Exploratory Data Analysis

Bivariate Data

Pramod Duvvuri

4/5/2019

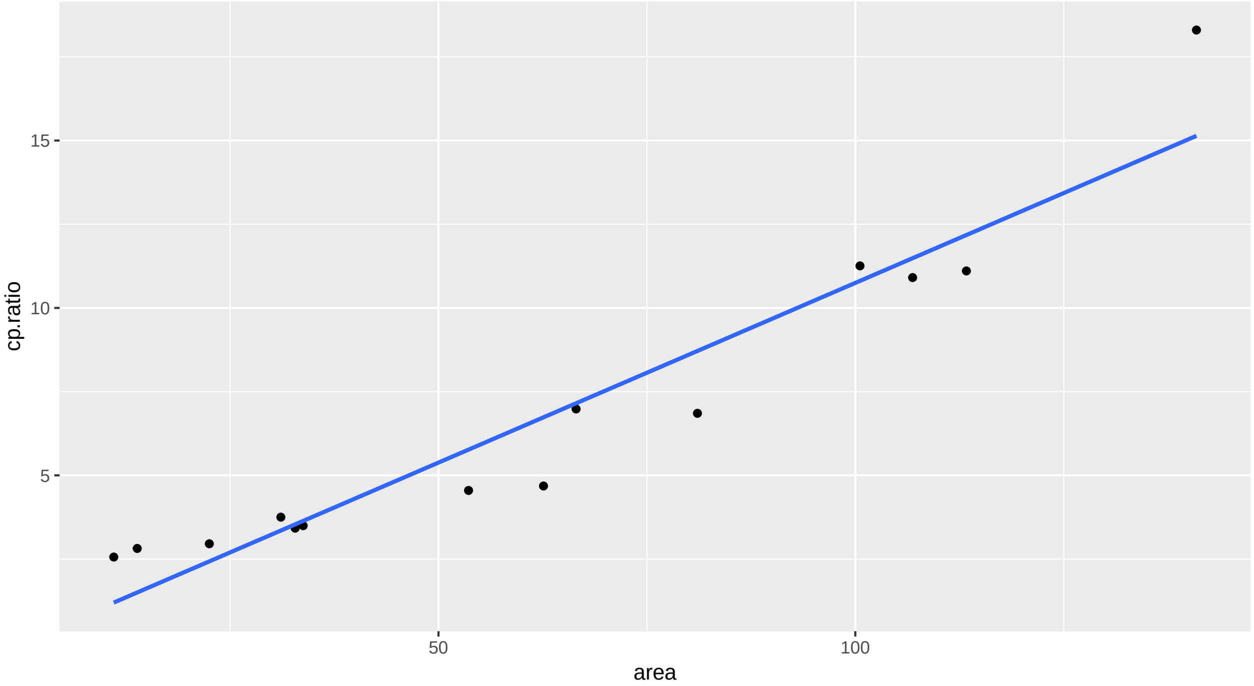
```
library(ggplot2)
load('./data/lattice.RData')
```

```
summary(ganglion)
```

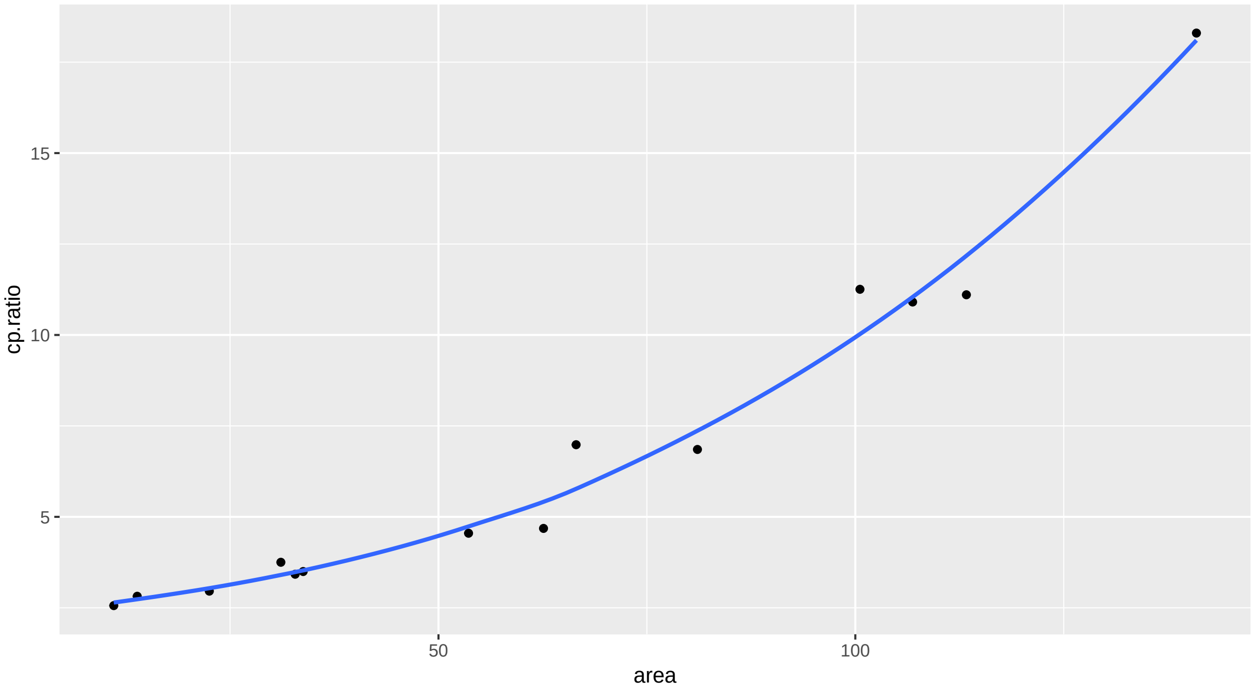
```
##       area           cp.ratio     
##  Min.   : 11.05   Min.   : 2.560  
##  1st Qu.: 31.52   1st Qu.: 3.441  
##  Median : 58.10   Median : 4.616  
##  Mean   : 62.18   Mean   : 6.688  
##  3rd Qu.: 95.68   3rd Qu.: 9.925  
##  Max.   :140.92   Max.   :18.300  
```

```
ganglion.gg = ggplot(ganglion, aes(x = area, y = cp.ratio)) +
  geom_point()
```

```
# Adding a Linear Curve
ganglion.gg + geom_smooth(method = 'lm', se = FALSE)
```



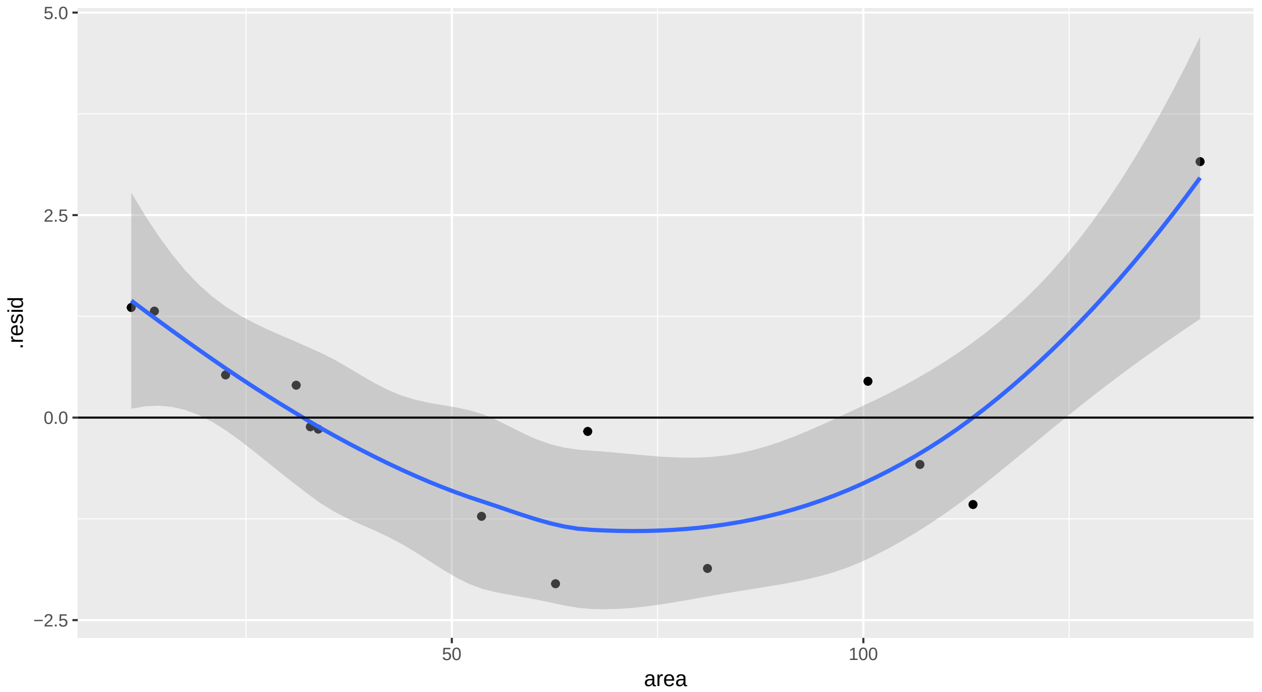
```
# Adding a Loess Curve
ganglion.gg + geom_smooth(method = 'loess', se = FALSE)
```



```r
ganglion.lm = lm(cp.ratio ~ area, data = ganglion)
# install.packages(broom)
library(broom)
gang.lm.df = augment(ganglion.lm)
summary(gang.lm.df)
```

```
##     cp.ratio          area          .fitted          .se.fit
##  Min.   : 2.560   Min.   : 11.05   Min.   : 1.200   Min.   :0.3851
##  1st Qu.: 3.441   1st Qu.: 31.52   1st Qu.: 3.397   1st Qu.:0.4386
##  Median : 4.616   Median : 58.10   Median : 6.251   Median :0.5132
##  Mean   : 6.688   Mean   : 62.18   Mean   : 6.688   Mean   :0.5303
##  3rd Qu.: 9.925   3rd Qu.: 95.68   3rd Qu.:10.284   3rd Qu.:0.6013
##  Max.   :18.300   Max.   :140.92   Max.   :15.139   Max.   :0.8580
##      .resid              .hat              .sigma            .cooksd
##  Min.   :-2.0509   Min.   :0.07144   Min.   :0.9262   Min.   :0.0004336
##  1st Qu.:-0.9493   1st Qu.:0.09286   1st Qu.:1.4356   1st Qu.:0.0065309
##  Median :-0.1277   Median :0.12714   Median :1.4770   Median :0.0249937
##  Mean   : 0.0000   Mean   :0.14286   Mean   :1.4258   Mean   :0.1871098
##  3rd Qu.: 0.5072   3rd Qu.:0.17421   3rd Qu.:1.4990   3rd Qu.:0.0870057
##  Max.   : 3.1603   Max.   :0.35459   Max.   :1.5044   Max.   :2.0477601
##    .std.resid
##  Min.   :-1.47721
##  1st Qu.:-0.73058
##  Median :-0.09395
##  Mean   : 0.03741
##  3rd Qu.: 0.38008
##  Max.   : 2.73029
```

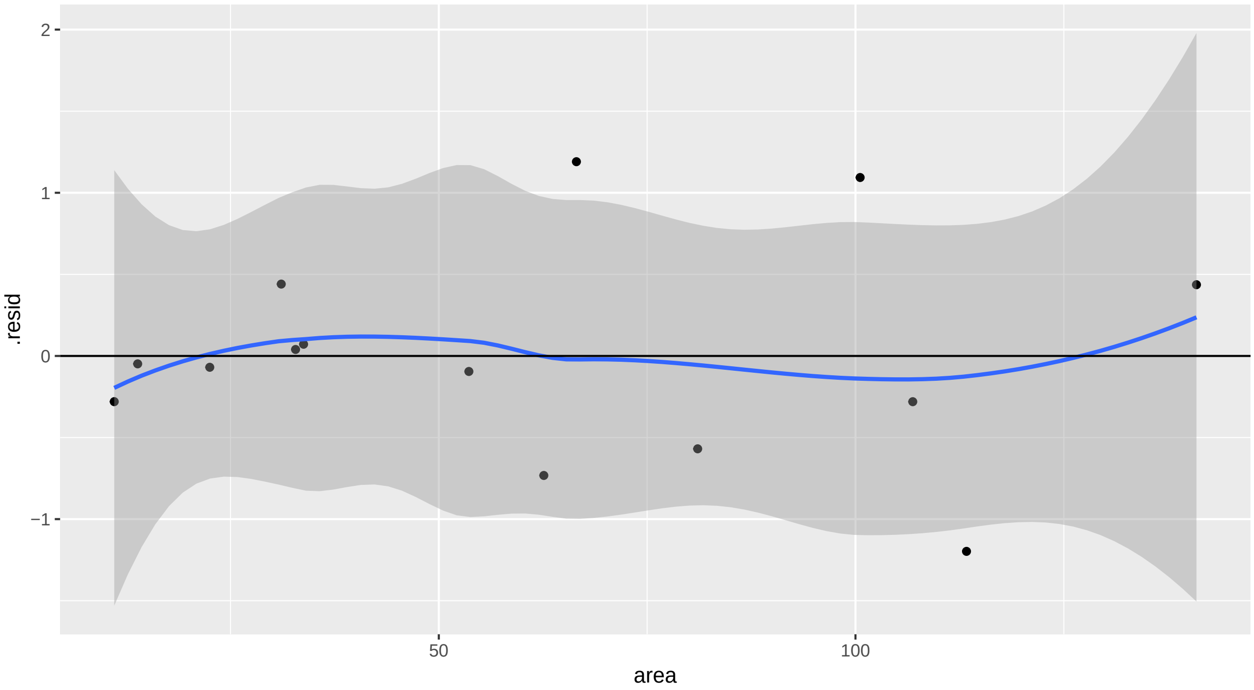
```r
# Does a linear model fit well for the data
ggplot(gang.lm.df, aes(x = area, y = .resid)) +
  geom_point() +
  geom_smooth(method = 'loess') +
  geom_abline(slope = 0, intercept = 0)
```



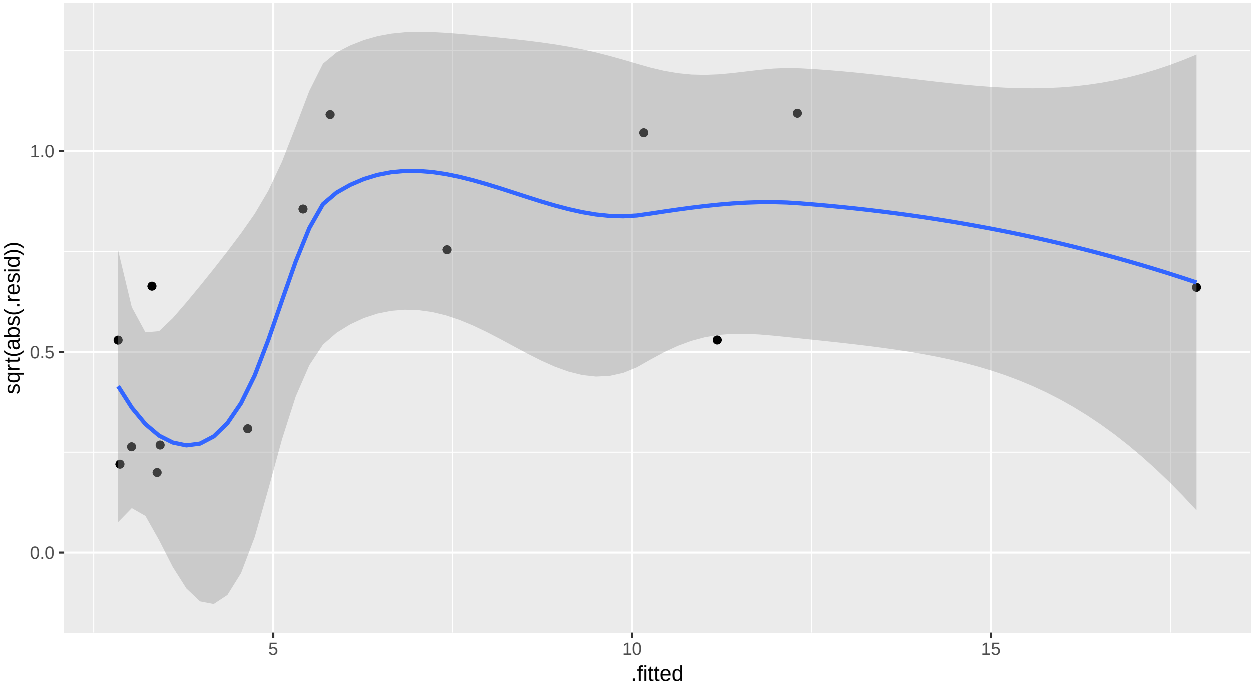
```r
# Does a quadratic model fit well for the data
ganglion.lm2 = lm(cp.ratio ~ area + I(area^2), data = ganglion)
gang.lm2.df = augment(ganglion.lm2)
summary(gang.lm2.df)
```

```
##     cp.ratio           area          I.area.2.          .fitted
##  Min.   : 2.560   Min.   : 11.05   Min.   :  122.1   Min.   : 2.840
##  1st Qu.: 3.441   1st Qu.: 31.52   1st Qu.:  994.3   1st Qu.: 3.329
##  Median : 4.616   Median : 58.10   Median : 3396.3   Median : 5.029
##  Mean   : 6.688   Mean   : 62.18   Mean   : 5430.5   Mean   : 6.688
##  3rd Qu.: 9.925   3rd Qu.: 95.68   3rd Qu.: 9226.8   3rd Qu.: 9.477
##  Max.   :18.300   Max.   :140.92   Max.   :19857.9   Max.   :17.863
##     .se.fit           .resid               .hat            .sigma
##  Min.   :0.2346   Min.   :-1.19758   Min.   :0.1107   Min.   :0.6081
##  1st Qu.:0.2683   1st Qu.:-0.28029   1st Qu.:0.1448   1st Qu.:0.6907
##  Median :0.2860   Median :-0.05893   Median :0.1644   Median :0.7283
##  Mean   :0.3130   Mean   : 0.00000   Mean   :0.2143   Mean   :0.7030
##  3rd Qu.:0.3051   3rd Qu.: 0.34532   3rd Qu.:0.1872   3rd Qu.:0.7393
##  Max.   :0.6052   Max.   : 1.19049   Max.   :0.7363   Max.   :0.7396
##     .cooksd            .std.resid
##  Min.   :0.0001503   Min.   :-1.88857
##  1st Qu.:0.0008978   1st Qu.:-0.47249
##  Median :0.0288280   Median :-0.09404
##  Mean   :0.1611383   Mean   : 0.02964
##  3rd Qu.:0.1569547   3rd Qu.: 0.52490
##  Max.   :1.3513989   Max.   : 1.84998
```

```r
ggplot(gang.lm2.df, aes(x = area, y = .resid)) +
  geom_point() +
  geom_smooth(method = 'loess') +
  geom_abline(slope = 0, intercept = 0)
```

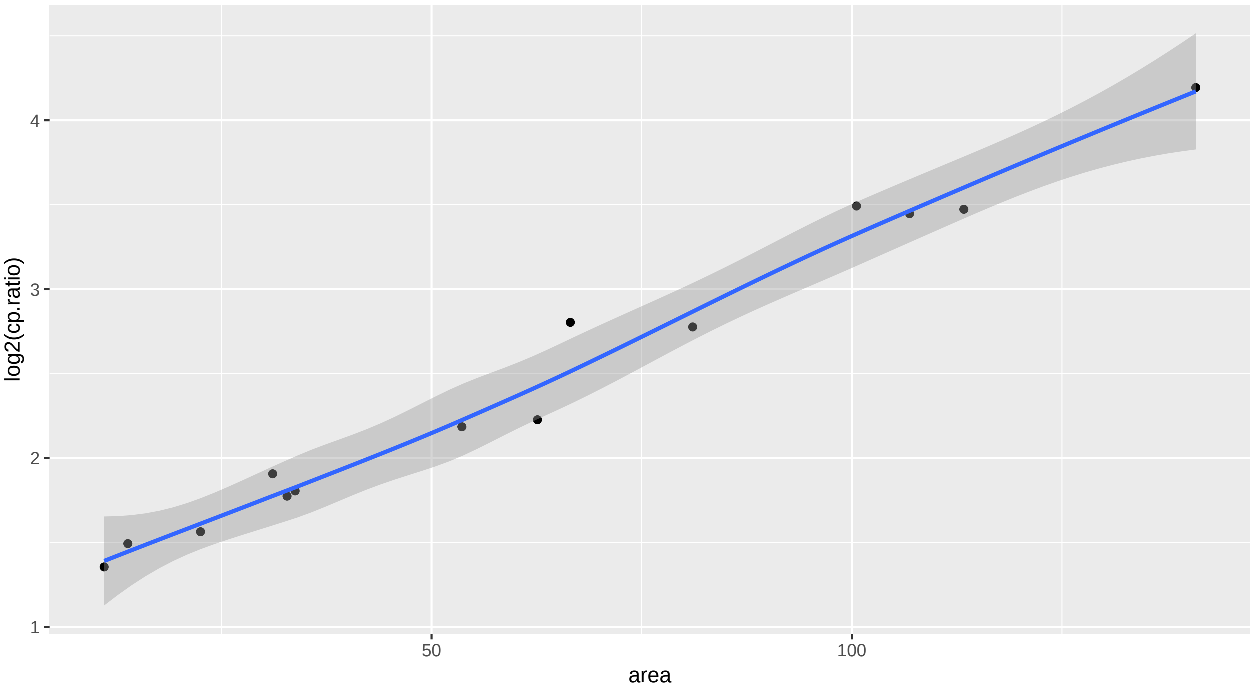


```
# Checking fit using a spread-location plot
ggplot(gang.lm2.df, aes(x = .fitted, y = sqrt(abs(.resid)))) +
  geom_point() +
  geom_smooth(method = 'loess')
```

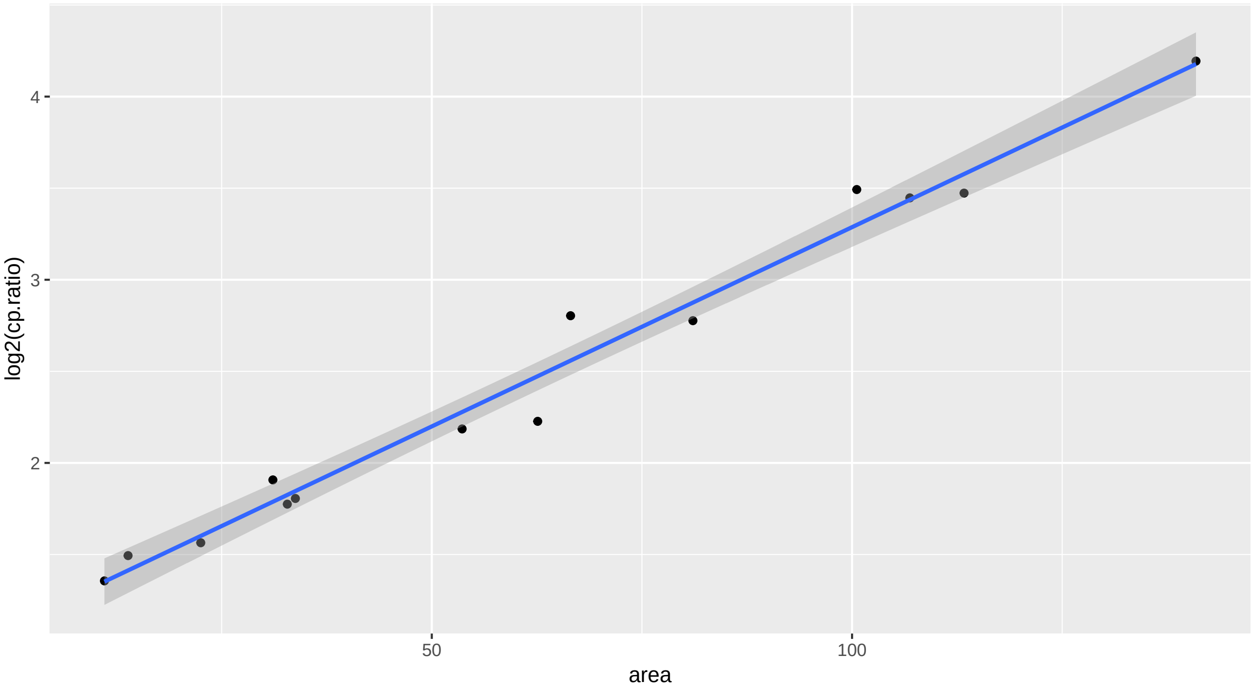


```
ggplot(ganglion, aes(x = area, y = log2(cp.ratio))) +
  geom_point() + geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

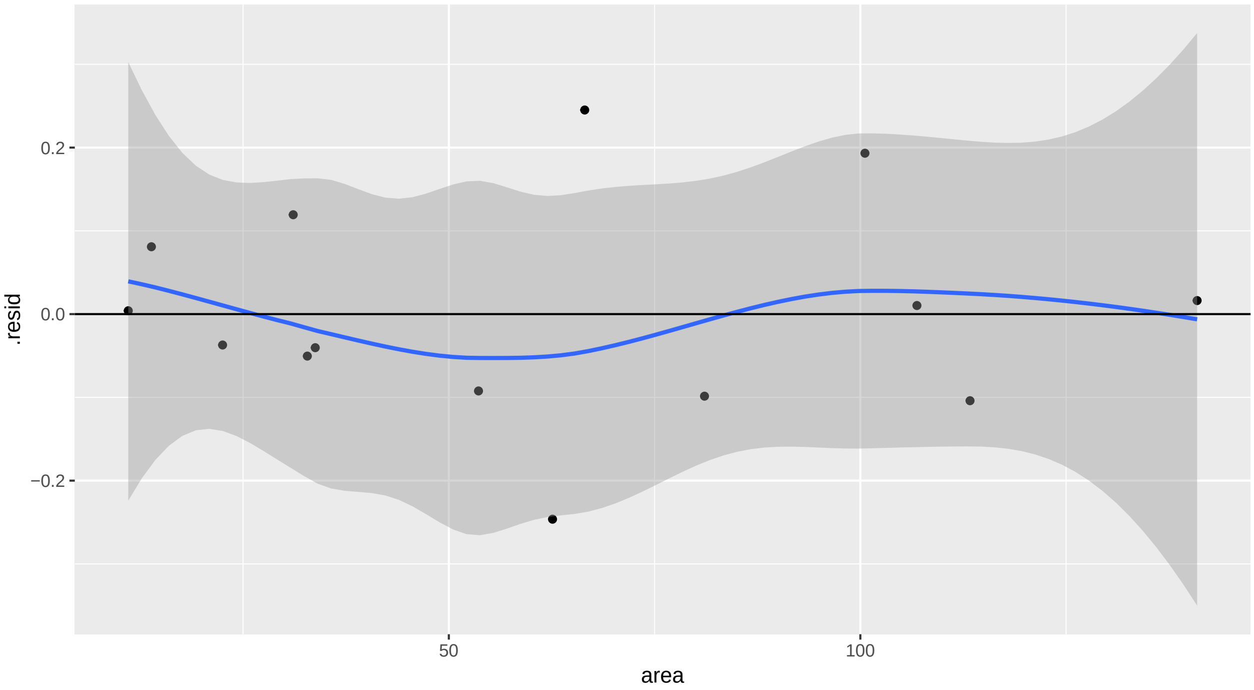


```
ggplot(ganglion, aes(x = area, y = log2(cp.ratio))) +
  geom_point() + geom_smooth(method = 'lm')
```



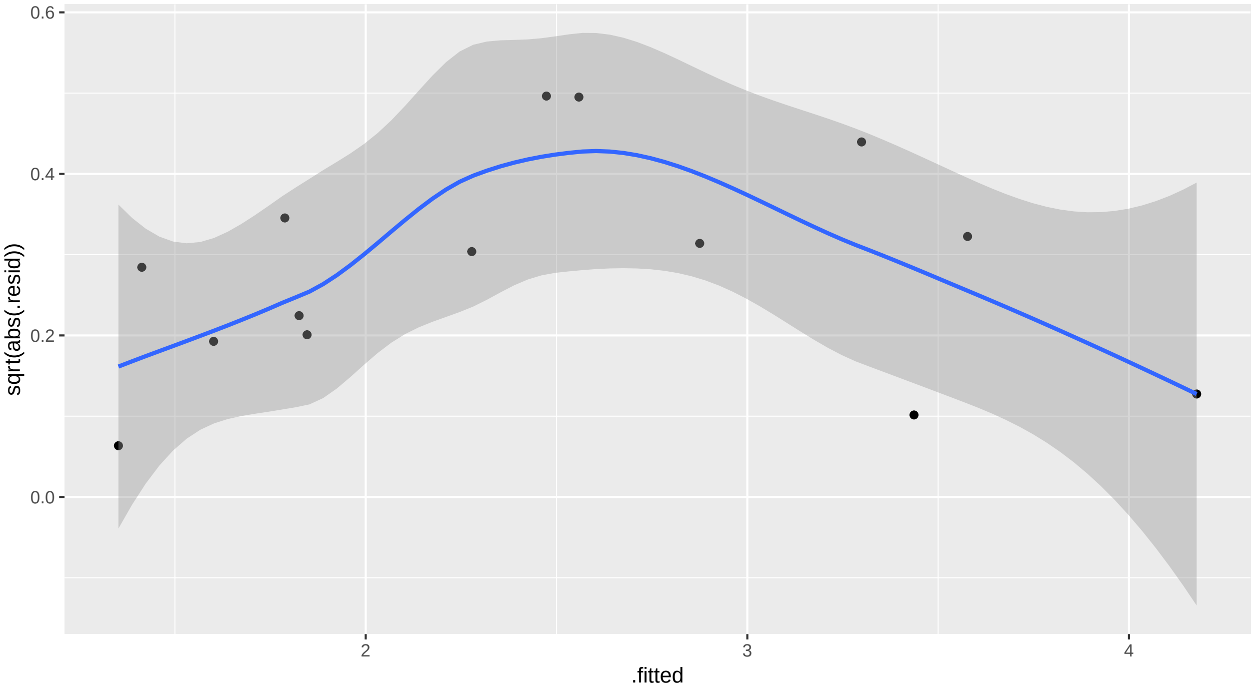
```
ganglion.log.lm = lm(log2(cp.ratio) ~ area, data = ganglion)
gang.log.lm.df = augment(ganglion.log.lm)
ggplot(gang.log.lm.df, aes(x = area, y = .resid)) +
  geom_point() + geom_smooth() +
  geom_abline(slope = 0, intercept = 0)
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



```r
# Better S-L Plot Using Transformation
ggplot(gang.log.lm.df, aes(x = .fitted, y = sqrt(abs(.resid)))) +
  geom_point() + geom_smooth()
```

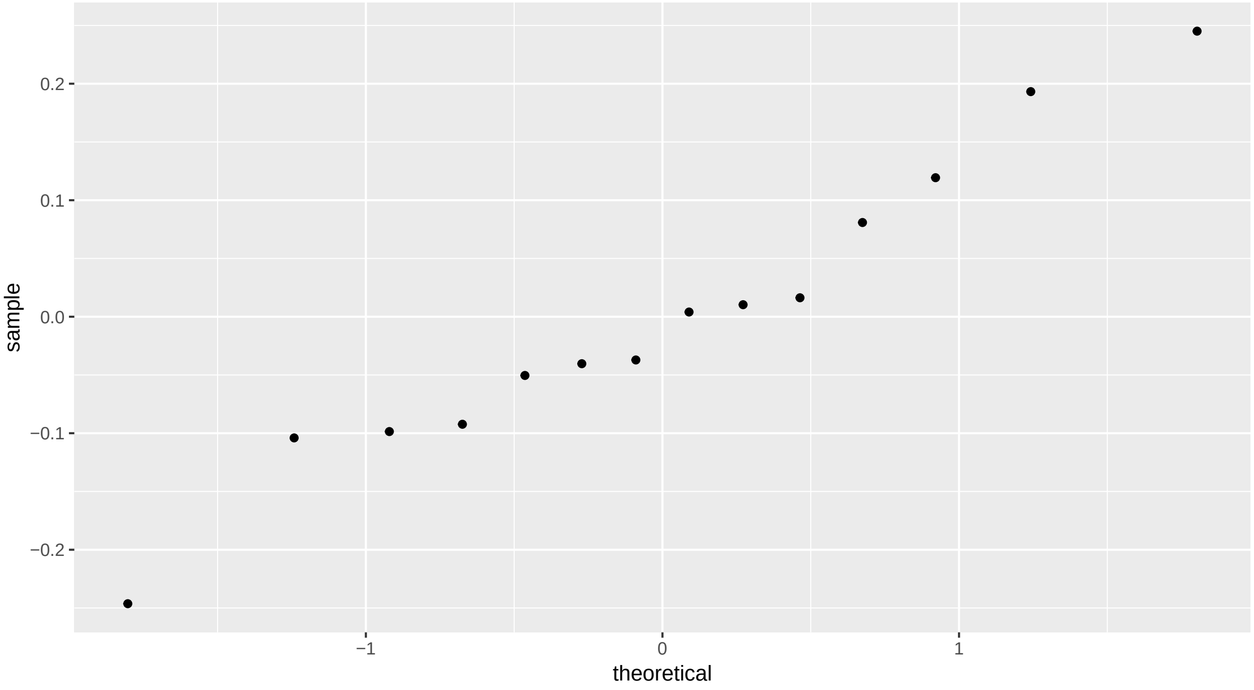
```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



```r
# Calculate R-Squared
round(var(gang.log.lm.df$.fitted)/var(log2(ganglion$cp.ratio)),3)
```

```
## [1] 0.98
```

```
# Checking for Normality in the Residuals
ggplot(gang.log.lm.df, aes(sample = .resid)) +
  stat_qq()
```


sample
theoretical