Regression III

Regrssion Diagnostics

Dave Armstrong

Goals for Today

1. Discuss Diagnostics for Heteroskedasticity
    - Use variance modeling in GAMLSS to fix
2. Investigate Randomized Normalized Quantile Residuals as a model diagnostic
    - Density Plots and Worm Plots as diagnostics.
3. Describe methods for identifying influential points
    - Implement Robust Regression in GAMLSS framework.
    - Residual-Residual plots.

Heteroskedasticity

- An important assumption of the least-squares regression model is that the variance of the errors around the regression surface is everywhere the same: $V(E) = V(Y|x_1, \ldots, x_k) = \sigma^2$.
- Non-constant error variance does not cause biased estimates, but it does pose problems for efficiency and the usual formulas for standard errors are inaccurate
  - OLS estimates are inefficient because they give equal weight to all observations regardless of the fact that those with large residuals contain less information about the regression
- Two types of non-constant error variance are relatively common:
  - Error variance increases as the expectation of Y increases;
  - There is a systematic relationship between the errors and one of the X's

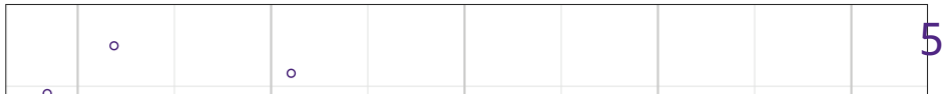
Notes

Type notes here...

Example

In the residual plot , we see the familiar "fanning" in the plot - i.e., the variance of the residuals is decreasing as the fitted values get larger

```
f <- "http://www.quantoid.net/files/reg3/weakliem.txt"
library(rio)
Weakliem <- import(f)
W <- Weakliem[-c(21,22,24, 25,49), ]
mod2 <- lm(secpay ~ log(gdp), data=W)
```



Notes

Type notes here...

Test of Heteroskedasticity

- We start by calculating the standardized squared residuals

$$U_i = \frac{E_i^2}{\hat{\sigma}^2} = \frac{E_i^2}{\frac{\sum E_i^2}{n}}$$

- Regress the U_i on all of the explanatory variable X's, finding the fitted values:

$$U_i = \eta_0 + \eta_1 X_{i1} + \cdots + \eta_p X_{ip} + \omega_i$$

- The score test, which s distributed as χ^2 with p degrees of freedom is:

$$S_0^2 = \frac{\sum(\hat{U}_i - \bar{U})^2}{2}$$

Notes

Type notes here...

By Hand

```
# make U = residuals^2/SEr
U <- (mod2$residuals^2)/
  (sum(mod2$residuals^2)/length(mod2$residuals))
# if using all varaibles use this one
upmod <- update(mod2, U ~ .)
# Alternatively, if using only fitted values
# upmod <- lm(U ~ fitted.values(mod2))
# find degrees of freedom
df <- upmod$rank-1
# calculate the score
score <- sum((fitted(upmod) - mean(U))^2)/2
score
```

```
## [1] 6.025183
```

```
# calculate the p-value
round(pchisq(score, df, lower.tail=FALSE), 3)
```

```
## [1] 0.014
```

or ...

```
ncvTest(mod2)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 6.025183, Df = 1, p = 0.014103
```

Notes

Type notes here...

Variance modeling

If the variance in residuals is related to the independent variables in the model, we could model:

$$log\left(\sigma_{\varepsilon}\right) = f(X)$$

where $f(\cdot)$ is some functional relationship to be estimated between the covariates and the log variance.

Notes

Type notes here...

In GAMLSS

```
library(gamlss)
W2 <- W %>%
  dplyr::select(secpay, gdp, gini,
                hetero, union,
                democrat) %>%
  na.omit
mm <- gamlss(secpay ~ log(gdp),data=W2)
```

```
## GAMLSS-RS iteration 1: Global Deviance = -60.3546
## GAMLSS-RS iteration 2: Global Deviance = -60.3546
```

```
vm <- gamlss(secpay ~ log(gdp),
             ~ log(gdp)
             ,data=W2)
```

```
## GAMLSS-RS iteration 1: Global Deviance = -64.0693
## GAMLSS-RS iteration 2: Global Deviance = -65.5385
## GAMLSS-RS iteration 3: Global Deviance = -65.7152
## GAMLSS-RS iteration 4: Global Deviance = -65.7267
## GAMLSS-RS iteration 5: Global Deviance = -65.7273
```

Notes

Type notes here...

Model Summaries

```
summary(mm)
```

```
## ******************************************************************
## Family:  c("NO", "Normal")
##
## Call:  gamlss(formula = secpay ~ log(gdp), data = W2)
##
## Fitting method: RS()
##
## ------------------------------------------------------------------
## Mu link function:  identity
## Mu Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.29516    0.20442   6.336 1.25e-06 ***
## log(gdp)    -0.05445    0.02133  -2.553   0.0172 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## ------------------------------------------------------------------
## Sigma link function:  log
## Sigma Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -2.4967     0.1336  -18.68 3.35e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## ------------------------------------------------------------------
## No. of observations in the fit:  28
```

```
summary(vm)
```

```
## ******************************************************************
## Family:  c("NO", "Normal")
##
## Call:  gamlss(formula = secpay ~ log(gdp), sigma.formula = ~log(gdp),
##     data = W2)
##
## Fitting method: RS()
##
## ------------------------------------------------------------------
## Mu link function:  identity
## Mu Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.45047    0.09944  14.586 1.99e-13 ***
## log(gdp)    -0.07144    0.01159  -6.165 2.28e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## ------------------------------------------------------------------
## Sigma link function:  log
## Sigma Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -8.1615     2.2528  -3.623  0.00136 **
## log(gdp)      0.5827     0.2353   2.477  0.02070 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' '
##
```

Notes

Type notes here...

Normalized (randomized) quantile residuals

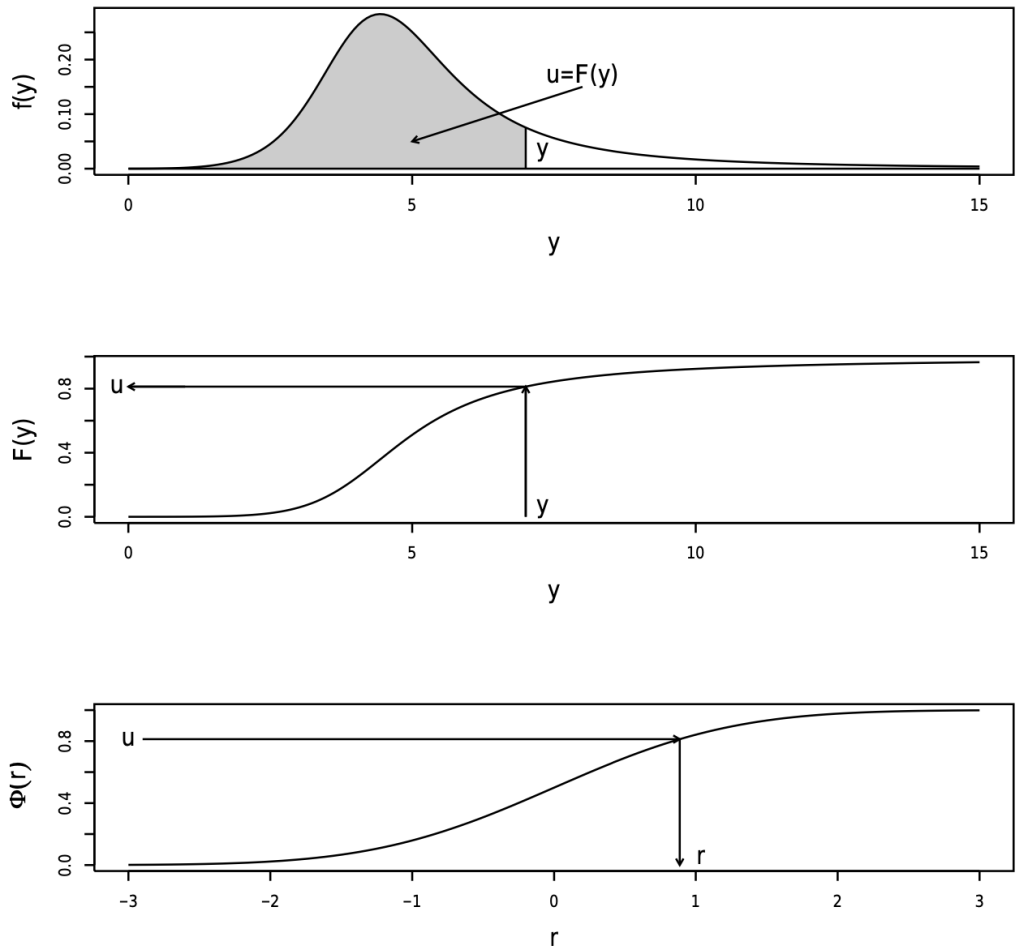
For any distribution $f(y|\theta)$ fit to y_i,

- $u_i = F(y_i|\hat{\theta})$, where $F(\cdot)$ is the CDF and $\hat{\theta}$ is a vector of moments for the distributions $f(\cdot)$ and $F(\cdot)$.
- u_i should have a uniform distribution under the correct model specification.
- $\Phi^{-1}(u_i)$, the quantile function for the standard normal is used to transform the uniform distribution of u_i into a standard normal distribution.

Notes

Type notes here...

Process


f(y)
0.00
0.10
0.20
u=F(y)
y
0
5
10
15
y
F(y)
0.0
0.4
0.8
u
y
0
5
10
15
y
Φ(r)
0.0
0.4
0.8
u
r
−3
−2
−1
0
1
2
3
r

Notes

Type notes here...

Mean and variance

```
set.seed(54434)
x <- runif(1000,-1,1)
mu <- 1 + 2*x
sig <- exp(x)
y <-  mu + rnorm(1000, 0, sig)
df1 <- data.frame(x=x, y=y)
m1 <- gamlss(y ~ x,
             ~x, data = df1)
mu.hat <- predict(m1, what="mu")
sig.hat <- predict(m1, what="sigma")
e <- y-mu.hat
u <- pNO(df1$y, mu.hat, exp(sig.hat))
r <- qnorm(u)
plot.df <- data.frame(
  res = c(e, r),
  type = factor(rep(1:2, each=1000),
                labels=c("Raw", "NRQ"))
)
```

Notes

Type notes here...

Discrete distributions

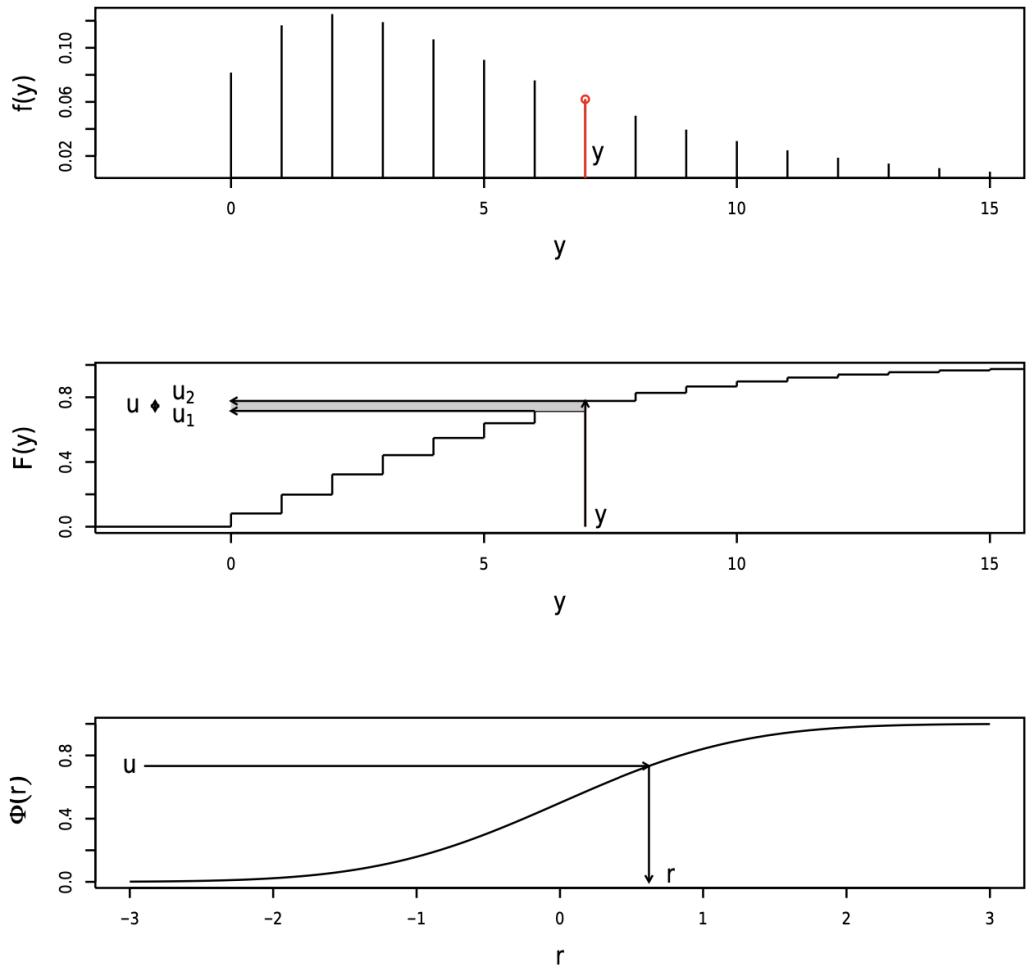
For any distribution $f(y|\theta)$ fit to y_i,

- u_i is a random draw from the interval $[u_1, u_2]$, where
  - $u_1 = F(y_i - 1|\hat{\theta})$ and $u_2 = F(y_i|\hat{\theta})$
  - $F(\cdot)$ is the CDF and $\hat{\theta}$ is a vector of moments for the distributions $f(\cdot)$ and $F(\cdot)$.
- $\Phi^{-1}(u_i)$, the quantile function for the standard normal is used to transform the uniform distribution of u_i into a standard normal distribution.

Notes

Type notes here...

Process


f(y)
0.02
0.06
0.10
y
0
5
10
15
y
F(y)
0.0
0.4
0.8
u
u2
u1
y
0
5
10
15
y
Φ(r)
0.0
0.4
0.8
u
r
−3
−2
−1
0
1
2
3
r

Notes

Type notes here...

Binomial Example

```
ystar <- 2*x
y <- rbinom(1000, 1, plogis(ystar))
df2 <- data.frame(x=x,y=y)
m2 <- gamlss(y ~ x, data=df2,
             family=BI)
```

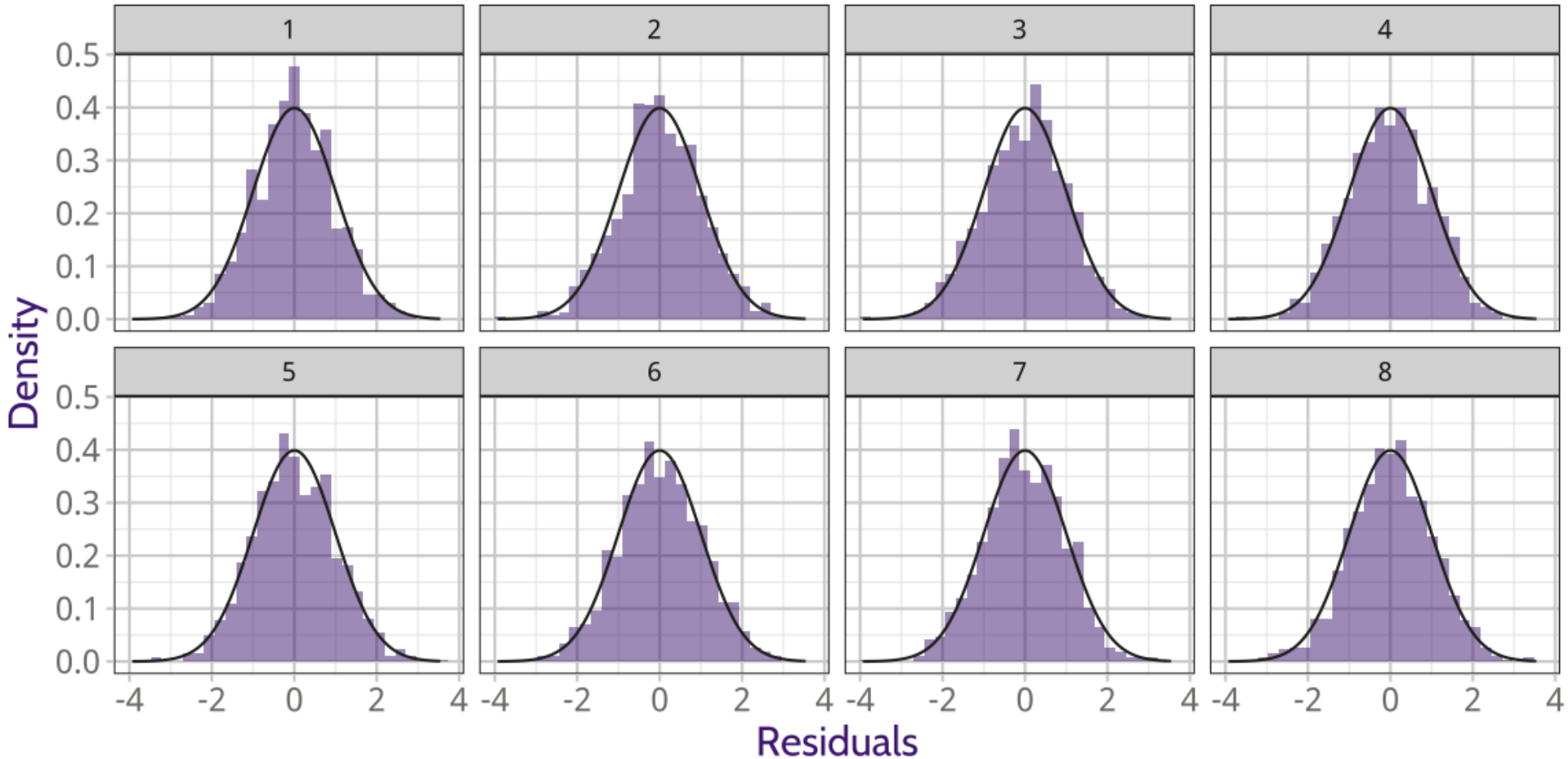
```
## GAMLSS-RS iteration 1: Global Deviance = 1168.468
## GAMLSS-RS iteration 2: Global Deviance = 1168.468
```

```
mu.hat <- predict(m2, what="mu",
                  type="response")
e <- y-mu.hat
tmp <- NULL
u1 <- dBI(0, 1, mu=mu.hat)
u1 <- ifelse(y == 0, 0, u1)
u2 <- pBI(y, 1, mu=mu.hat)
for(i in 1:8){
  u <- runif(length(y), u1, u2)
  u <- ifelse(u > 0.999999,
              u - 1e-16, u)
  u <- ifelse(u < 1e-06,
              u + 1e-16, u)
  rqres <- qnorm(u)
  tmp <- rbind(tmp,
    data.frame(r=rqres, draw=i))
}
```

Notes

Type notes here...

Binomial Example 2



Notes

Type notes here...

Worm plot

A worm plot is a de-trended quantile-quantile plot

- 95% of the residuals should be inside the point-wise confidence bounds.
- patterns can identify potential problems.

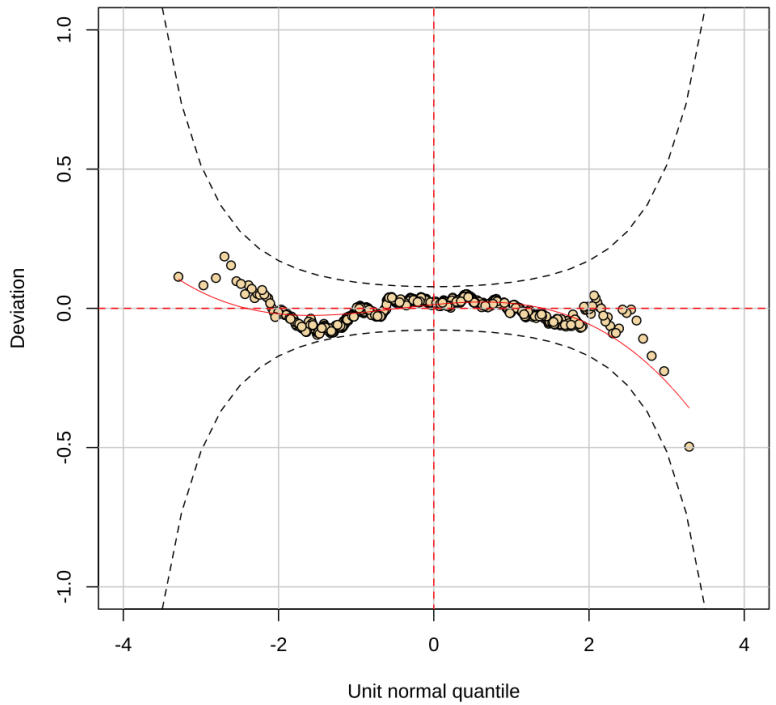
Shape of worm plot (or its fitted curve)	**Residuals**	**Fitted distribution**
level: above the origin	mean too high	fitted location too low
level: below the origin	mean too low	fitted location too high
line: positive slope	variance too high	fitted scale too low
line: negative slope	variance too low	fitted scale too high
U-shape	positive skewness	fitted skewness too low
inverted U-shape	negative skewness	fitted skewness too high
S-shape with left bent down	leptokurtosis	tails of fitted distribution too light
S-shape with left bent up	platykurtosis	tails of fitted distribution too heavy

Notes

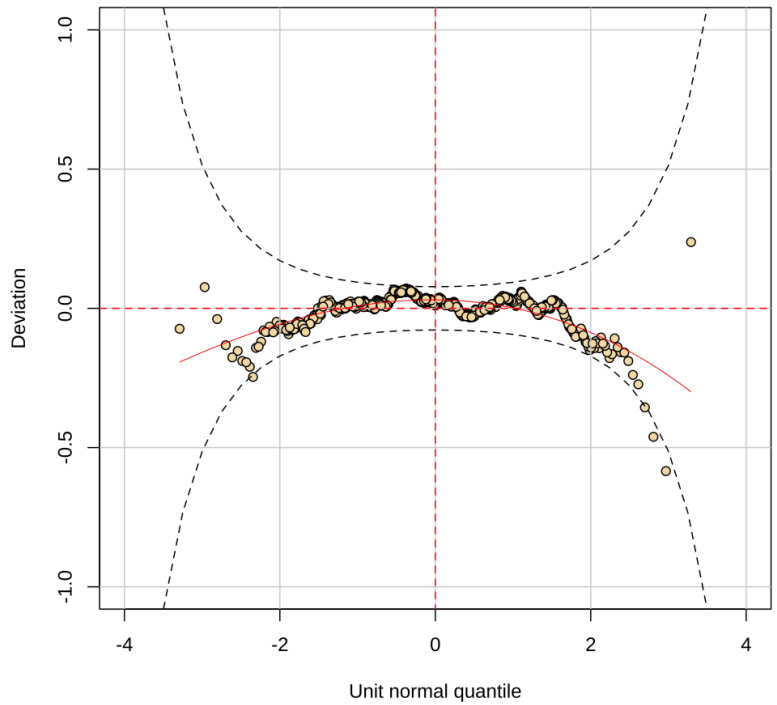
Type notes here...

Worm Plots from Examples

```
wp(m1, ylim.all=TRUE)
```




```
wp(m2, ylim.all=TRUE)
```



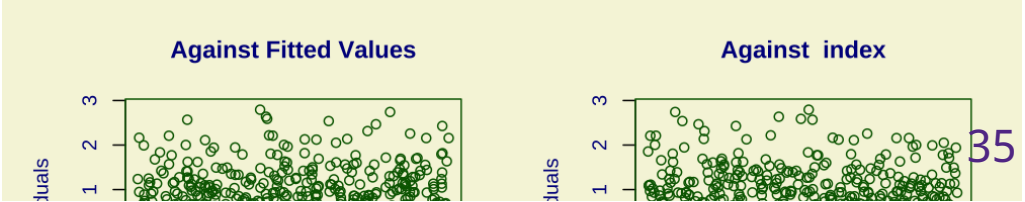
Notes

Type notes here...

plot method (m1)

```
plot(m1)
```

```
## ****************************************************************
##           Summary of the Quantile Residuals
##                             mean   =  0.0004486005
##                         variance   =  1.001001
##                 coef. of skewness  =  -0.07516835
##                 coef. of kurtosis  =  2.796859
## Filliben correlation coefficient  =  0.9992205
## ****************************************************************
```



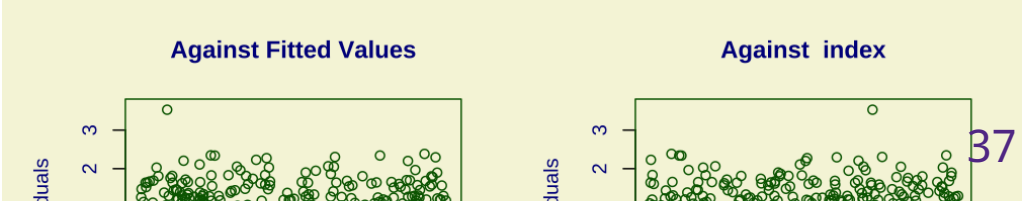
Notes

Type notes here...

plot method (m2)

```
plot(m2)
```

```
## ****************************************************************
##      Summary of the Randomised Quantile Residuals
##                           mean   =  0.004974562
##                       variance   =  0.9899672
##                   coef. of skewness  =  -0.1461771
##                   coef. of kurtosis  =  2.996606
## Filliben correlation coefficient  =  0.9986558
## ****************************************************************
```



Notes

Type notes here...

Bad model

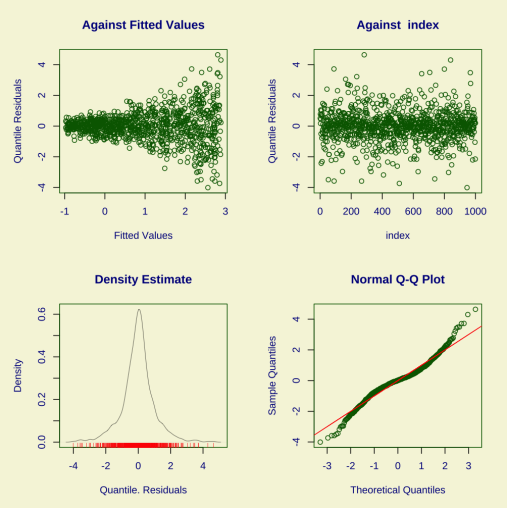
Note that in the model below, the variance is a function of x, but is unmodeled.

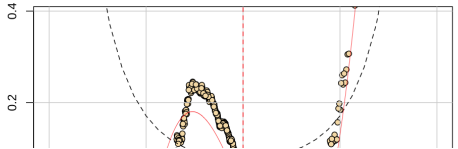
```
m3 <- gamlss(y ~ x, data=df1)
```

```
## GAMLSS-RS iteration 1: Global Deviance = 3516.111
## GAMLSS-RS iteration 2: Global Deviance = 3516.111
```

```
plot(m3)
```

```
## ******************************************************************
##           Summary of the Quantile Residuals
##                               mean   =  -1.707241e-15
##                           variance   =  1.001001
##                   coef. of skewness  =  0.04431577
##                   coef. of kurtosis  =  5.533666
## Filliben correlation coefficient  =  0.9787666
## ******************************************************************
```





Notes

Type notes here...

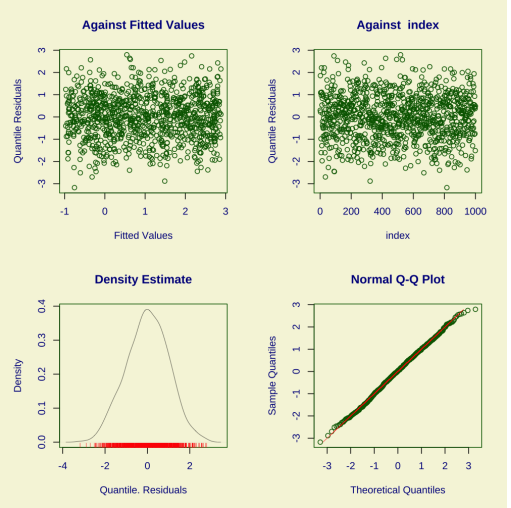
Fixed model

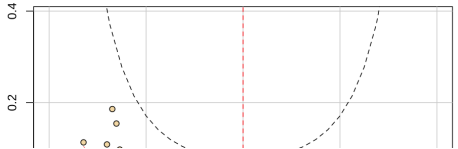
```
m3a <- gamlss(y ~ x,
              sigma.formula= ~ x,
              data=df1)
```

```
## GAMLSS-RS iteration 1: Global Deviance = 2891.18
## GAMLSS-RS iteration 2: Global Deviance = 2891.177
## GAMLSS-RS iteration 3: Global Deviance = 2891.177
```

```
plot(m3a)
```

```
## ******************************************************************
##           Summary of the Quantile Residuals
##                               mean   =  0.0004486005
##                           variance   =  1.001001
##                   coef. of skewness  =  -0.07516835
##                   coef. of kurtosis  =  2.796859
## Filliben correlation coefficient  =  0.9992205
## ******************************************************************
```





Notes

Type notes here...

Bad Model (2)

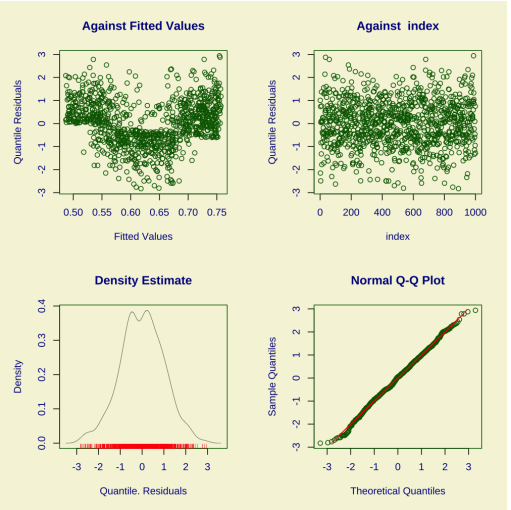
Note that in the model below, has an un-modeled non-linearity in x.

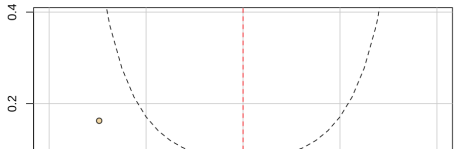
```
set.seed(54434)
a <- sqrt(12)
x <- runif(1000, -a,a)
p <- plogis(-1.9 + .7*x + x^2)
y <- rbinom(1000, 1, p)
df2 <- data.frame(x=x,y=y)
m4 <- gamlss(y ~ x, data=df2,
             family=BI)
```

```
## GAMLSS-RS iteration 1: Global Deviance = 1289.852
## GAMLSS-RS iteration 2: Global Deviance = 1289.852
```

```
plot(m4)
```

```
## ******************************************************************
##      Summary of the Randomised Quantile Residuals
##                           mean   =  -0.001778572
##                       variance   =  0.9943234
##                   coef. of skewness  =  -0.05718645
##                   coef. of kurtosis  =  2.996498
```





Notes

Type notes here...

Fixed Model (2)

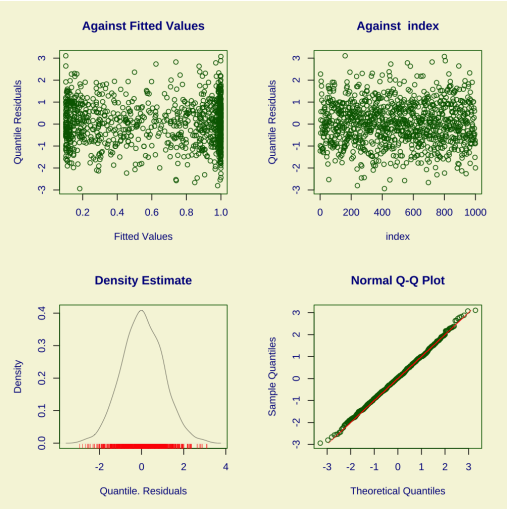
Note that in the model below, has an un-modeled non-linearity in x.

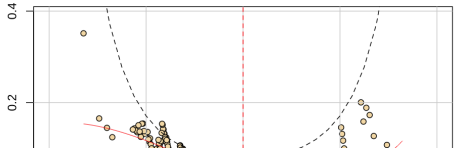
```
m4a <- gamlss(y ~ pb(x), data=df2,
              family=BI)
```

```
## GAMLSS-RS iteration 1: Global Deviance = 656.2069
## GAMLSS-RS iteration 2: Global Deviance = 656.2069
```

```
plot(m4a)
```

```
## ******************************************************************
##      Summary of the Randomised Quantile Residuals
##                           mean   =  0.04478385
##                       variance   =  0.9513026
##                   coef. of skewness  =  0.05771978
##                   coef. of kurtosis  =  3.038232
## Filliben correlation coefficient  =  0.9995606
## ******************************************************************
```





Notes

Type notes here...

Diagnosing Linearity Problems (GLM)

Let's look at a non-linear model (logit)

```
library(splines)
library(car)
library(rio)
dat <- import(
  "http://www.quantoid.net/files/9591/anes2008_binary.dta")
vmod1 <- glm(voted ~ age + educ + female +
    leftright, data=dat, family=binomial(link="logit"))
vmod1a <- glm(voted ~ age + educ + female +
    bs(leftright, df=5), data=dat, family=binomial(link="logit"))
```

How can we diagnose non-linearity problems in this model?

- C+R plot

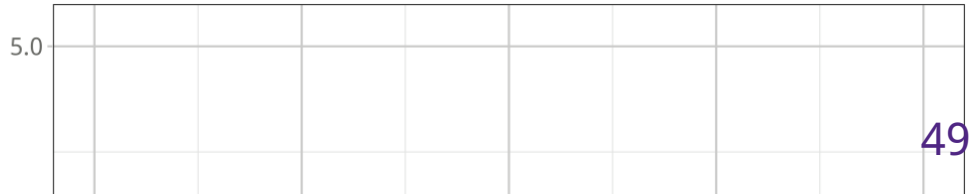
Notes

Type notes here...

C+R Plot

We'll make it ourself to have a bit more control over the plot.

```
cprw1 <- residuals(vmod1, type="partial")[,"leftright"]
g <- ggplot(mapping=aes(
    x=dat$leftright,
    y=cprw1)) +
  geom_point(pch=1) +
  geom_smooth(method="loess",
              se=T) +
  theme_bw() +
  mytheme() +
  coord_cartesian(ylim=c(-5,5)) +
  labs(x="Left-Right",
       y="Component + Residual")
```



Notes

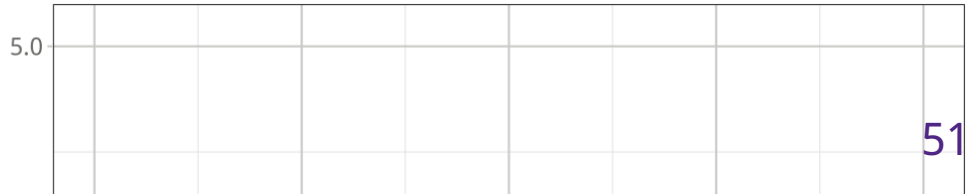
Type notes here...

C+R Plot (2)

This one actually looks **worse** than the previous one.

```
vmod1a <- glm(voted ~ age + educ +
  female + bs(leftright, df=5),
  data=dat, family=binomial(link="logit"))

cprw1a <- residuals(vmod1a,
    type="partial")[,"bs(leftright, df = 5)"]
g <- ggplot(mapping=aes(x=dat$leftright, y=cprw1a)) +
  geom_point(pch=1) +
  geom_smooth(method="loess",
              se=T) +
  theme_bw() +
  mytheme() +
  coord_cartesian(ylim=c(-5,5)) +
  labs(x="Left-Right",
       y="Component + Residual")
```



Notes

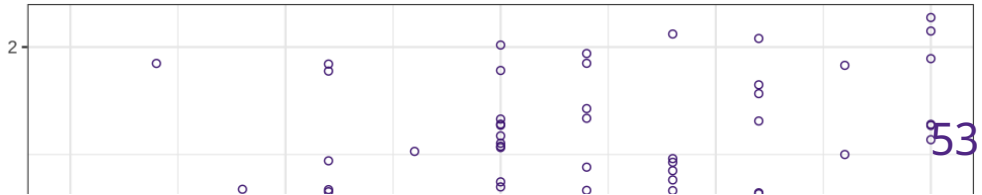
Type notes here...

C+R GAMLSS

```
vmod2a <- gamlss(voted ~ age + educ + female +
    leftright, data=dat, family=BI)
```

```
## GAMLSS-RS iteration 1: Global Deviance = 543.4565
## GAMLSS-RS iteration 2: Global Deviance = 543.4565
```

```
eg2a <- residuals(vmod2a, type="partial",
                  terms="leftright")
g <- ggplot(mapping=aes(y=eg2a,
                        x=dat$leftright)) +
  geom_point(pch=1) +
  geom_smooth(method="loess", se=TRUE) +
  coord_cartesian(ylim=c(-2,2)) +
  theme_bw() +
  labs(x="left-right",
       y="Component + Residual")
```



Notes

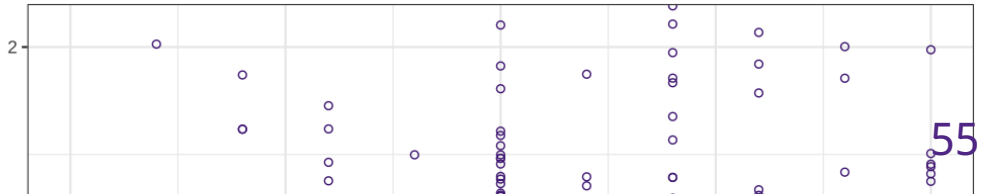
Type notes here...

GAMLSS Fixed Model

```
vmod2b <- gamlss(voted ~ age + educ + female +
    pb(leftright), data=dat, family=BI)
```

```
## GAMLSS-RS iteration 1: Global Deviance = 526.0427
## GAMLSS-RS iteration 2: Global Deviance = 526.0805
## GAMLSS-RS iteration 3: Global Deviance = 526.0821
## GAMLSS-RS iteration 4: Global Deviance = 526.0821
```

```
eg2b <- residuals(vmod2b, type="partial", terms="leftright")
g <- ggplot(mapping=aes(y=eg2b, x=dat$leftright)) +
  geom_point(pch=1) +
  geom_smooth(method="loess", se=TRUE) +
  coord_cartesian(ylim=c(-2,2)) +
  theme_bw() +
  labs(x="left-right", y="Component + Residual")
```

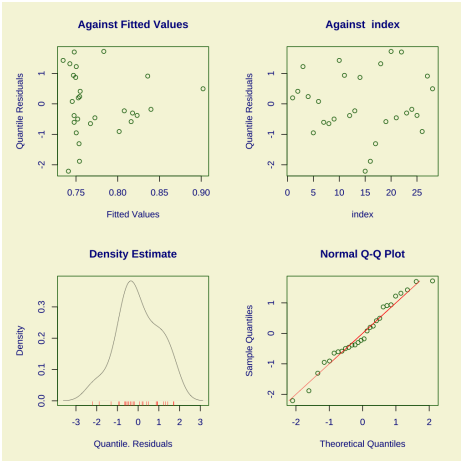


Notes

Type notes here...

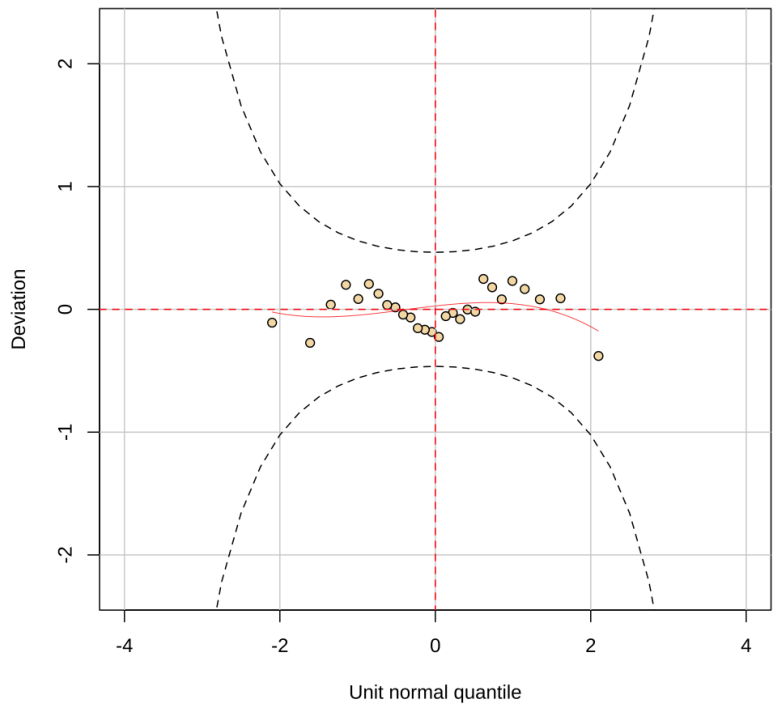
Secpay (mean only)

```
plot(mm)
```




```
## ****************************************************************
##           Summary of the Quantile Residuals
##                            mean   =  7.442273e-18
##                        variance   =  1.037037
##                  coef. of skewness  =  -0.09825738
##                  coef. of kurtosis  =  2.324909
## Filliben correlation coefficient  =  0.9877011
## ****************************************************************
```

```
wp(mm)
```

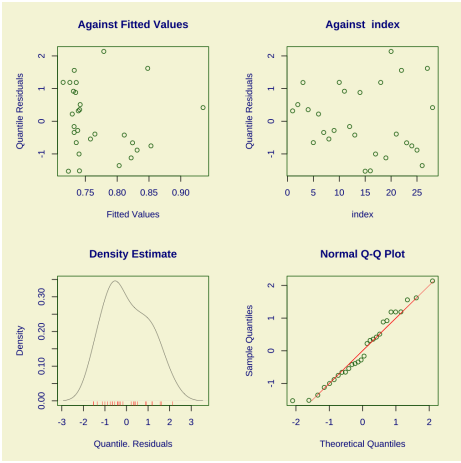


Notes

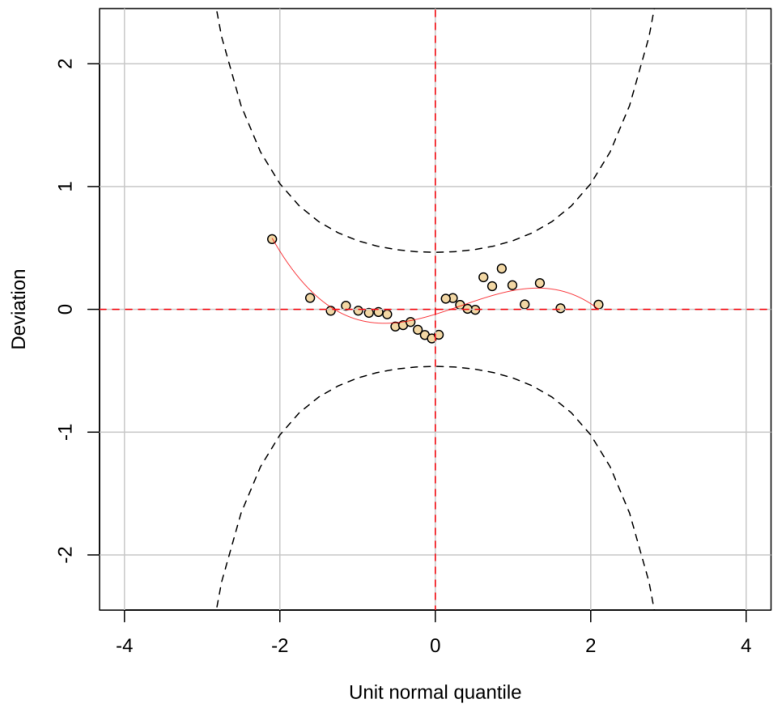
Type notes here...

Secpay (mean and variance)

```
plot(vm)
```




```
wp(vm)
```




```
## ****************************************************************
##           Summary of the Quantile Residuals
##                            mean   =  0.03154716
##                        variance   =  1.036
##                  coef. of skewness  =  0.2683496
##                  coef. of kurtosis  =  1.935488
## Filliben correlation coefficient  =  0.9847595
## ****************************************************************
```

Notes

Type notes here...

Outliers

- Can cause us to misinterpret patterns in plots
  - Temporarily removing them can sometimes help see patterns that we otherwise would not have
  - Transformations can also spread out clustered observations and bring in the outliers
- More importantly, separated points can have a strong influence on statistical models - removing outliers from a regression model can sometimes give completely different results
  - Unusual cases can substantially influence the fit of the OLS model - Cases that are both outliers and high leverage exert influence on both the slopes and intercept of the model
  - Outliers may also indicate that our model fails to capture important characteristics of the data

Notes

Type notes here...

Regression Outliers

- An observation that is unconditionally unusual in either its Y or X value is called a univariate outlier, but it is not necessarily a regression outlier
- A regression outlier is an observation that has an unusual value of the outcome variable Y, conditional on its value of the explanatory variable X
  - In other words, for a regression outlier, neither the X nor the Y value is necessarily unusual on its own
- Regression outliers often have large residuals but do not necessarily affect the regression slope coefficient
- Also sometimes referred to as vertical outliers

Notes

Type notes here...

High leverage points

- An observation that has an unusual X value - i.e., it is far from the mean of X - has leverage on the regression line
  - The further the outlier sits from the mean of X (either in a positive or negative direction), the more leverage it has
- High leverage does not necessarily mean that it influences the regression coefficients
  - It is possible to have a high leverage and yet follow straight in line with the pattern of the rest of the data. Such cases are sometimes called "good" leverage points because they help the precision of the estimates. Remember, $V(B) = \sigma_\varepsilon^2(\mathbf{X}'\mathbf{X})^{-1}$, so outliers could increase the variance of X.

Notes

Type notes here...

Influential points

- An observation with high leverage that is also a regression outlier will strongly influence the regression line
  - In other words, it must have an unusual X-value with an unusual Y-value given its X-value
- In such cases both the intercept and slope are affected, as the line chases the observation

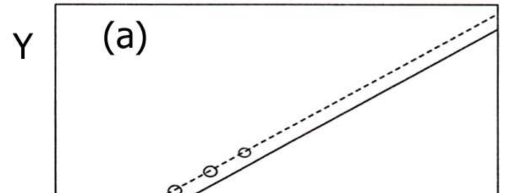
$$\text{Discrepancy} \times \text{Leverage} = \text{Influence}$$

Notes

Type notes here...

Influence

- Figure (a): Outlier without influence because it is in the middle of the X-range
- Figure (b) High leverage without influence because it has a high value of X, but its Y value is in line with the pattern.
- Figure (c): Discrepancy (unusual Y value) and leverage (unusual X value) results in strong influence.



Notes

Type notes here...

Simple solutions

Find the left-out variable

- Great if you can do it.

Leave influential observations in:

- potentially contaminate the relationship and miss important insights from the data.

Take influential observations out:

- potentially harm the external validity of your findings

Include a dummy variable for the outlier

- Same as taking it out, though more disingenuous.

Notes

Type notes here...

Robustness Weighting

Can down-weight influential observations to make them less "important" in the fit of the model.

M-estimation is an iterative technique that iteratively down-weights observations until it converges.

- Still susceptible to groups of influential points.
- For our purposes, it will probably work alright.

Notes

Type notes here...

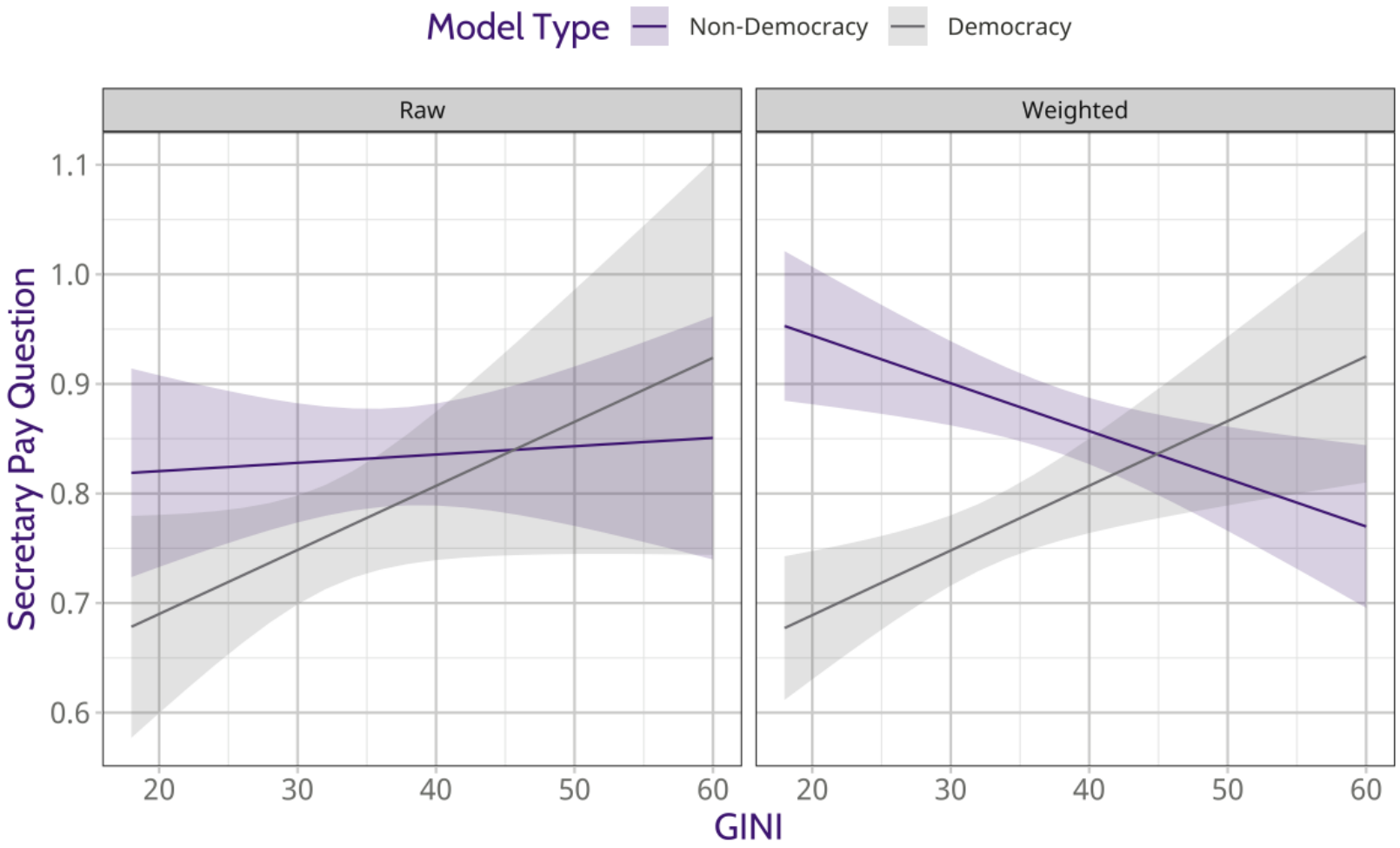
Inequality Data

```
W3 <- Weakliem %>%
  mutate(orig = 1:nrow(Weakliem),
         weight=1) %>%
  dplyr::select(country, orig, secpay, gini,
                democrat, weight) %>%
  na.omit
mod1 <- mod1o <- gamlss(secpay ~ gini*democrat, data=W3, weights=weight, trace=FALSE)
devDiff <- 1
prevDev <- deviance(mod1)
maxit <- 30
k <- 1
while(devDiff > 0 & k < maxit){
  e <- residuals(mod1, type="simple")
  S2e <- sum(e^2)/mod1$df.residual
  se <- e/sqrt(S2e)
  w <- psi.bisquare(se)
  W3$weight <-  w
  mod1 <- gamlss(secpay ~ gini*democrat, data=W3, weights = weight, trace=FALSE)
  devDiff <- abs(deviance(mod1) - prevDev)
  k <- k+1
}
```

Notes

Type notes here...

Result


Model Type
Non-Democracy
Democracy
Raw
Weighted
Secretary Pay Question
1.1
1.0
0.9
0.8
0.7
0.6
20
30
40
50
60
GINI

Notes

Type notes here...

Weights < 0.9

```
W3 %>% filter(weight < .9) %>% arrange(weight)
```

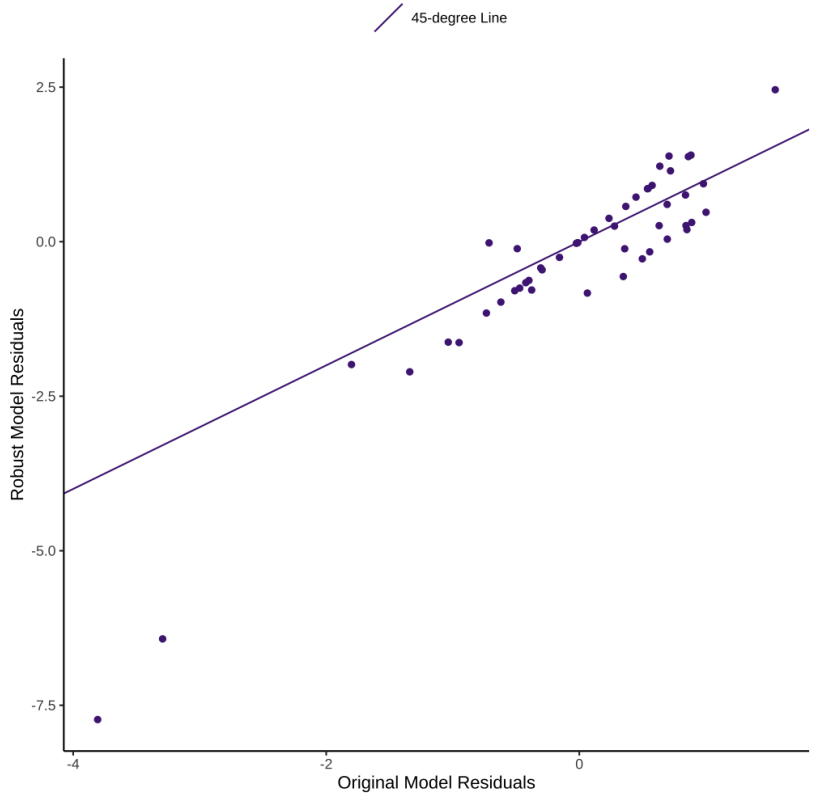
```
##         country orig secpay gini democrat     weight
## 1      Slovakia   49  0.378 19.5        0 0.02366234
## 2 CzechRepublic   25  0.443 26.6        0 0.17255078
## 3       Austria   32  0.888 23.1        1 0.83623632
## 4        Norway   16  0.559 24.2        1 0.87843556
## 5         Chile   22  0.639 56.5        0 0.89128958
```

Notes

Type notes here...

Residual-Residual Plot

```
W3$r0 <- residuals(mod1o)
W3$r1 <- residuals(mod1)
ggplot(W3, aes(x=r0, y=r1)) +
  geom_point() +
  geom_abline(aes(linetype="45-degree Line",
                  slope=1, intercept=0)) +
  theme_classic() +
  theme(legend.position="top") +
  labs(x="Original Model Residuals",
       y="Robust Model Residuals",
       linetype="")
```



Notes

Type notes here...