Day 9: Example

Data set: PalmBeach

Data set with number of votes for George W. Bush and Pat Buchanan in Florida counties for the 2000 US presidential election.

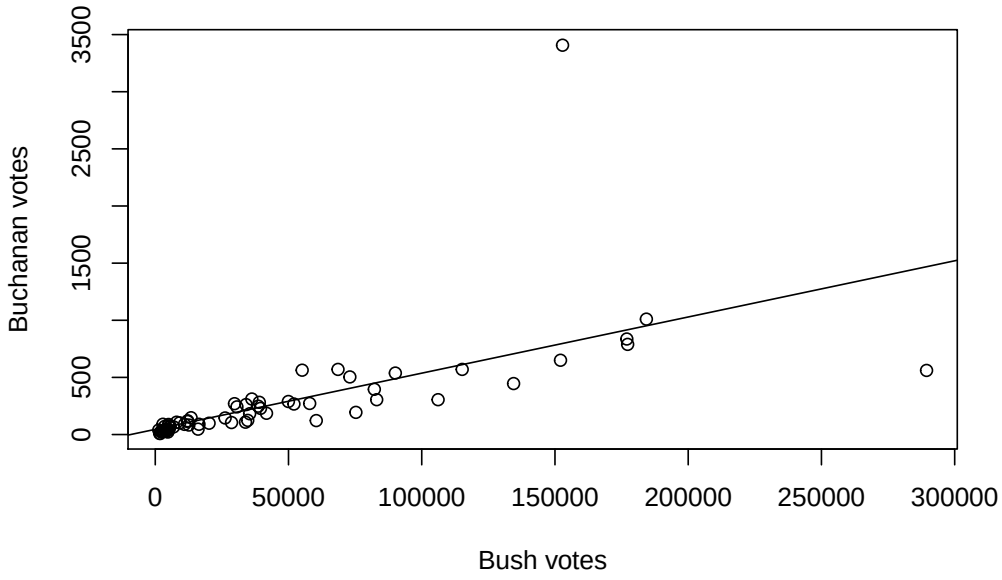
```
library(Stat2Data)
data(PalmBeach)
head(PalmBeach)
```

```
##     County Buchanan   Bush
## 1  ALACHUA      262  34062
## 2    BAKER       73   5610
## 3      BAY      248  38637
## 4 BRADFORD       65   5413
## 5  BREVARD      570 115185
## 6  BROWARD      789 177279
```

The race for the presidency of the United States in the fall of 2000 was very close, with the electoral votes from Florida determining the outcome. In the disputed final tally in Florida, George W. Bush won by just 537 votes over Al Gore, out of almost 6 million votes cast. About 2.3% of the votes cast in Florida were awarded to other candidates, including the Reform Party candidate Pat Buchanan. See the details in the R Documentation for more information.

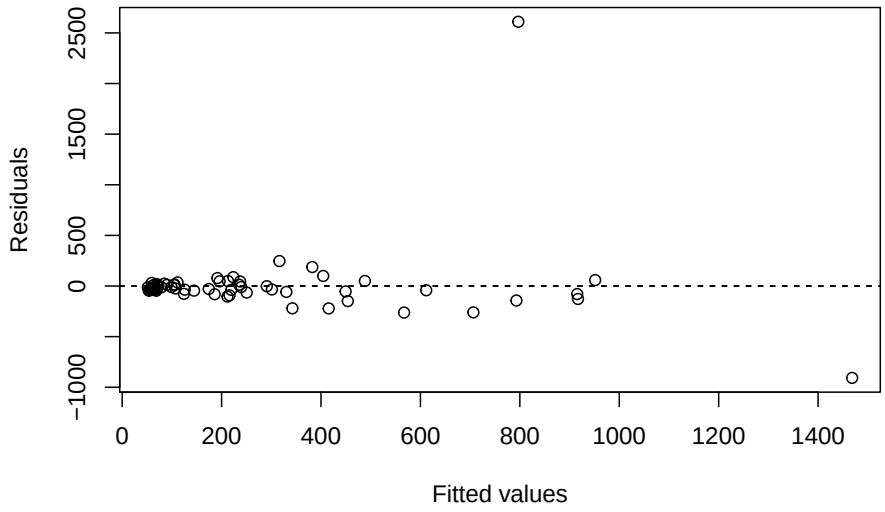
Scatter plot with the regression line

```
fit <- lm(Buchanan ~ Bush, data = PalmBeach)
plot(Buchanan ~ Bush, data = PalmBeach, ylab = "Buchanan votes", xlab = "Bush votes")
abline(fit)
```



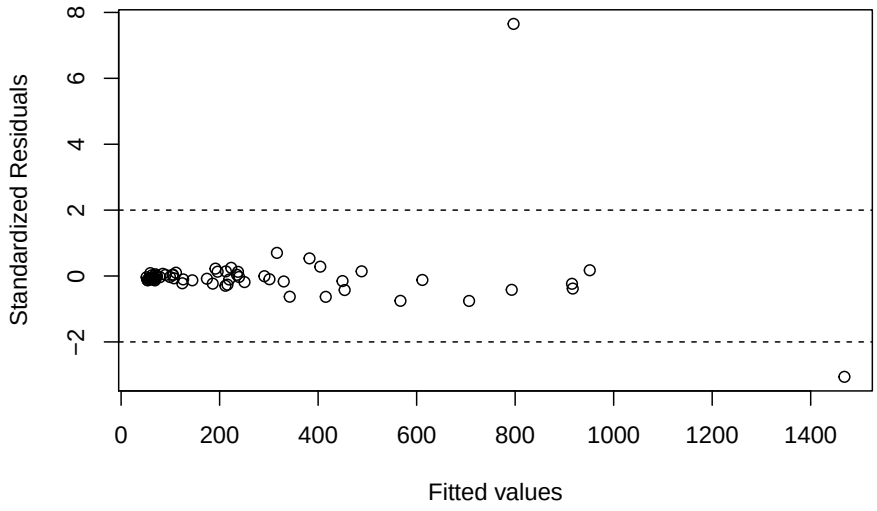
Residuals plot

```
plot(predict(fit), resid(fit), xlab = "Fitted values", ylab = "Residuals")
abline(h=c(-2,2), lty=2)
```



Standardized Residuals plot

```
plot(predict(fit), rstandard(fit), xlab = "Fitted values", ylab = "Standardized Residuals")
abline(h=c(-2,2), lty=2)
```



Identifying outliers

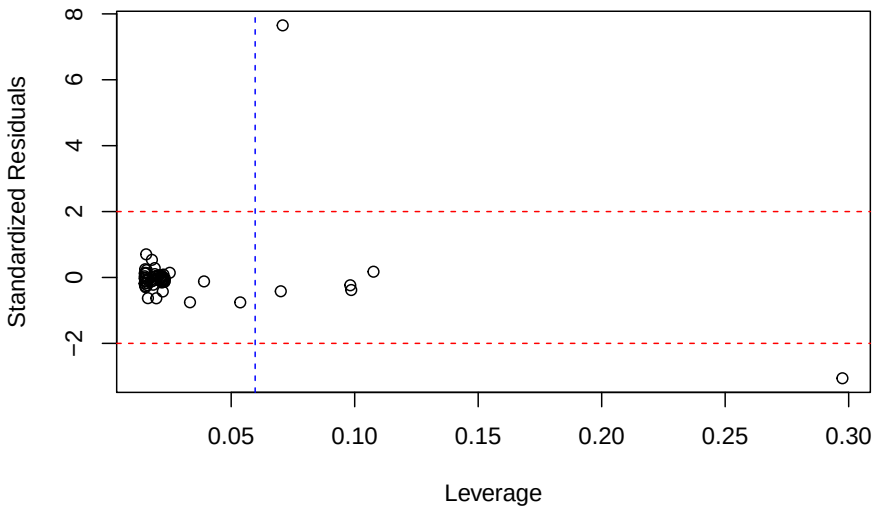
```
ind <- which(abs(rstandard(fit)) > 2)
PalmBeach[ind, ]
```

```
##         County Buchanan   Bush
## 13       DADE      561 289456
## 50 PALM BEACH     3407 152846
```

Palm Beach county used a unique "butterfly ballot," which had a layout that was confusing for many voters. Some voters that intended to vote for Al Gore mistakenly marked their ballots for Pat Buchanan.

Indetifying high leverage points

```
plot(hatvalues(fit), rstandard(fit), xlab = "Leverage", ylab = "Standardized Residuals")
n <- nrow(PalmBeach)
abline(v = 4/n, lty = 2, col = "blue") # threshold for high leverage
abline(h = c(-2,2), lty = 2, col = "red") # threshold for outliers
```



Observation resampling

```
library(boot)

boot.fl <- function(data, indices){
  data <- data[indices,] # select obs. in bootstrap sample
  mod <- lm(Buchanan ~ Bush, data = data)
  coefficients(mod) # return coefficient vector
}

fl.boot <- boot(PalmBeach, boot.fl, 5000)

plot(fl.boot, index = 1)
plot(fl.boot, index = 2)

confint(fit) # Theoretical CI
```

```
##                     2.5 %       97.5 %
## (Intercept) -63.513059697 1.540928e+02
## Bush          0.003390142 6.443514e-03
```

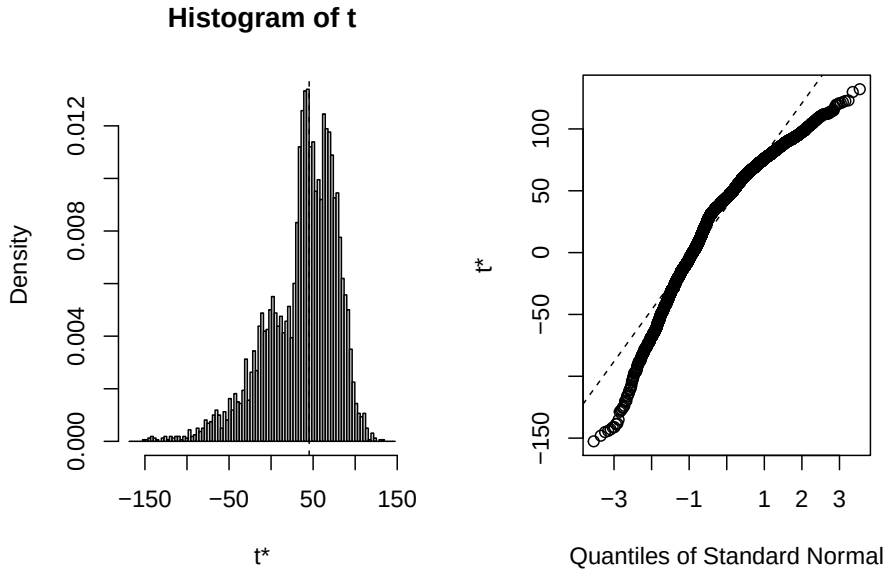
```
# Observation resampling
boot.ci(fl.boot, index = 1, type = c("norm", "perc", "bca"))
```

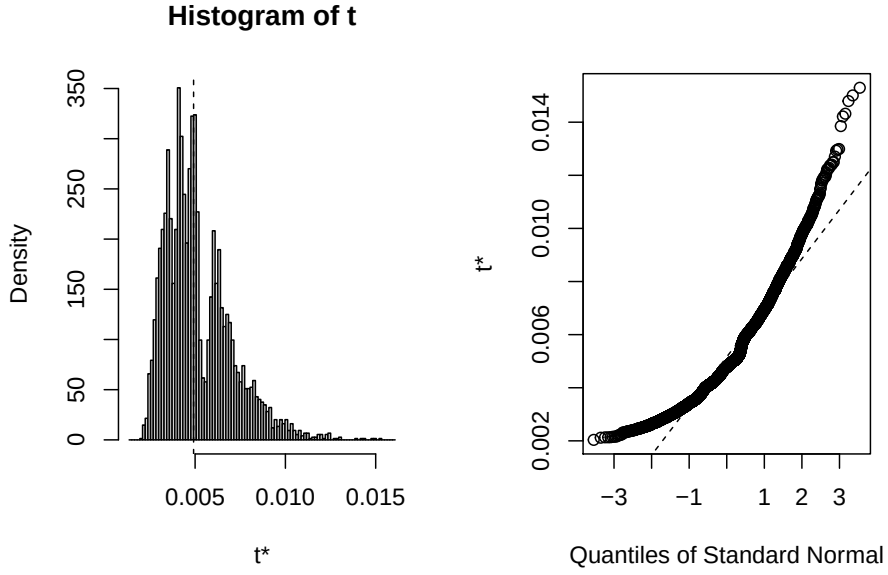
```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 5000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = fl.boot, type = c("norm", "perc", "bca"),
##     index = 1)
##
```

```
## Intervals :
## Level      Normal             Percentile            BCa
## 95%   (-28.55, 135.18 )   (-65.66,  96.49 )   (-49.07, 103.09 )
## Calculations and Intervals on Original Scale
```

```
boot.ci(fl.boot, index = 2, type = c("norm", "perc", "bca"))
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 5000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = fl.boot, type = c("norm", "perc", "bca"),
##     index = 2)
##
## Intervals :
## Level      Normal             Percentile            BCa
## 95%   ( 0.0010,  0.0083 )   ( 0.0027,  0.0097 )   ( 0.0029,  0.0113 )
## Calculations and Intervals on Original Scale
```





Residual resampling

```
fits <- fitted(fit)
e <- residuals(fit)
X <- model.matrix(fit)

boot.fl.fixed = function(data, indices) {
  y_b <-  fits + e[indices]
  mod <- lm(y_b ~ X - 1)
  coefficients(mod)
}

fl.fixed.boot <- boot(PalmBeach, boot.fl.fixed, 5000)
fl.fixed.boot
```

```
##
## ORDINARY NONPARAMETRIC BOOTSTRAP
##
##
## Call:
## boot(data = PalmBeach, statistic = boot.fl.fixed, R = 5000)
##
##
## Bootstrap Statistics :
##         original        bias     std. error
## t1* 45.289861271  1.541772e+00 5.384839e+01
## t2*  0.004916828 -4.925587e-06 7.464135e-04
```

```
plot(fl.fixed.boot, index = 1)
plot(fl.fixed.boot, index = 2)

confint(fit)# Theoretical CI
```

```
##                     2.5 %       97.5 %
## (Intercept) -63.513059697 1.540928e+02
## Bush          0.003390142 6.443514e-03
```

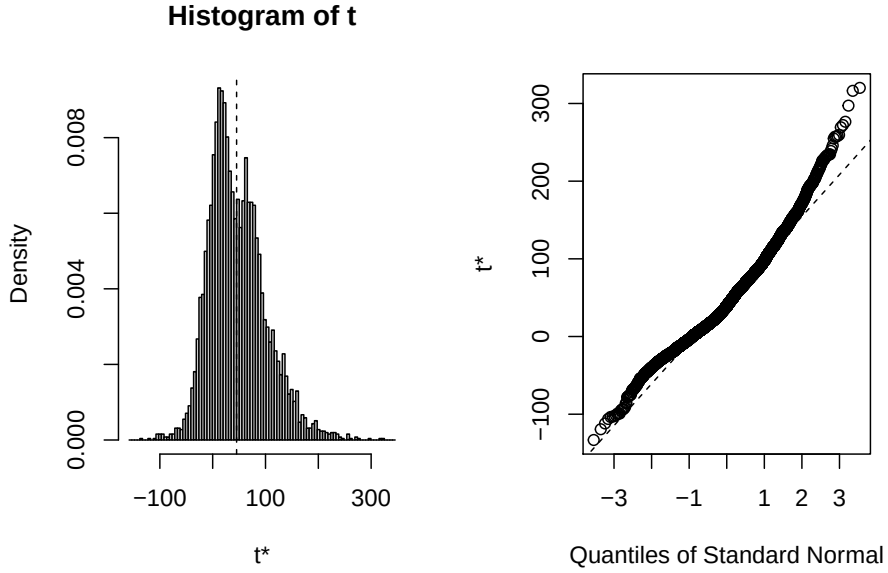
```
# Residual resampling
boot.ci(fl.fixed.boot, index = 1, type = c("norm", "perc", "bca"))
```

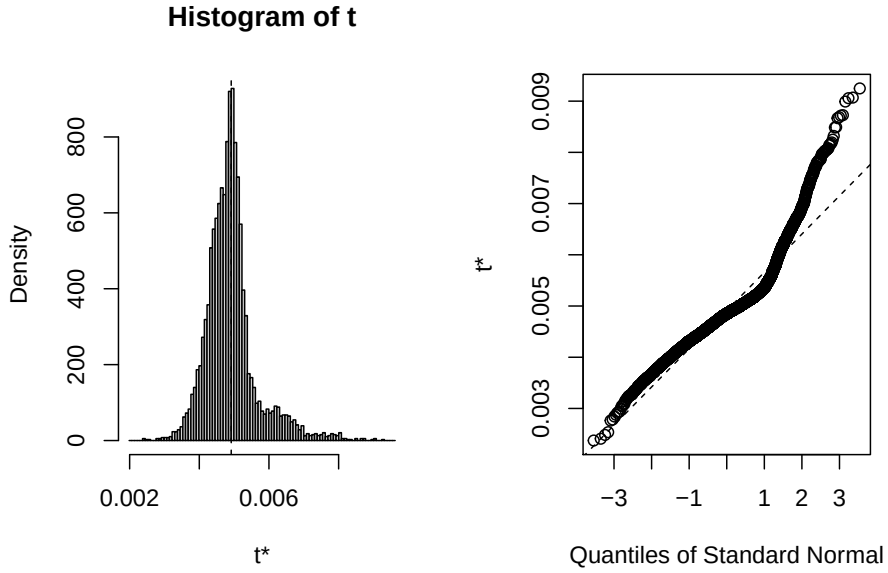
```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 5000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = fl.fixed.boot, type = c("norm", "perc", "bca"),
##     index = 1)
##
## Intervals :
## Level      Normal              Percentile            BCa
## 95%   (-61.79, 149.29 )   (-38.65, 168.72 )   (-19.69, 240.91 )
## Calculations and Intervals on Original Scale
```

```
boot.ci(fl.fixed.boot, index = 2, type = c("norm", "perc", "bca"))
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 5000 bootstrap replicates
##
```

```
## CALL :
## boot.ci(boot.out = fl.fixed.boot, type = c("norm", "perc", "bca"),
##     index = 2)
##
## Intervals :
## Level      Normal              Percentile            BCa
## 95%   ( 0.0035,  0.0064 )   ( 0.0037,  0.0069 )   ( 0.0039,  0.0075 )
## Calculations and Intervals on Original Scale
```





comparison

```
confint(fit)
```

```
##                     2.5 %       97.5 %
## (Intercept) -63.513059697 1.540928e+02
## Bush          0.003390142 6.443514e-03
```

```r
boot.ci(fl.boot, index = 2, type = c("norm", "perc", "bca"))
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 5000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = fl.boot, type = c("norm", "perc", "bca"),
##     index = 2)
##
## Intervals :
## Level      Normal              Percentile            BCa
## 95%   ( 0.0010,  0.0083 )   ( 0.0027,  0.0097 )   ( 0.0029,  0.0113 )
## Calculations and Intervals on Original Scale
```

```r
boot.ci(fl.fixed.boot, index = 2, type = c("norm", "perc", "bca"))
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 5000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = fl.fixed.boot, type = c("norm", "perc", "bca"),
##     index = 2)
##
## Intervals :
## Level      Normal              Percentile            BCa
## 95%   ( 0.0035,  0.0064 )   ( 0.0037,  0.0069 )   ( 0.0039,  0.0075 )
## Calculations and Intervals on Original Scale
```