bonds

Chris Parrish

December 11, 2016

Contents

bonds 1
standardized residuals 2
regression diagnostics 3
exclude flower bonds 4
Cook's distance 6

bonds

reference:
- Sheather, A Modern Approach to Regression with R, chapter 3, pp.62-68
- label points in `geom_point`, Stack Overflow

Load packages.

```
library(ggplot2)
```

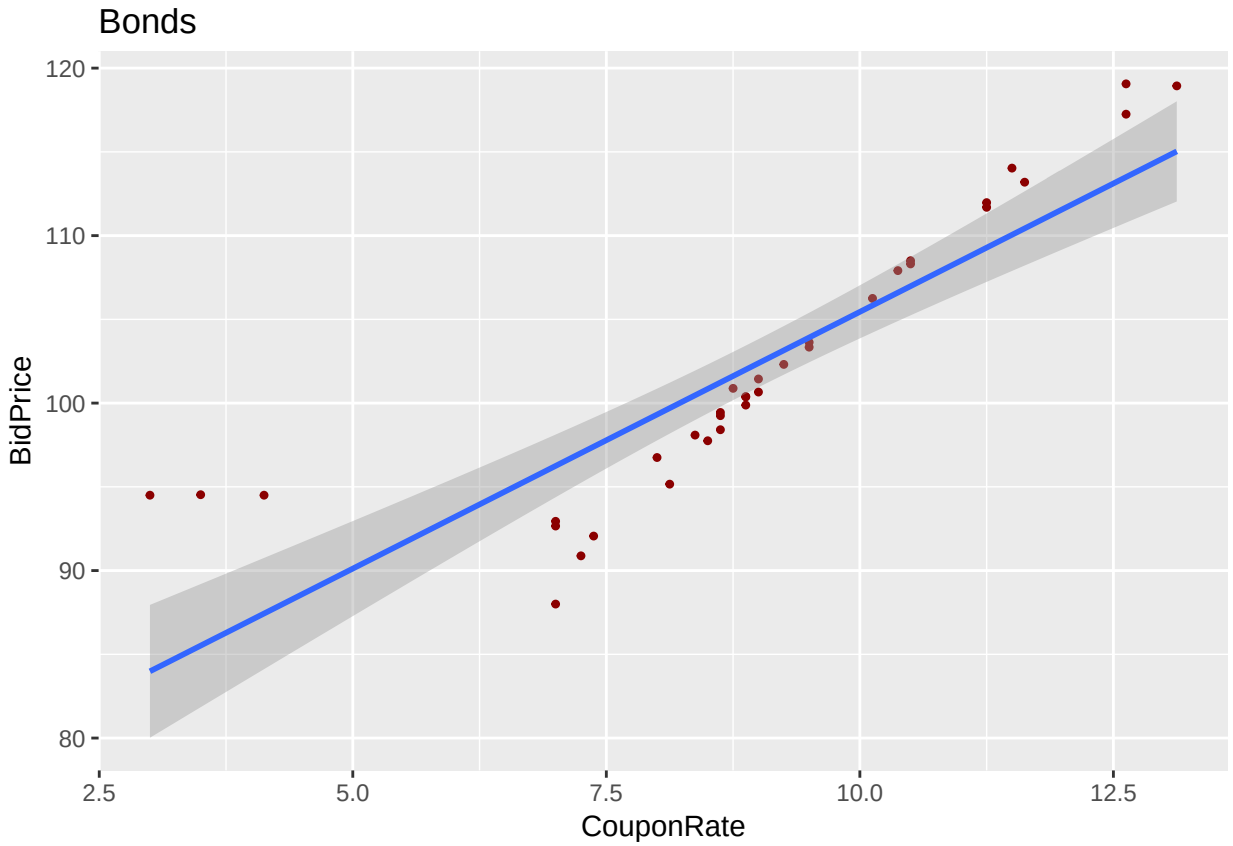
Import the data.

```
data <- read.table("bonds.txt", header = TRUE)
head(data)
```

```
##   Case CouponRate BidPrice
## 1    1      7.000    92.94
## 2    2      9.000   101.44
## 3    3      7.000    92.66
## 4    4      4.125    94.50
## 5    5     13.125   118.94
## 6    6      8.000    96.75
```

View the data.

```
ggplot(data, aes(CouponRate, BidPrice)) +
  geom_point(shape = 20, color = "darkred") +
  geom_smooth(method = "lm") +
  ggtitle("Bonds")
```



standardized residuals

```
bonds.lm <- lm(BidPrice ~ CouponRate, data = data)
options(show.signif.stars = FALSE)
summary(bonds.lm)
```

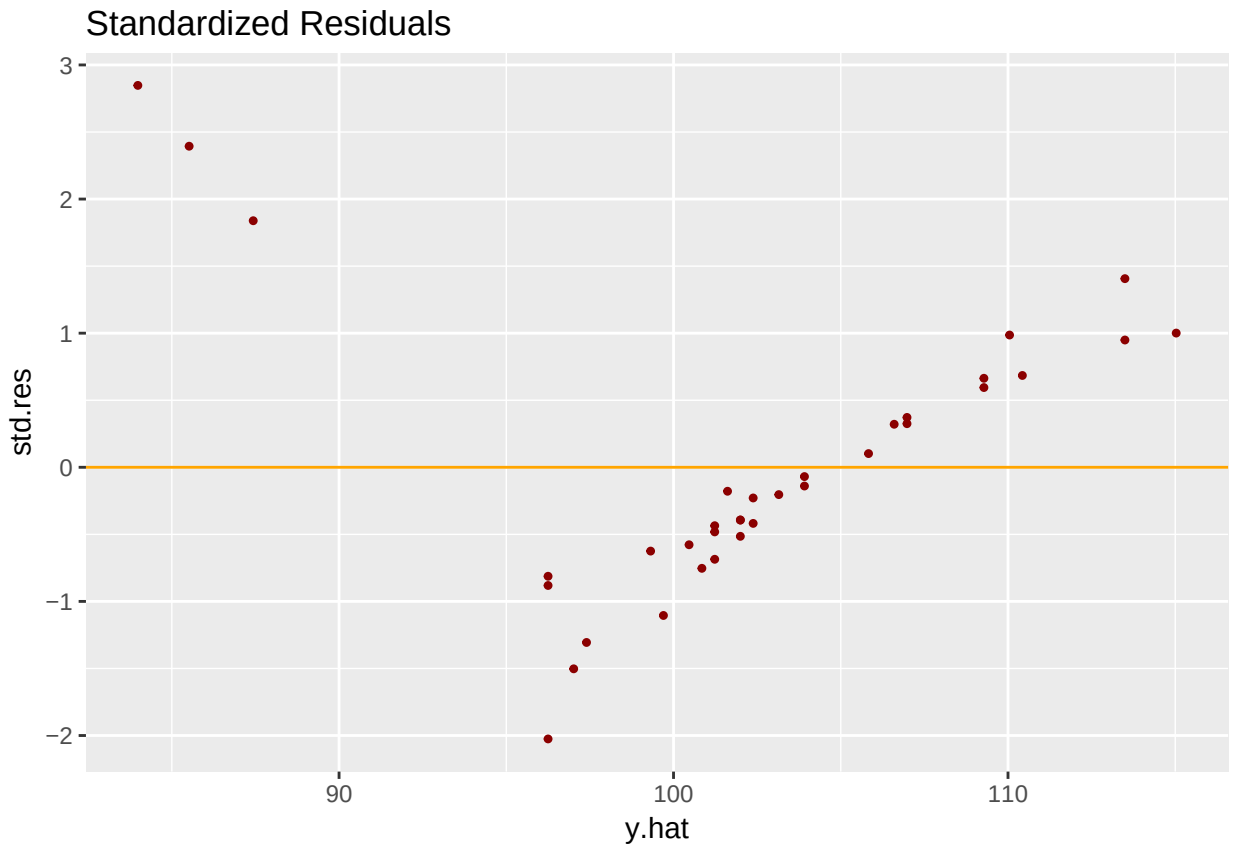
```
##
## Call:
## lm(formula = BidPrice ~ CouponRate, data = data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -8.249 -2.470 -0.838  2.550 10.515
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  74.7866     2.8267  26.458  < 2e-16
## CouponRate    3.0661     0.3068   9.994 1.64e-11
##
## Residual standard error: 4.175 on 33 degrees of freedom
## Multiple R-squared:  0.7516, Adjusted R-squared:  0.7441
## F-statistic: 99.87 on 1 and 33 DF,  p-value: 1.645e-11
```

```
confint(bonds.lm)
```

```
##                 2.5 %    97.5 %
```

```
## (Intercept) 69.035683 80.537437
## CouponRate   2.441906  3.690299
```

```
data.lm <- data.frame(case = data$Case,
                      y.hat <- predict(bonds.lm),
                      std.res <- rstandard(bonds.lm))
ggplot(data.lm, aes(y.hat, std.res)) +
  geom_point(shape = 20, color = "darkred") +
  geom_hline(yintercept = 0, color = "orange") +
  ggtitle("Standardized Residuals")
```



regression diagnostics

A point x_i is a *leverage point* if

$$h_{i,i} > 2 * average.leverage = 2 * \frac{2}{n}$$

```
n <- nrow(data)
average.leverage <- 2 / n
average.leverage
```

```
## [1] 0.05714286
```

```
tbl <- data.frame(coupon.rate = data$CouponRate,
                  bid.price = data$BidPrice,
                  leverage = hatvalues(bonds.lm),
                  residual = bonds.lm$residuals,
```

```
                  std.res = rstandard(bonds.lm))
tbl[tbl$leverage > 2 * average.leverage, ]
```

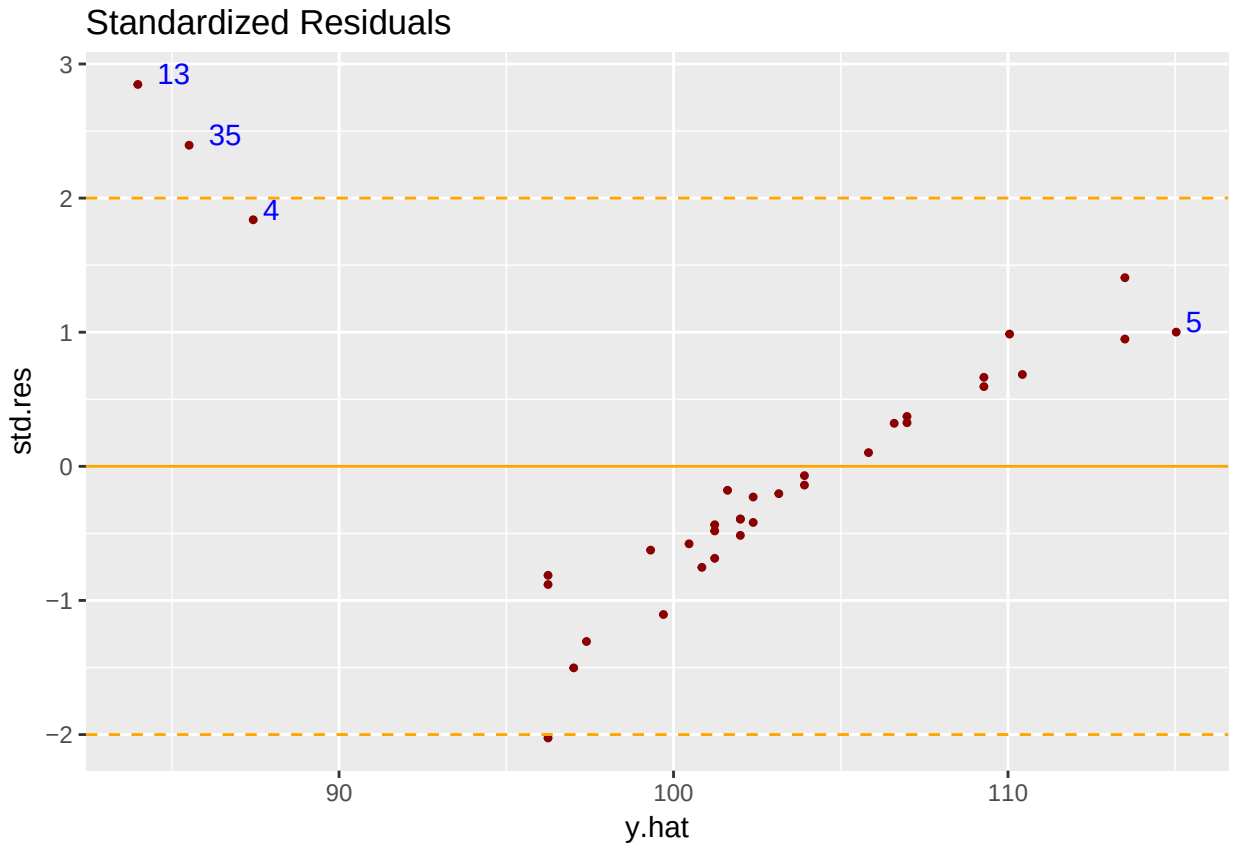
```
##    coupon.rate bid.price  leverage  residual  std.res
## 4        4.125     94.50 0.1527780  7.065769 1.838463
## 5       13.125    118.94 0.1239708  3.910849 1.000704
## 13       3.000     94.50 0.2178763 10.515133 2.847548
## 35       3.500     94.53 0.1872566  9.012082 2.394101
```

Identify high leverage points and outliers.

A point is an *outlier* if its standardized residual is > 2 or < -2.

A *bad leverage point* is a leverage point which is also an outlier.

```
ggplot(data.lm, aes(y.hat, std.res)) +
  geom_point(shape = 20, color = "darkred") +
  geom_hline(yintercept = 0, color = "orange") +
  geom_text(aes(label = ifelse(tbl$leverage > 2 * average.leverage,
                               as.character(case), '')),
                hjust = -0.6, vjust = 0, color = "blue") +
  geom_hline(yintercept = c(-2, 2), color = "orange", lty = 2) +
  ggtitle("Standardized Residuals")
```



exclude flower bonds

Bonds 13, 35, and 4 belong in a different category. Exclude them.

```r
data.mod <- data[-c(13, 35, 4), ]
head(data.mod)
```

```
##   Case CouponRate BidPrice
## 1    1      7.000    92.94
## 2    2      9.000   101.44
## 3    3      7.000    92.66
## 5    5     13.125   118.94
## 6    6      8.000    96.75
## 7    7      8.750   100.88
```

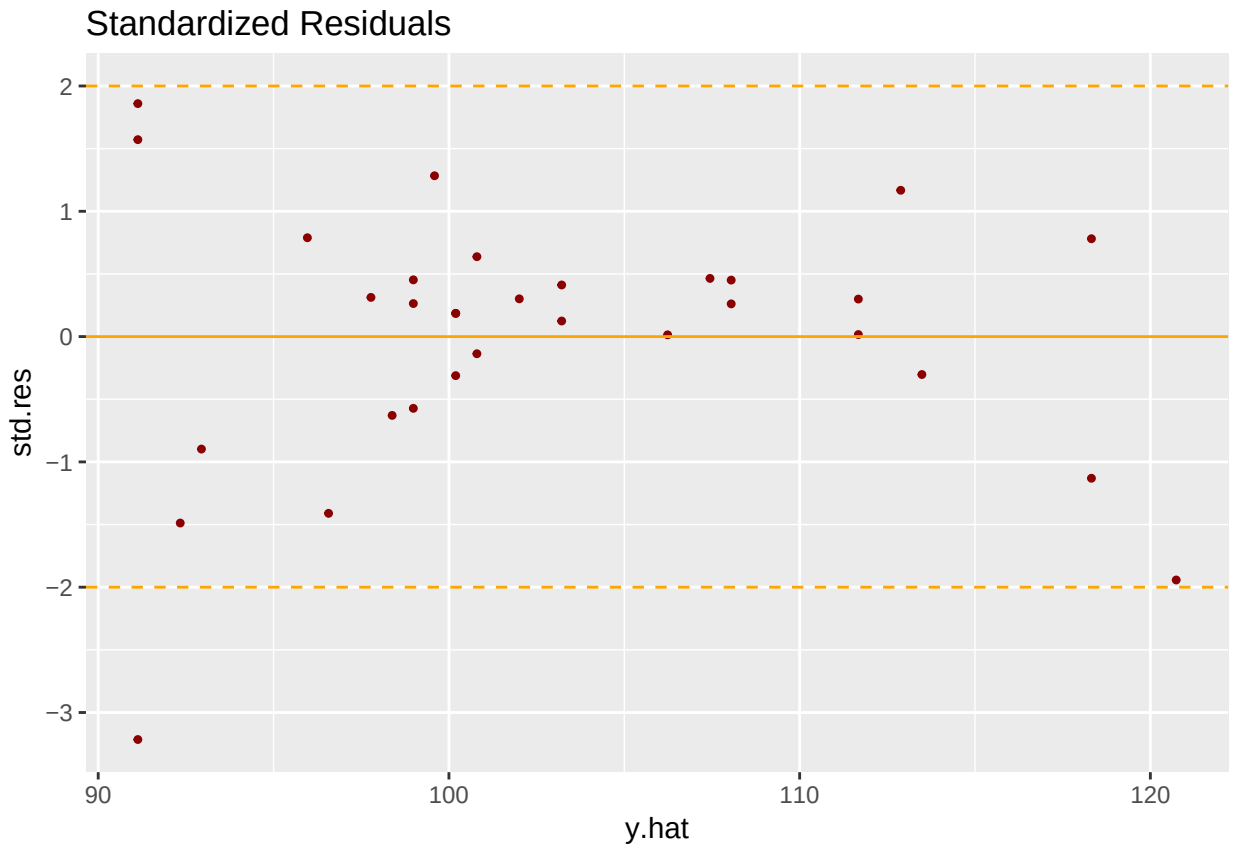
```r
bonds.lm2 <- lm(BidPrice ~ CouponRate, data = data.mod)
summary(bonds.lm2)
```

```
##
## Call:
## lm(formula = BidPrice ~ CouponRate, data = data.mod)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.1301 -0.3789  0.2240  0.4576  1.8099
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  57.2932     1.0358   55.31   <2e-16
## CouponRate    4.8338     0.1082   44.67   <2e-16
##
## Residual standard error: 1.024 on 30 degrees of freedom
## Multiple R-squared:  0.9852, Adjusted R-squared:  0.9847
## F-statistic:  1996 on 1 and 30 DF,  p-value: < 2.2e-16
```

```r
confint(bonds.lm2)
```

```
##                 2.5 %    97.5 %
## (Intercept) 55.177793 59.408659
## CouponRate   4.612865  5.054814
```

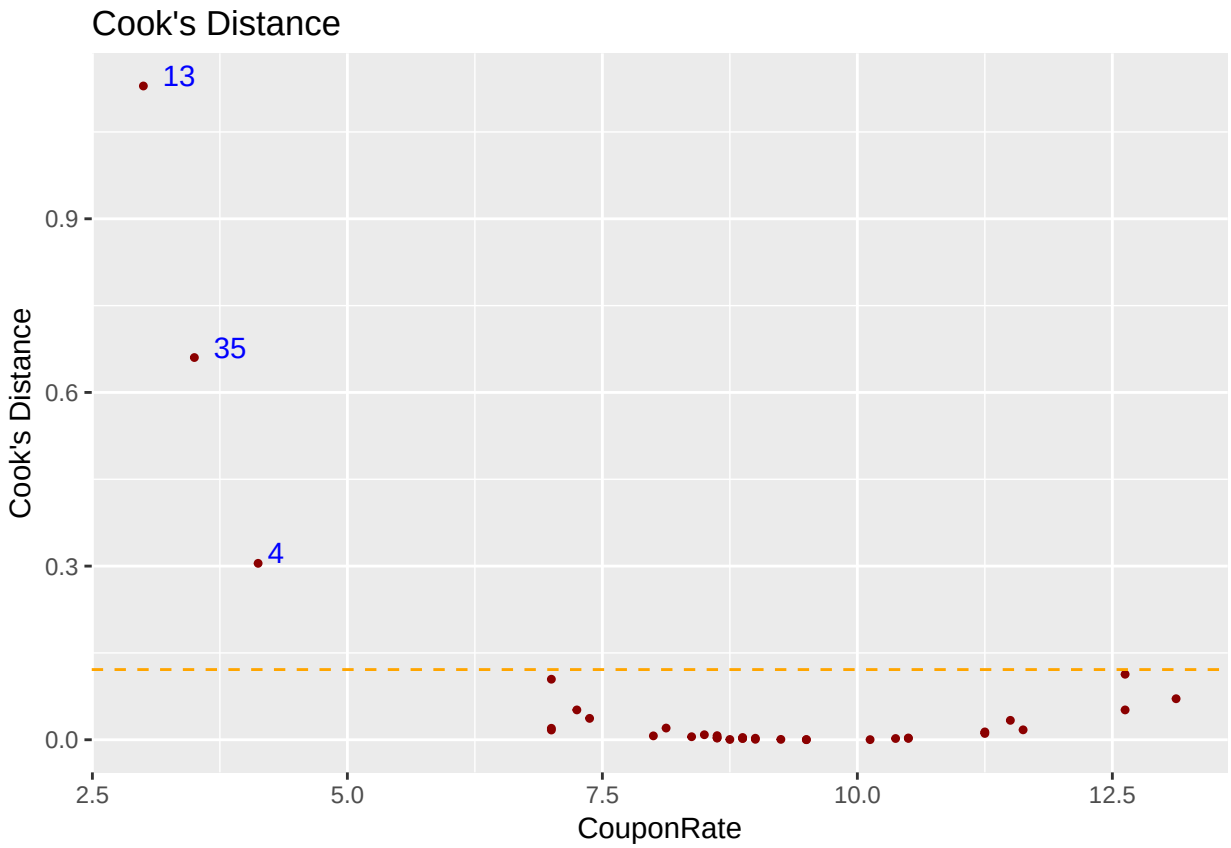
```r
data.lm2 <- data.frame(case = data.mod$Case,
                       y.hat <- predict(bonds.lm2),
                       std.res <- rstandard(bonds.lm2))
ggplot(data.lm2, aes(y.hat, std.res)) +
  geom_point(shape = 20, color = "darkred") +
  geom_hline(yintercept = 0, color = "orange") +
  geom_hline(yintercept = c(-2, 2), color = "orange", lty = 2) +
  ggtitle("Standardized Residuals")
```



Cook's distance

```
data$cd <- cooks.distance(bonds.lm)
cutoff <- 4 / (n - 2)
```

```
ggplot(data, aes(CouponRate, cd)) +
  geom_point(shape = 20, color = "darkred") +
  geom_hline(yintercept = cutoff, color = "orange", lty = 2) +
  geom_text(aes(label = ifelse(cd > cutoff,
                               as.character(Case), '')),
                hjust = -0.6, vjust = 0, color = "blue") +
  ylab("Cook's Distance") +
  ggtitle("Cook's Distance")
```


Cook's Distance
13
35
4
0.9
0.6
0.3
0.0
Cook's Distance
2.5
5.0
7.5
10.0
12.5
CouponRate