.4379 46.86373 -0.33117  0.00596
```

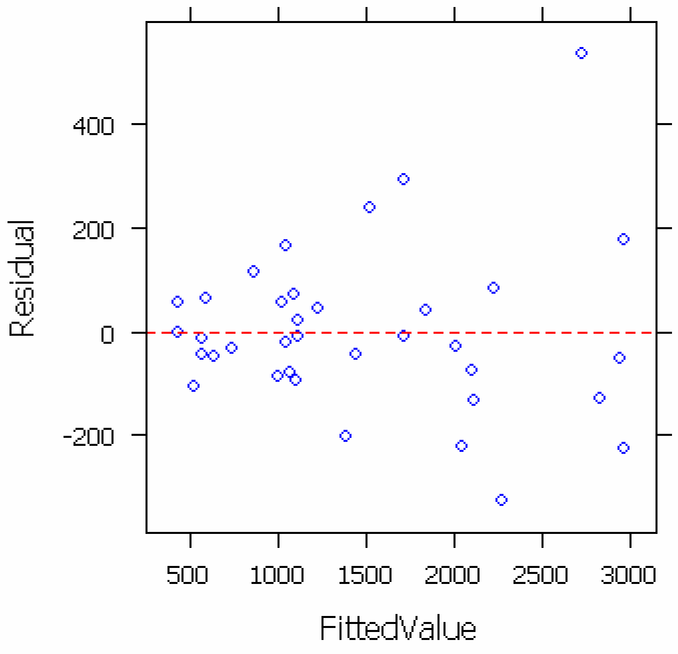
The effect of centering

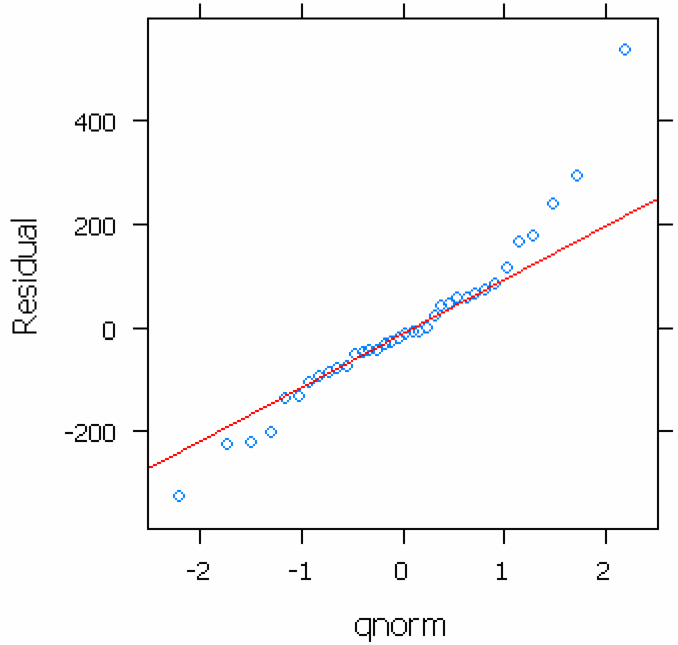
```
janka$d <- scale(janka$Density, scale=F)
jank.1a <- lm(Hardness ~ d, janka)
jank.2a <- update(jank.1a, .~.+I(d^2))
jank.3a <- update(jank.2a, .~.+I(d^3))
range(fitted(jank.3a) - fitted(jank.3)) # check
[1] -4.547474e-13  4.547474e-13
```

```
x <- c(mean(janka$Hard), coef(jank.1a), coef(jank.2a),
       coef(jank.3a))
w[!lower.tri(w)] <- x
round(t(w), 5)
        Degree 0 Degree 1 Degree 2 Degree 3
Model 0 1469.472  0.00000  0.00000  0.00000
Model 1 1469.472 57.50667  0.00000  0.00000
Model 2 1378.197 55.99764  0.50908  0.00000
Model 3 1379.103 53.96095  0.48636  0.00596
```

Diagnostic plots

```
janka <- transform(janka,
           FittedValue = fitted(jank.2),
           Residual = resid(jank.2))

g1 <- xyplot(Residual ~ FittedValue, janka,
   panel = function(x, y, ...) {
        panel.xyplot(x, y, col = "blue", ...)
        panel.abline(h = 0,
                 lty = "dashed", col = "red")
   }, las = 0)
g2 <- qqmath(~Residual, janka,
   panel = function(x, y, ...) {
        panel.qqmath(x, ...)
        panel.qqmathline(x, x, distribution = qnorm,
                 col = "red", lty = "solid")
   }, las = 0)

print(g1, position = c(0, 0.45, 0.55, 1), more = T)
print(g2, position = c(0.45, 0, 1, 0.55))
```

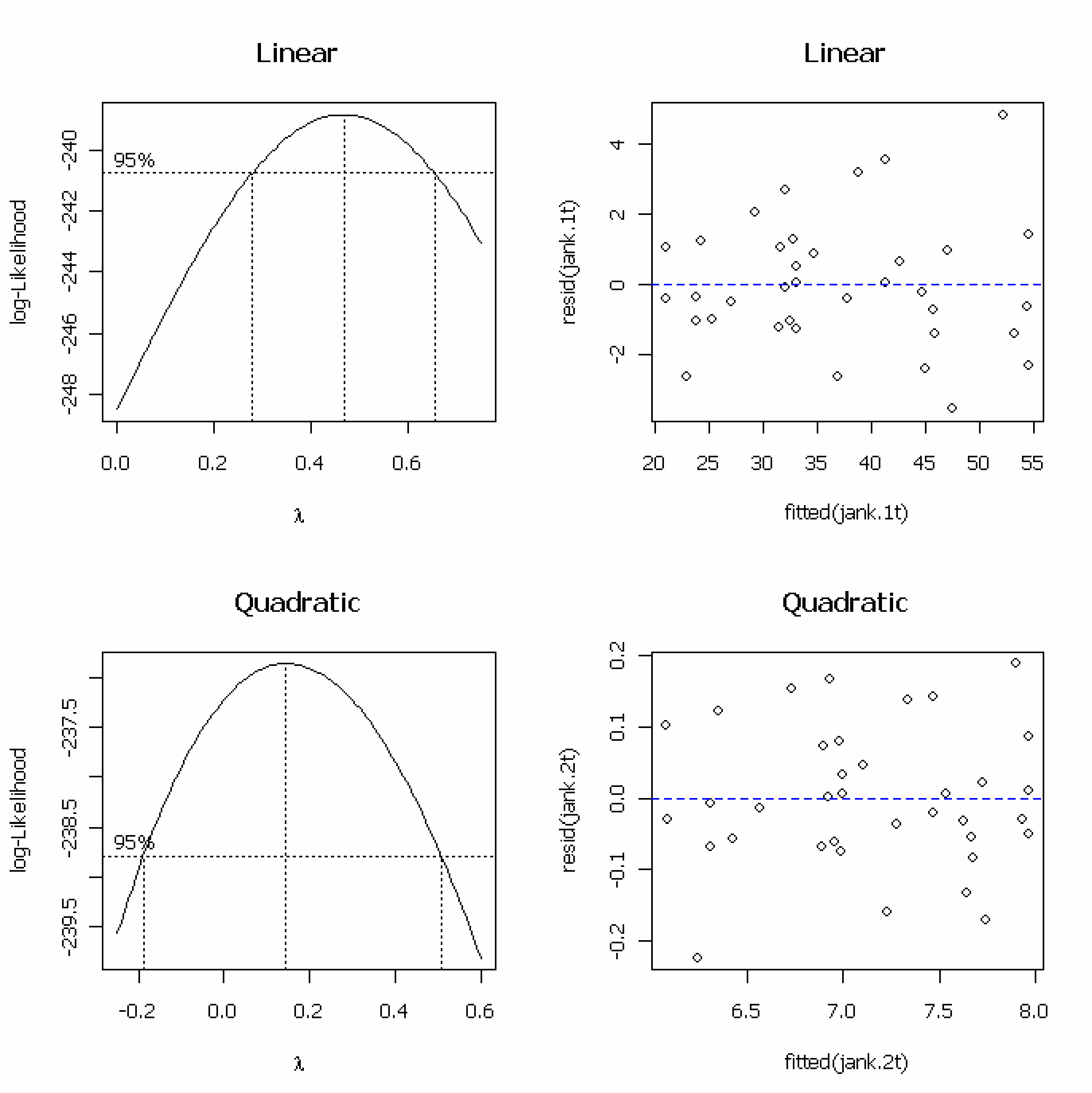

400
200
0
-200
Residual
500
1000
1500
2000
2500
3000
FittedValue


400
200
0
-200
Residual
-2
-1
0
1
2
qnorm

Transformations

- A transformation may be needed to stabilize the variance.
- This will also have an effect on the mean and may tend to linearise it as well.

```
library(MASS)
par(mfrow = c(2,2))
boxcox(jank.1, lambda = seq(0, 0.75, len=10))
title(main = "Linear")
jank.1t <- update(jank.1, sqrt(.) ~ .)
plot(fitted(jank.1t), resid(jank.1t), main = "Linear")
abline(h = 0, lty="dashed", col="blue")

boxcox(jank.2, lambda = seq(-0.25, 0.6, len=10))
title(main = "Quadratic")
jank.2t <- update(jank.2, log(.) ~ .)
plot(fitted(jank.2t), resid(jank.2t), main = "Quadratic")
abline(h = 0, lty="dashed", col="blue")
```


Linear
log-Likelihood
-240
-242
-244
-246
-248
95%
0.0
0.2
0.4
0.6
λ
Linear
resid(jank.1t)
4
2
0
-2
20
25
30
35
40
45
50
55
fitted(jank.1t)
Quadratic
log-Likelihood
-237.5
-238.5
-239.5
95%
-0.2
0.0
0.2
0.4
0.6
λ
Quadratic
resid(jank.2t)
0.2
0.1
0.0
-0.1
-0.2
6.5
7.0
7.5
8.0
fitted(jank.2t)

Messages so far

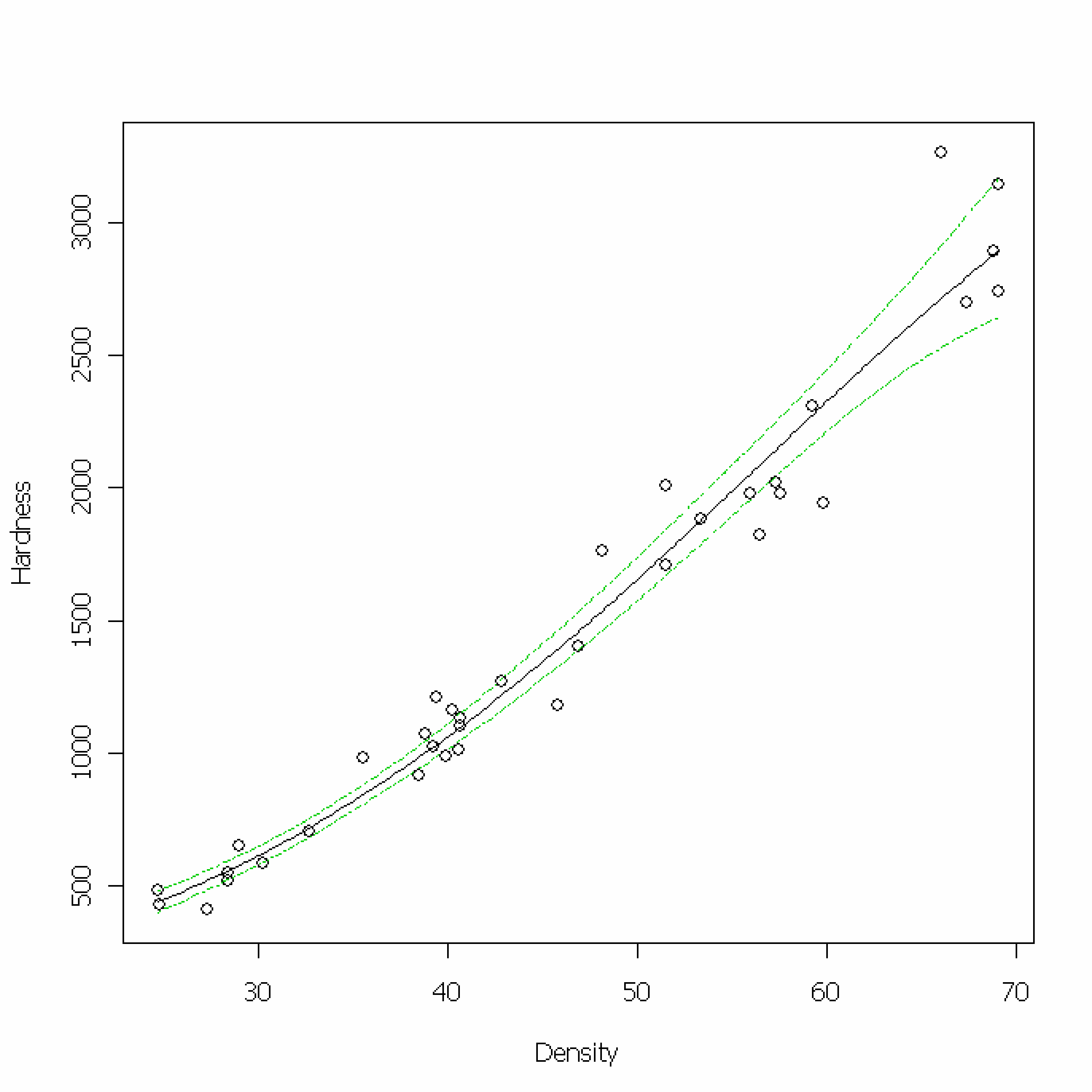
- A square-root transformation provides a straight-line relationship, but does not quite make the variance stable
- A ~log transformation stabilizes the variance, but still requires a quadratic model.
- A stable variance is much more important than a straight-line relationship, but can we have both?
- A natural model to consider is a generalized linear model with constant coefficient of variation and square-root link
- These suggest a gamma model.

Predictions from the transformed model

```
janka2 <- with(janka,
  data.frame(Density =
  seq(min(Density), max(Density), len=200)))

pjank.2t <- predict(jank.2t, new = janka2, se=T)
tau <- qt(0.975, 36 - 3)

janka2 <- with(pjank.2t,
  transform(janka2,
      mean = fit,
      lo = fit - tau*se.fit,
      up = fit + tau*se.fit))
bias.corr <-
  0.5*sum(resid(jank.2t)^2)/pjank.2t$df
```

Predictions, cont'd

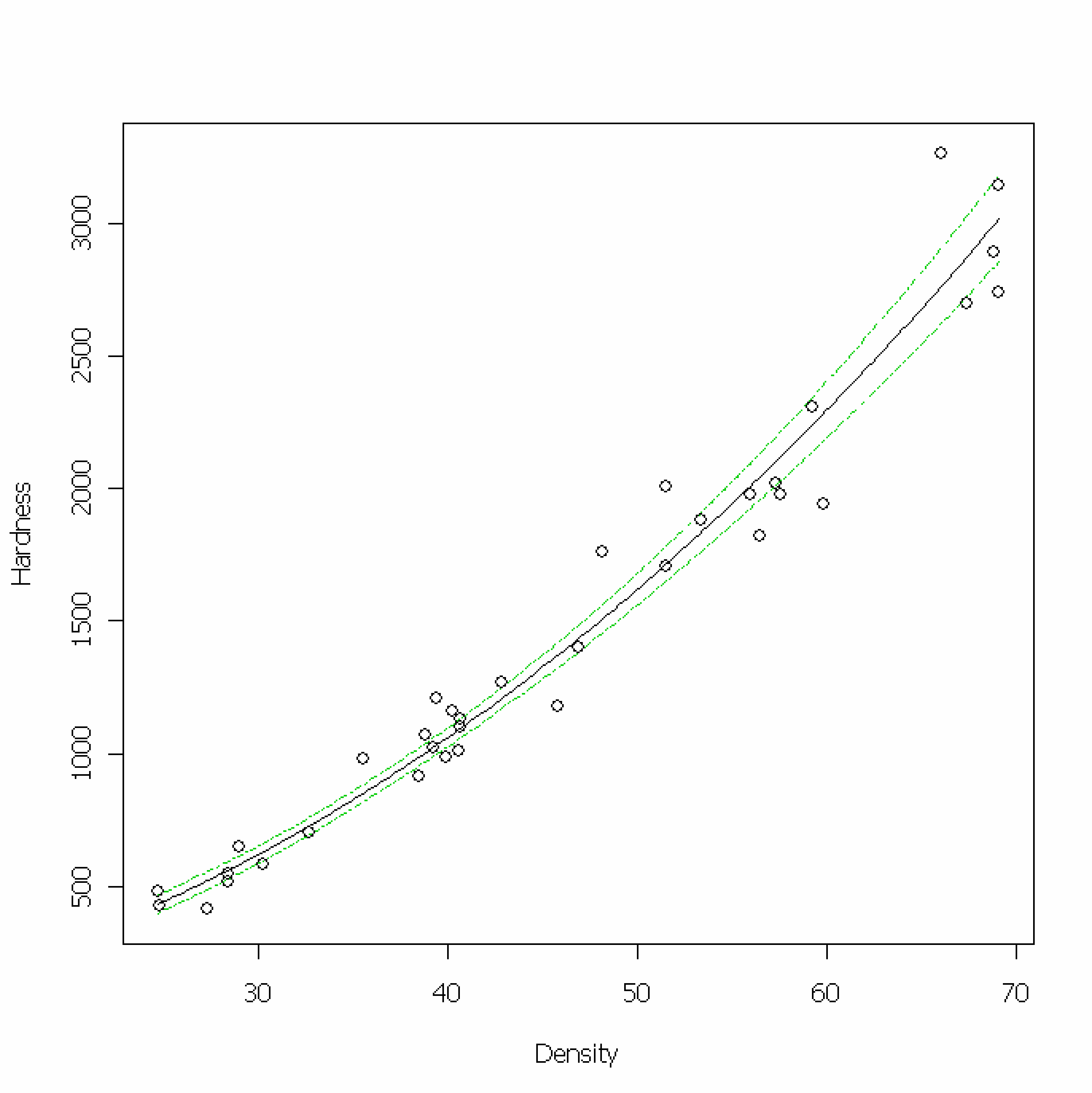
```
janka2 <- transform(janka2,
  Hardness = exp(mean + bias.corr),
  upper = exp(up + bias.corr),
  lower = exp(lo + bias.corr))
rg <- with(janka2,
  range(Hardness, lower, upper, janka$Hardness))

with(janka2, {
  par(mfrow=c(1,1))
  plot(Density, Hardness, type = "l", ylim = rg)
  lines(Density, upper, lty=4, col=3)
  lines(Density, lower, lty=4, col=3)
})
with(janka, points(Density, Hardness))
```



Hardness
500
1000
1500
2000
2500
3000
Density
30
40
50
60
70

A quasi-likelihood GLM

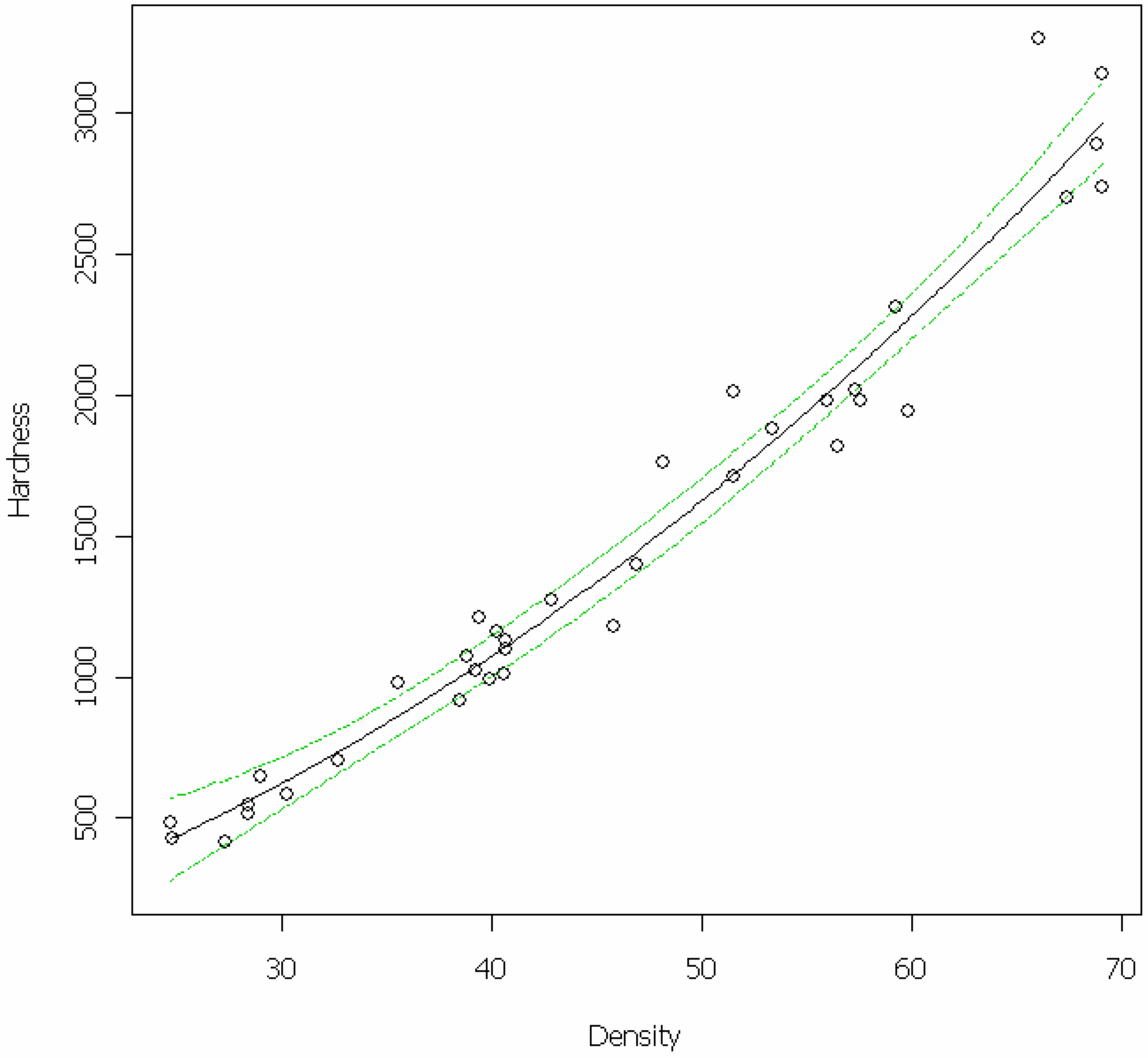
```
jank.glm <- glm(Hardness ~ Density, family =
        quasi(link = sqrt, variance = "mu^2"), janka, trace = T)
Deviance = 0.3335429 Iterations - 1
Deviance = 0.3287643 Iterations - 2
Deviance = 0.3287642 Iterations - 3
Deviance = 0.3287642 Iterations - 4

pjank.glm <- predict(jank.glm, newdata = janka2, se.fit = T)

janka3 <- with(pjank.glm,
  transform(janka2, Hardness = fit^2,
                    lower = (fit - 2*se.fit)^2,
                    upper = (fit+2*se.fit)^2))

rg <- with(janka3, range(lower, upper, janka$Hardness))

par(mfrow=c(1,1))
with(janka3, {
   plot(Density, Hardness, type = "l", ylim = rg)
   lines(Density, upper, lty=4, col=3)
   lines(Density, lower, lty=4, col=3)
})
with(janka, points(Density, Hardness))
```



3000
2500
2000
1500
1000
500
Hardness
30
40
50
60
70
Density

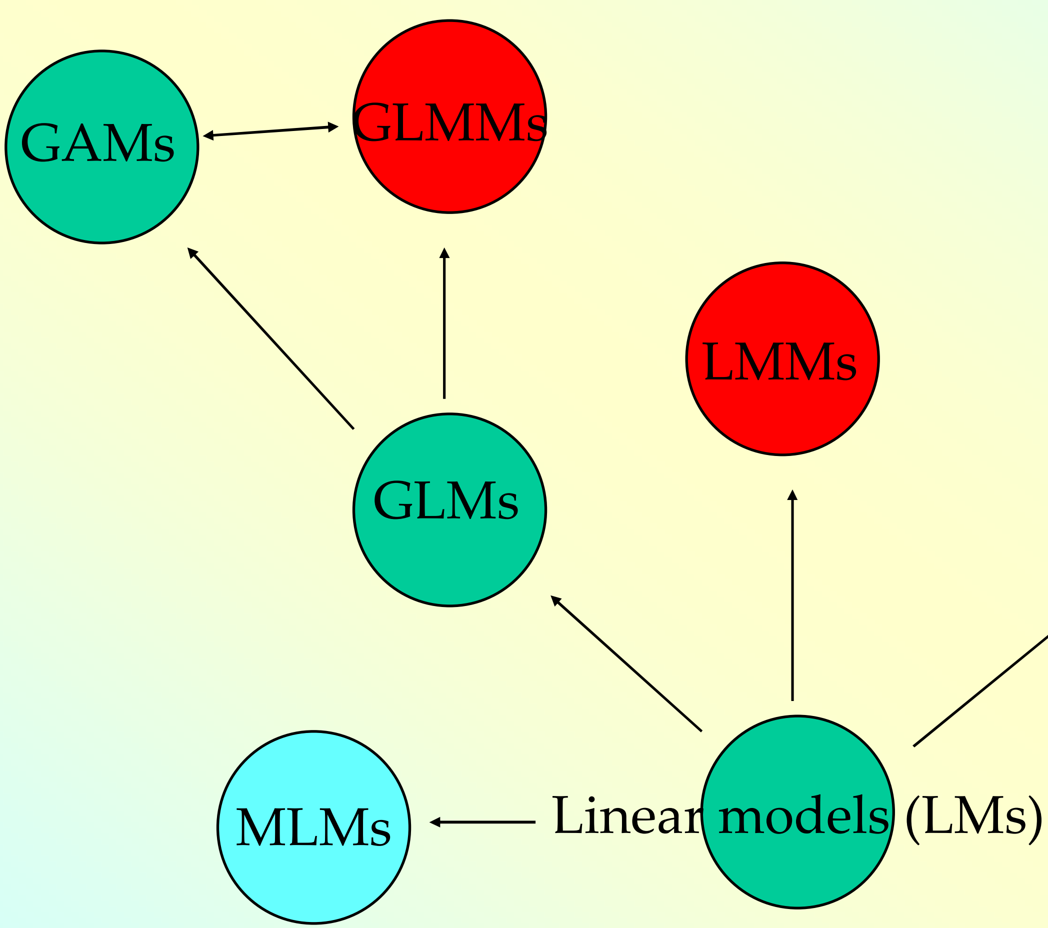
Predictions from the untransformed model

```
pjank.2 <- predict(jank.2, new = janka2, se=T)
tau <- qt(0.975, 36 - 3)

janka2 <- with(pjank.2,
   transform(janka2,
        Hardness = fit,
        upper = fit + tau*se.fit,
        lower = fit-tau*se.fit))
rg <- with(janka2, range(lower, upper, janka$Hard))
with(janka2, {
   par(mfrow=c(1,1))
   plot(Density, Hardness, type = "l", ylim = rg)
   lines(Density, upper, lty=4, col=3)
   lines(Density, lower, lty=4, col=3)
})
with(janka, points(Density, Hardness))
```



CSIRO

Generalizations of Traditional Linear Models: a Roadmap



Explanation of acronyms

	Acronym	S function
Linear Models	LM	lm, aov
Multivariate LMs	MLM	manova
Generalized LMs	GLM	glm
Linear Mixed Models	LMM	lme, lmer, aov
Non-linear Models	NLM	nls
Non-linear Mixed Models	NLMM	nlme
Generalized LMMs	GLMM	glmmPQL, lmer
Generalized Additive Ms	GAM	gam (mgcv)