# Bootstrapping Regression Models

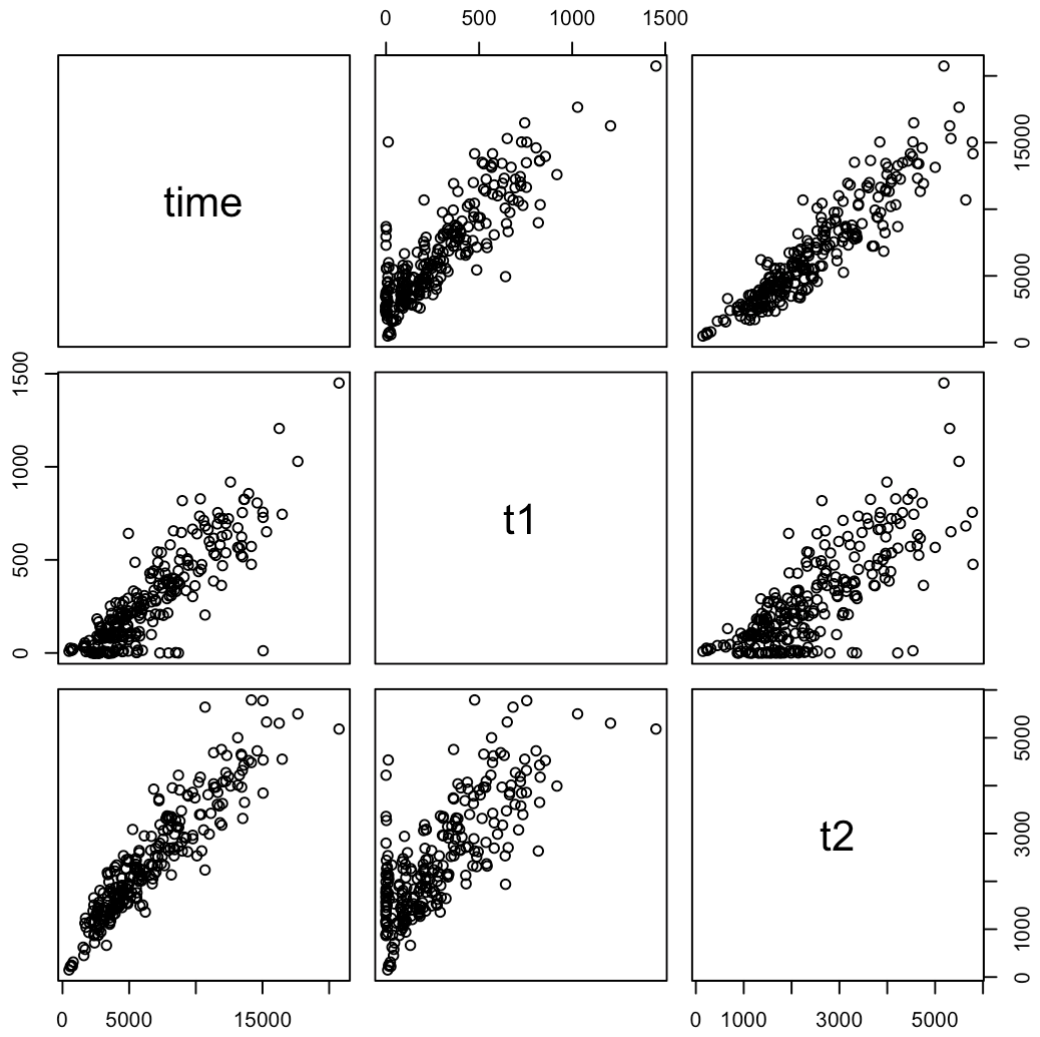
Math 430, Winter 2017

# Motivation

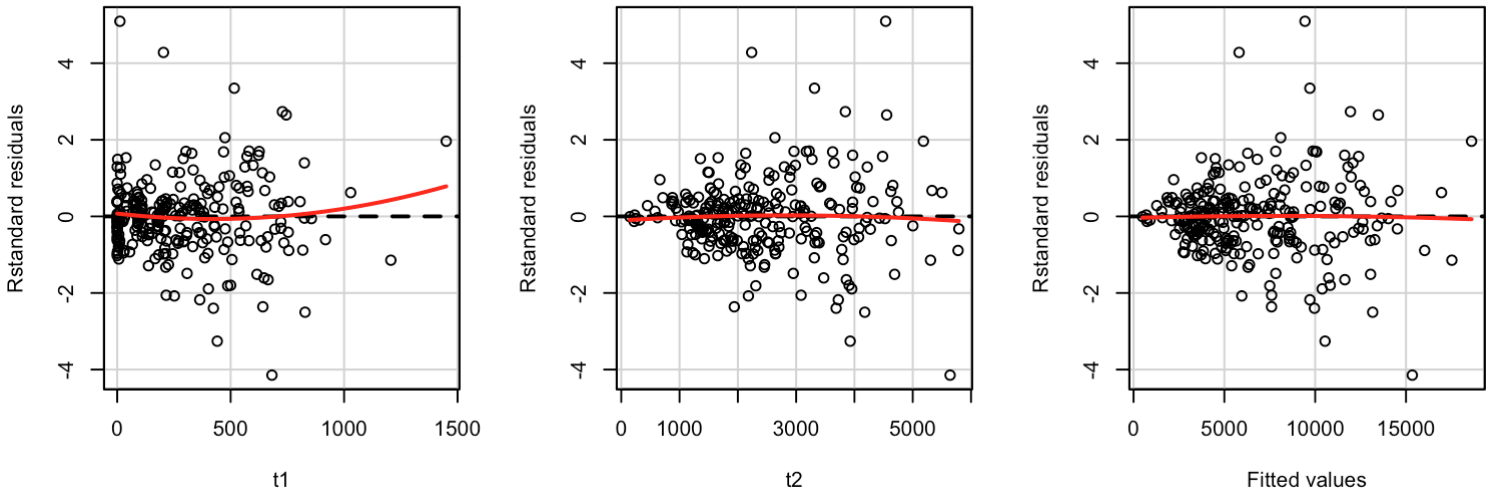
# Transactions data

Data on transaction times in branch offices of a large Australian bank.

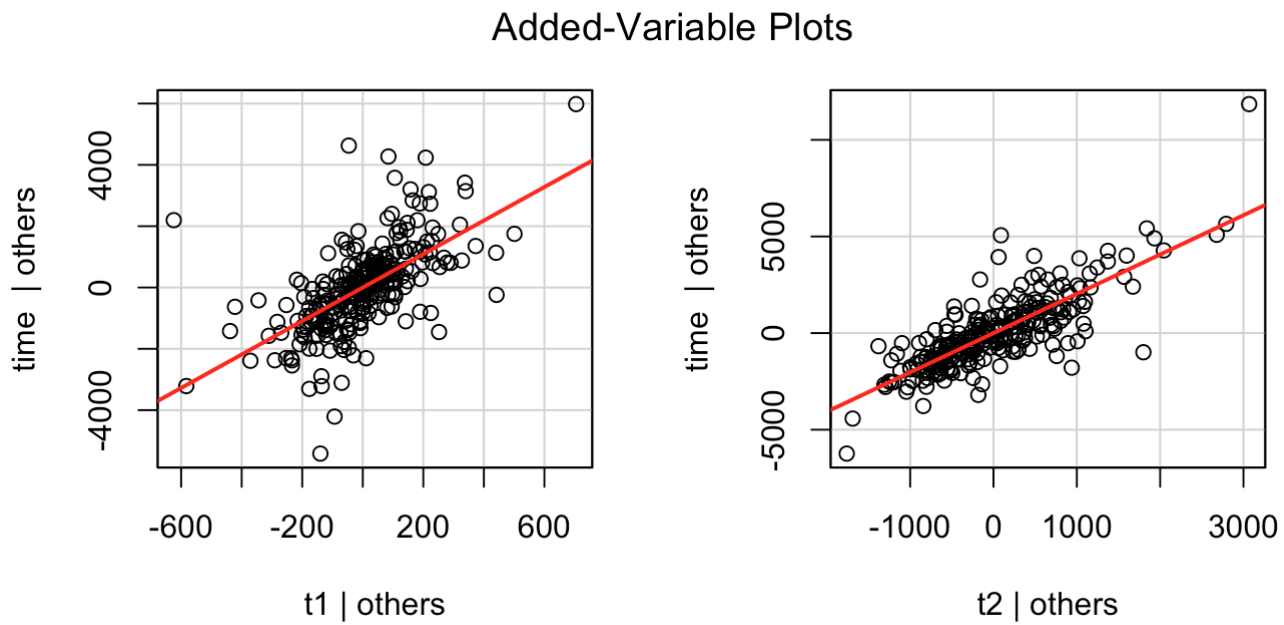
| Variable | Description |
|---|---|
| `time` | total transaction time (minutes) |
| `t1` | # type 1 transactions |
| `t2` | # type 2 transactions |


time
t1
t2
0
500
1000
1500
0
5000
15000
0
1000
3000
5000

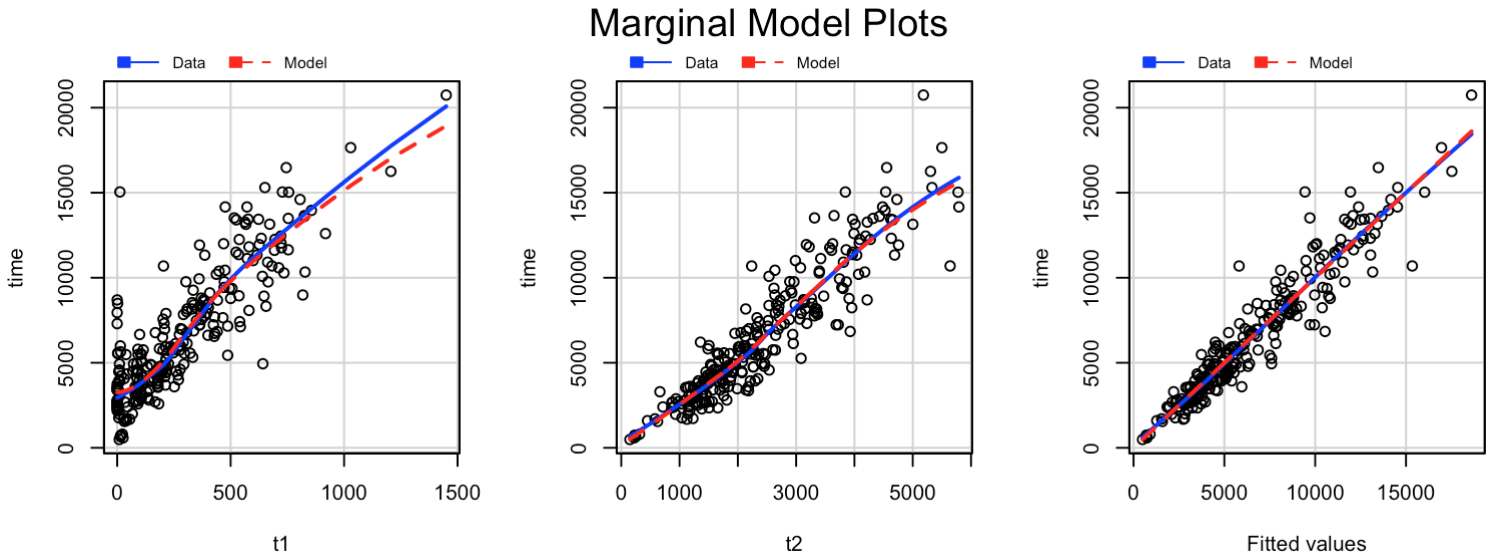
```
transact_mod <- lm(time ~ t1 + t2, data = Transact)
residualPlots(transact_mod, type = "rstandard",
              layout = c(1, 3), tests = FALSE)
```



```
avPlots(transact_mod,layout = c(1,2))
```



```
mmps(transact_mod, layout = c(1,3))
```



```
ncvTest(transact_mod)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 61.65942    Df = 1     p = 4.083091e-15
```

```
ncvTest(transact_mod, ~ t1)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ t1
## Chisquare = 26.52501    Df = 1     p = 2.601485e-07
```

```
ncvTest(transact_mod, ~ t2)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ t2
## Chisquare = 76.5892    Df = 1     p = 2.104878e-18
```

# The Bootstrap

# Why bootstrap?

- We need an alternative method when the assumptions are suspect, or where standard methods are not readily available
- Can be used to
  - compute standard errors
  - compute confidence intervals
  - conduct tests

# Case resampling bootstrap

1. Number cases in data set from 1 to $n$.
2. Take a random sample with replacement of size $n$ from these numbers.
3. Create a new data set by pulling the rows (cases) from the original data set that were selected in the random sample.
4. Fit the regression model to this new data set and save the values of the estimated coefficients, or other summary statistics.
5. Repeat steps 2-4 times.

We can build confidence intervals from this list of sets of coefficients.

# Case resampling bootstrap in R

```
library(car) # For Boot function
transact_boot <- Boot(transact_mod, R = 1999, method = "case")
summary(transact_boot)
```

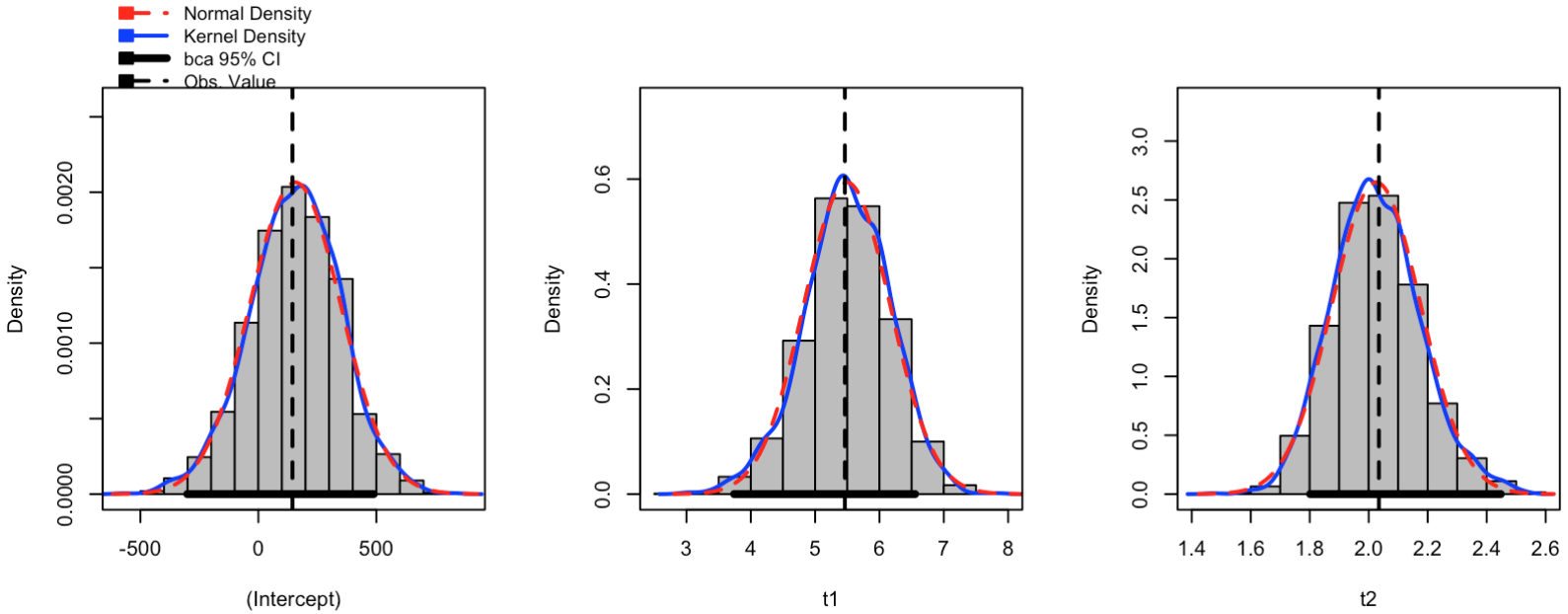
```
##                R original   bootBias    bootSE  bootMed
## (Intercept) 1999 144.3694 11.9775343 193.14915 161.9674
## t1          1999   5.4621  0.0321454   0.67197   5.5006
## t2          1999   2.0345 -0.0090864   0.15059   2.0175
```

```
# Compare to the original fit
broom::tidy(transact_mod)
```

```
##          term   estimate    std.error  statistic      p.value
## 1 (Intercept) 144.369443 170.54410348  0.8465226 3.980457e-01
## 2          t1   5.462057   0.43326792 12.6066488 1.031784e-28
## 3          t2   2.034549   0.09433682 21.5668576 1.123799e-59
```

# Case resampling bootstrap in R

```
hist(transact_boot, col = "gray", layout = c(1,3))
```



# Bootstrap confidence intervals

**Method 1:** Normal

$$\text{Estimate} \pm z^*_{\alpha/2} SE$$

where we use the bias corrected estimate: $statistic - \widehat{bias}$

```
confint(transact_boot, level = .95, type = "norm")
```

```
## Bootstrap quantiles, type =  normal
##
##                   2.5 %     97.5 %
## (Intercept) -246.173475 510.957292
## t1             4.112867   6.746956
## t2             1.748487   2.338784
```

# Bootstrap confidence intervals

**Method 2:** CIs based on percentiles of the bootstrap distribution

$$T^*_{(\text{lower})} \quad \text{to} \quad T^*_{(\text{upper})}$$

```
confint(transact_boot, level = .95, type = "perc")
```

```
## Bootstrap quantiles, type =  percent
##
##                   2.5 %     97.5 %
## (Intercept) -236.538683 537.007482
## t1             4.084872   6.730953
## t2             1.749102   2.349539
```

# Bootstrap confidence intervals

**Method 3:** Bias corrected and accelerated ($BC_a$) CIs

- More complicated, so don't worry about the formula
- Usually performs better than the percentile bootstrap CI

```
confint(transact_boot, level = .95, type = "bca")
```

```
## Bootstrap quantiles, type =  bca
##
##                   2.5 %     97.5 %
## (Intercept) -301.574100 489.152294
## t1             3.739213   6.554452
## t2             1.801175   2.448074
```

# Residual bootstrap

1. Fit the regression model to get $\widehat{\mathbf{y}} = \mathbf{X}\widehat{\boldsymbol{\beta}}$ and compute the residuals, $e_i = y_i - \widehat{y}_i$
2. Obtain a random sample, with replacement, from the residuals to get a new sample ($\mathbf{e}' = (e_1^*, \ldots, e_n^*)$)
3. Create simulated $y$ values by $\mathbf{y}^* = \mathbf{X}\widehat{\boldsymbol{\beta}} + \mathbf{e}^*$
4. Fit the regression model to the simulated $y^*$ values and save the values of the estimated coefficients, or other summary statistics
5. Repeat steps 2-4 R times

We can build confidence intervals from this list of sets of coefficients.

# Residual bootstrap in R

Note: This is only for illustrating how to use R to run the residual bootstrap, as there is evidence of nonconstant variance.

```
transact_boot2 <- Boot(transact_mod, R = 1999, method = "residual")
summary(transact_boot2) # print summary
```

```
##                R original   bootBias     bootSE  bootMed
## (Intercept) 1999 144.3694  3.9037488 169.879761 146.8923
## t1          1999   5.4621  0.0012388   0.446833   5.4566
## t2          1999   2.0345 -0.0016946   0.095262   2.0336
```

```
confint(transact_boot2, level = .95, type = "bca") # calc. CIs
```

```
## Bootstrap quantiles, type =  bca
##
##                   2.5 %     97.5 %
## (Intercept) -180.364724 488.419523
## t1             4.581376   6.369099
## t2             1.848683   2.221492
```

# Case vs. Residual

- The residual bootstrap assumes: (1) linearity, and (2) constant variance.
- If this is true, then the residual bootstrap can be more accurate than the case bootstrap.