Statistical Methods for Insurance: Bootstrap, Permutation and Linear Models

Di Cook & Souhaib Ben Taieb, Econometrics and Business Statistics, Monash University
W6.C1

Overview of this class

- Review of t-tests, confidence intervals and prediction intervals
- Review of bootstrap and permutation
- Application to linear models

Recall the olympics model

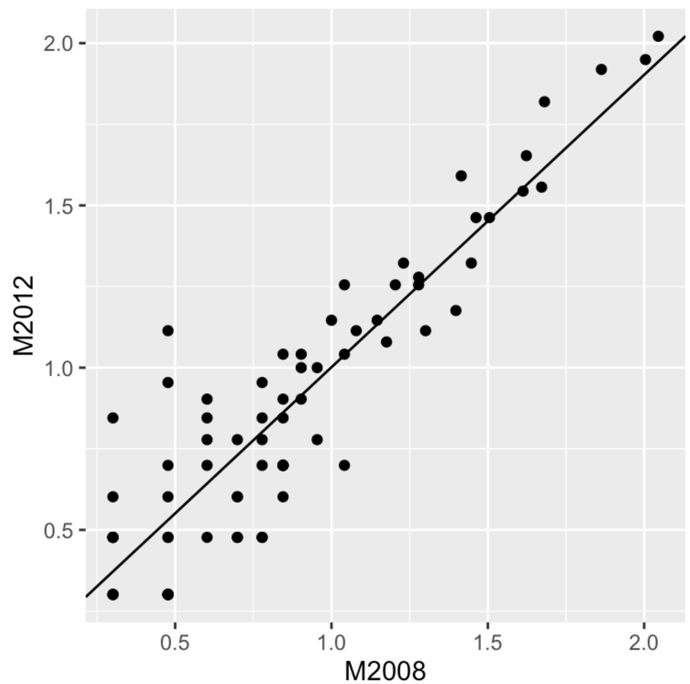
Counts on the log scale

term	estimate	std.error	statistic	p.value
(Intercept)	0.1004	0.0482	2.086	0.0406
M2008	0.9010	0.0491	18.350	0.0000

Model is $log10(M2012+1) =$ 0.1004 + 0.901 $log10(M2008+1) + \varepsilon$.

Your turn

Write down the formula that was used to get the test statistic for the slope parameter.



5/42

Answer

$$\frac{b_1}{SE(b_1)}$$

where

$$SE(b_1) = \frac{MSE}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}}$$

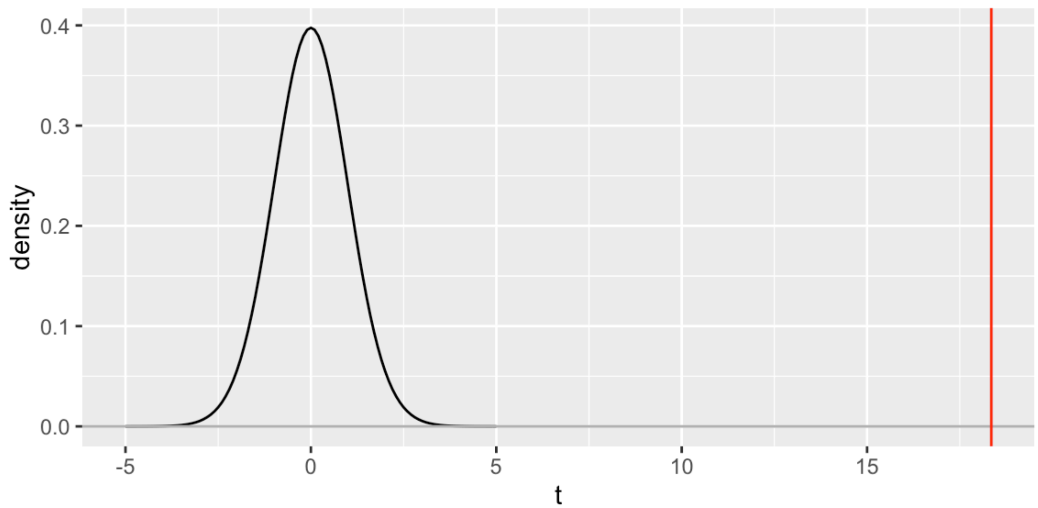
and

$$MSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{(n-2)}}$$

Check the numbers in the table.

t-test

$$H_o : \beta_1 = 0 \;\; vs \;\; H_a : \beta_1 \neq 0$$



Decision: p-value is very small (twice the area to the right of red line), reject H_o

Conclusion: The slope parameter for the regression model using the entire population is not 0.

7/42

Confidence interval for slope

$$b_1 \pm t_{\alpha/2,n-2} SE(b_1)$$

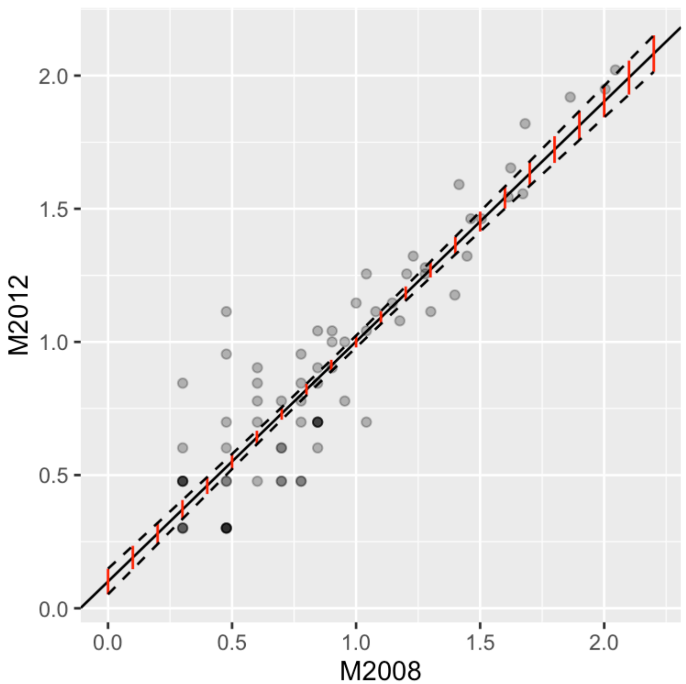
For $\alpha = 0.05$, yielding 95% confidence level, n = 73, $t_{\alpha/2,n-2}$ = 1.9939,

0.901 $\pm$ 1.9939 $\times$ 0.0491 = (0.8031 , 0.9989)

Explanation: We are 95% sure that the slope of a regression model fitted to the entire population is between 0.8 and 1.0.

Confidence interval for predicted value

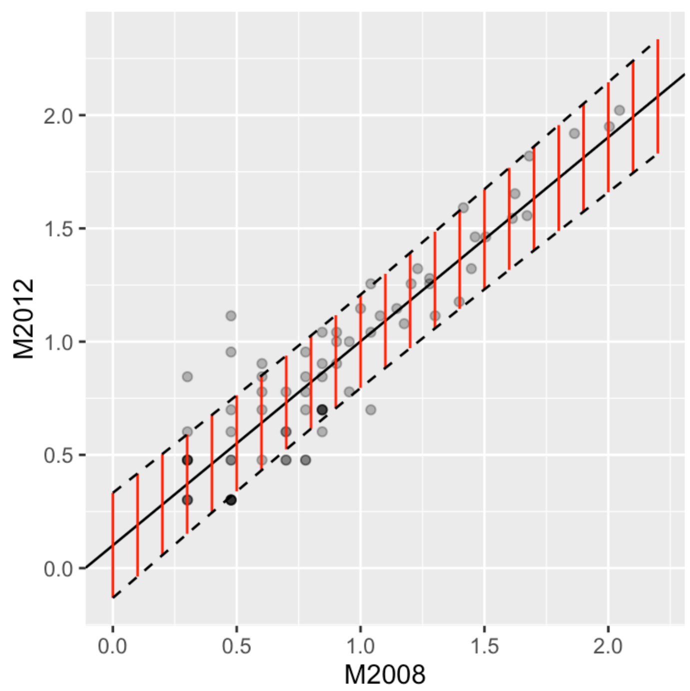
For a given value of x, $\hat{y} \pm t_{\alpha/2,n-2} MSE \sqrt{\frac{1}{n} + \frac{n(x - \bar{X})^2}{n \sum_{i=1}^{n}(X_i - \bar{X})^2}}$



Prediction interval for NEW value

For a given value of x, $\hat{y} \pm t_{\alpha/2,n-2} MSE \sqrt{1 + \frac{1}{n} + \frac{n(x-\bar{X})^2}{n\sum_{i=1}^{n}(X_i-\bar{X})^2}}$

MSE from model fit is 0.1838.



Computational approach

- Hypothesis test can be conducted using permutation
- Confidence and prediction intervals can be generated using bootstrap
- WHY???
- Classical methods have strict assumptions about the distribution of errors. Computational approaches relax these assumptions

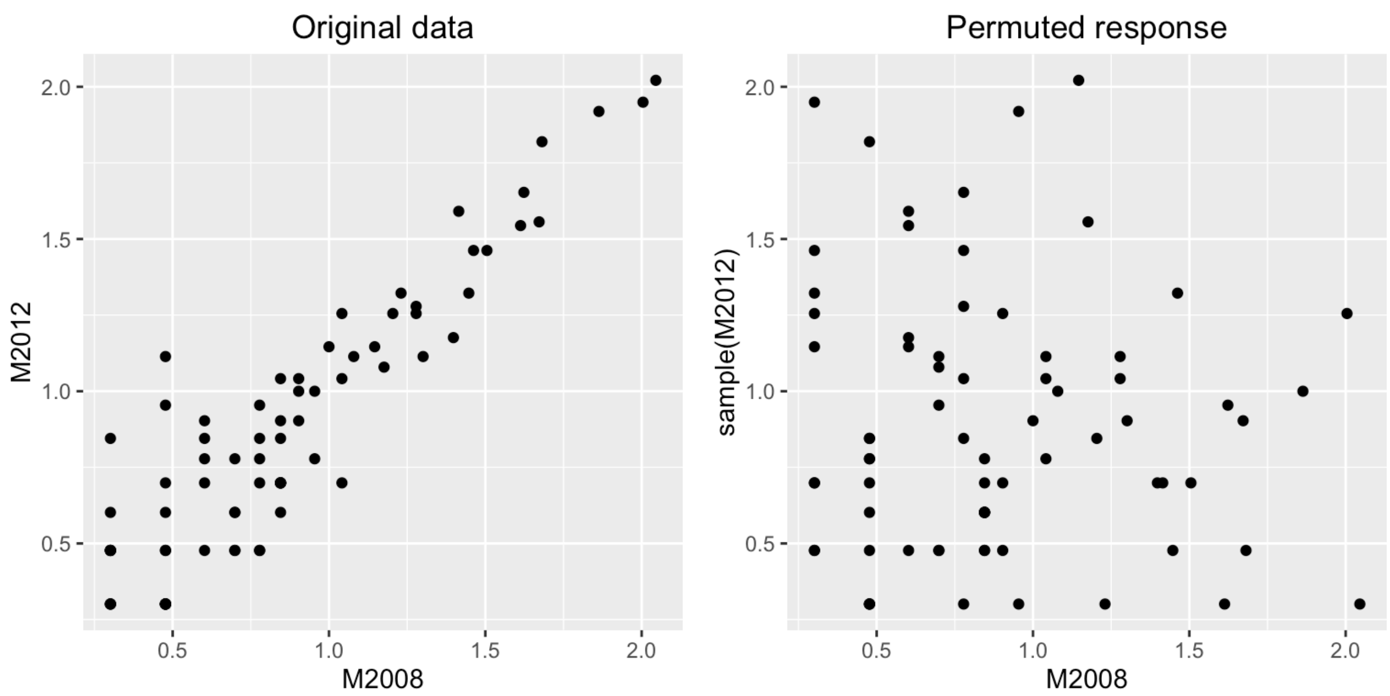
Permutation hypothesis tests

For regression, to test H_o, one column of the two is permuted to break association.

```
df <- data.frame(x=letters[1:5], y=letters[1:5])
head(data.frame(df, py=sample(df$y)))
#>   x y py
#> 1 a a  b
#> 2 b b  d
#> 3 c c  c
#> 4 d d  e
#> 5 e e  a
```

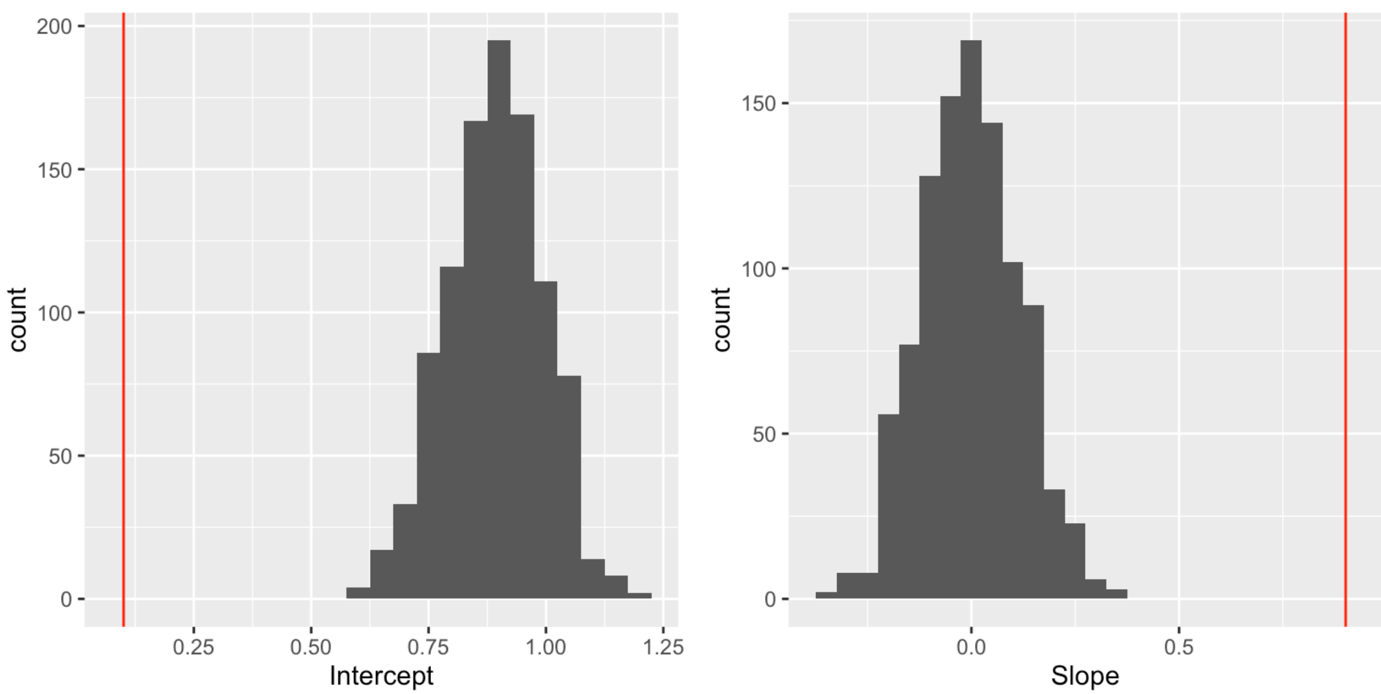
Make many more permutation sets.

```
p1 <- ggplot(oly, aes(x=M2008, y=M2012)) + geom_point() + ggtitle("Original data")
p2 <- ggplot(oly, aes(x=M2008, y=sample(M2012))) + geom_point() + ggtitle("Permuted response")
grid.arrange(p1, p2, ncol=2)
```



13/42

Permutation distribution of intercept and slope



Red lines indicate values from our data, which are far from the values obtained from the permuted data.

Statistical significance

- Permutation gives us samples consistent with $H_o : \beta_1 = 0$, whilst keeping the marginal distributions of X and Y the same.
- In the example we see that the values from the permuted data, center on 0. We are seeing what the variation in b_1 might be, from one sample to another, if the parameter β_1 (slope computed for the whole population) is actually 0.
- To compute the p-value, count the number of values computed on the permuted data that are more extreme than the values from the actual data.
- In this example, the p-value is 0 for both intercept and slope.

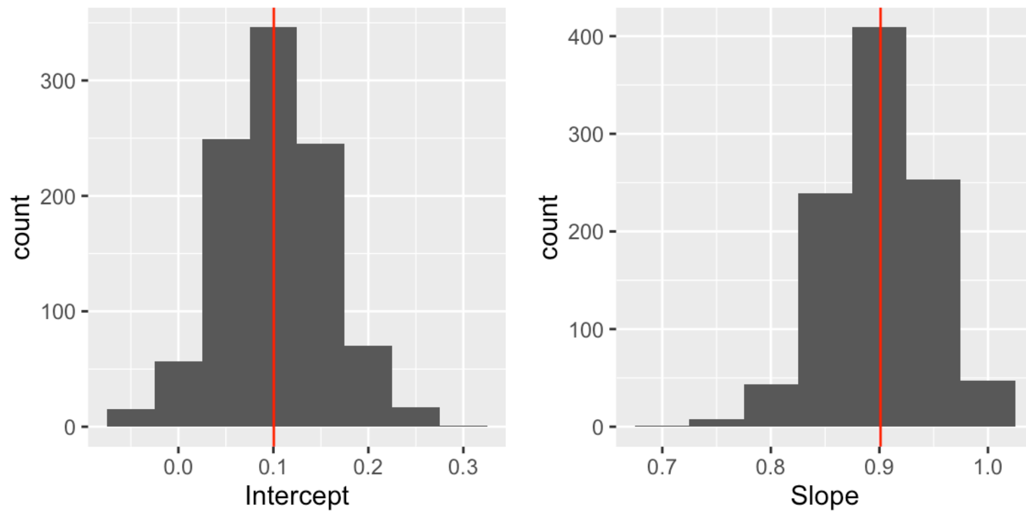
Confidence intervals via bootstrap

1. Make a N boostrap samples (sample data rows, with replacement)
2. Fit the model for each
3. Compute lower and upper C% bounds, by sorting values and pulling the relevant ones, e.g. if N=1000, C=95, we would take the 25^{th} and 975^{th} values as the lower and upper CI bounds

Bootstrap samples

```
orig <- letters[1:10]
orig
#>  [1] "a" "b" "c" "d" "e" "f" "g" "h" "i" "j"
boot1 <- sort(sample(orig, replace=TRUE))
boot1
#>  [1] "a" "b" "f" "f" "g" "h" "h" "i" "i" "i"
```

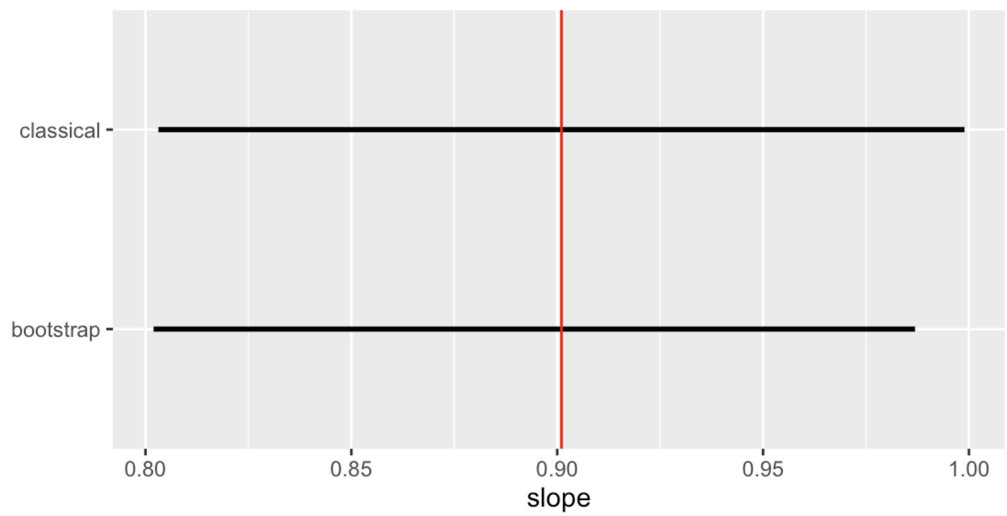
Bootstrap confidence interval for the slope



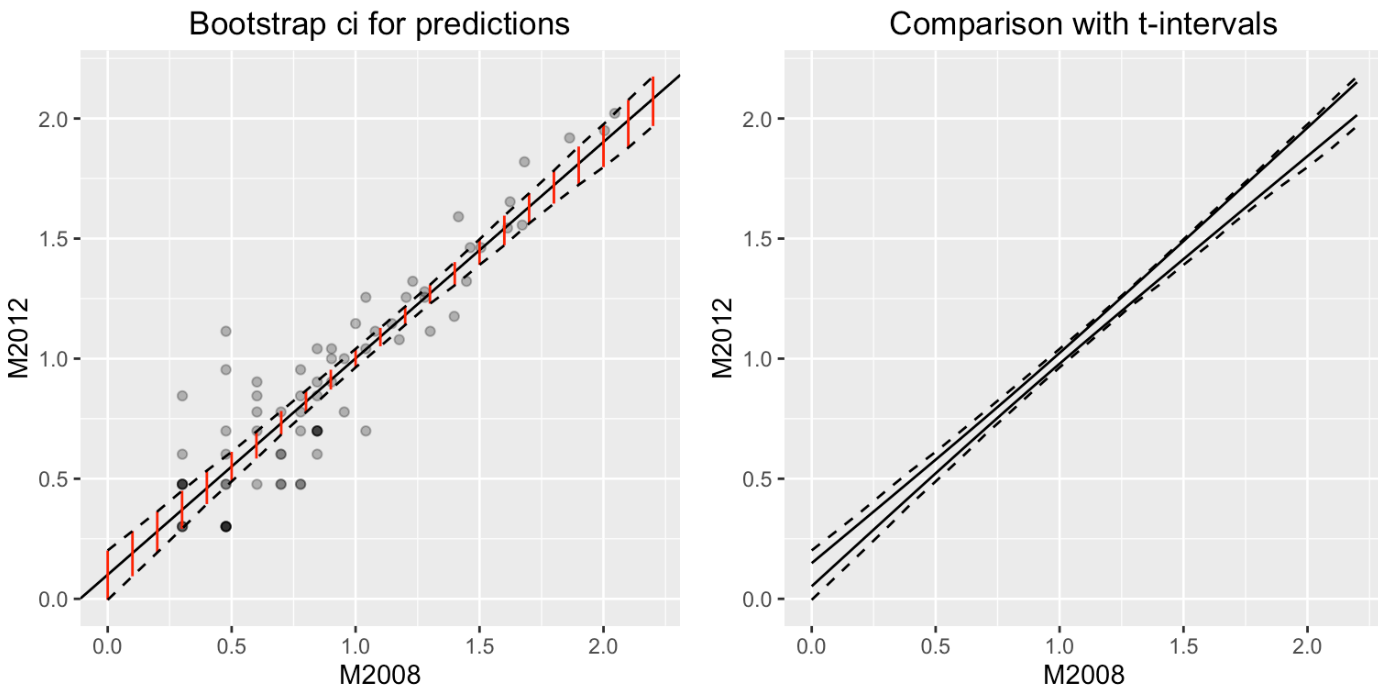
- Intercept: (-0.0085 , 0.2135)
- Slope: (0.8019 , 0.9869)

Compare intervals

```
#>       label      l      u
#> 1 classical 0.8031 0.9989
#> 2 bootstrap 0.8019 0.9869
```



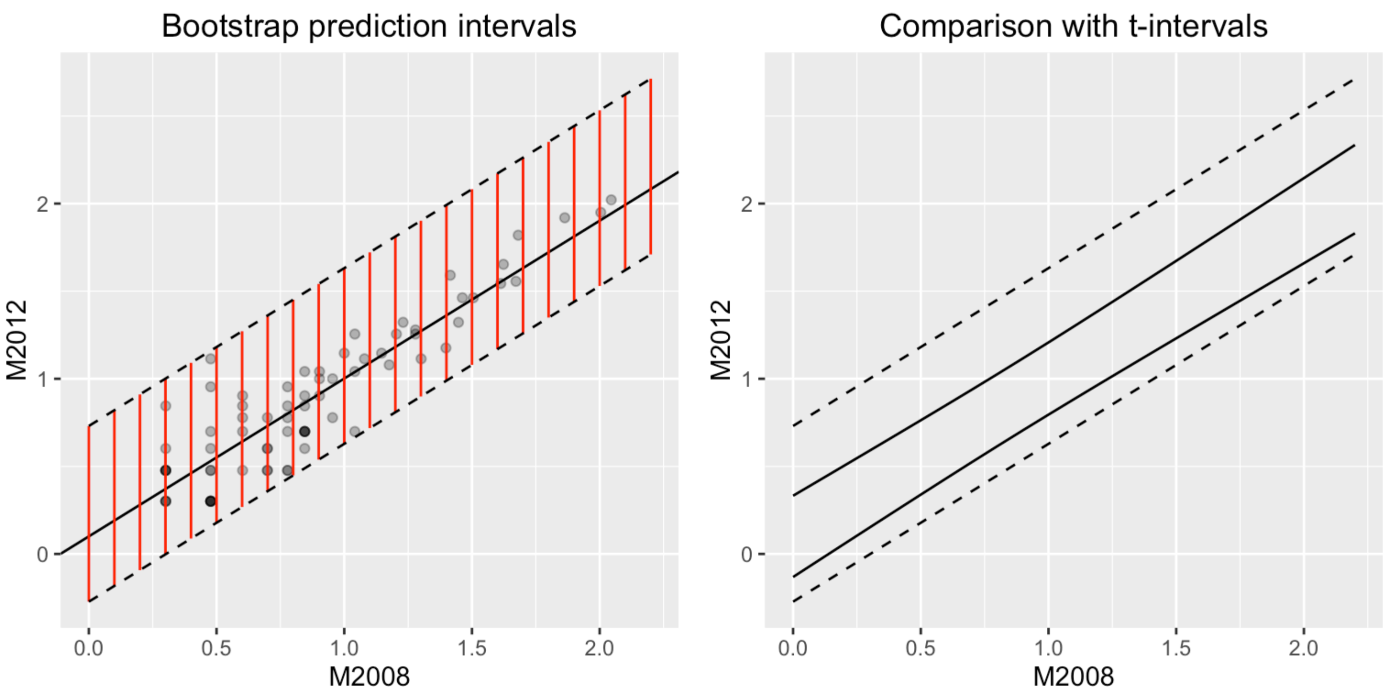
Bootstrap confidence intervals for predicted value



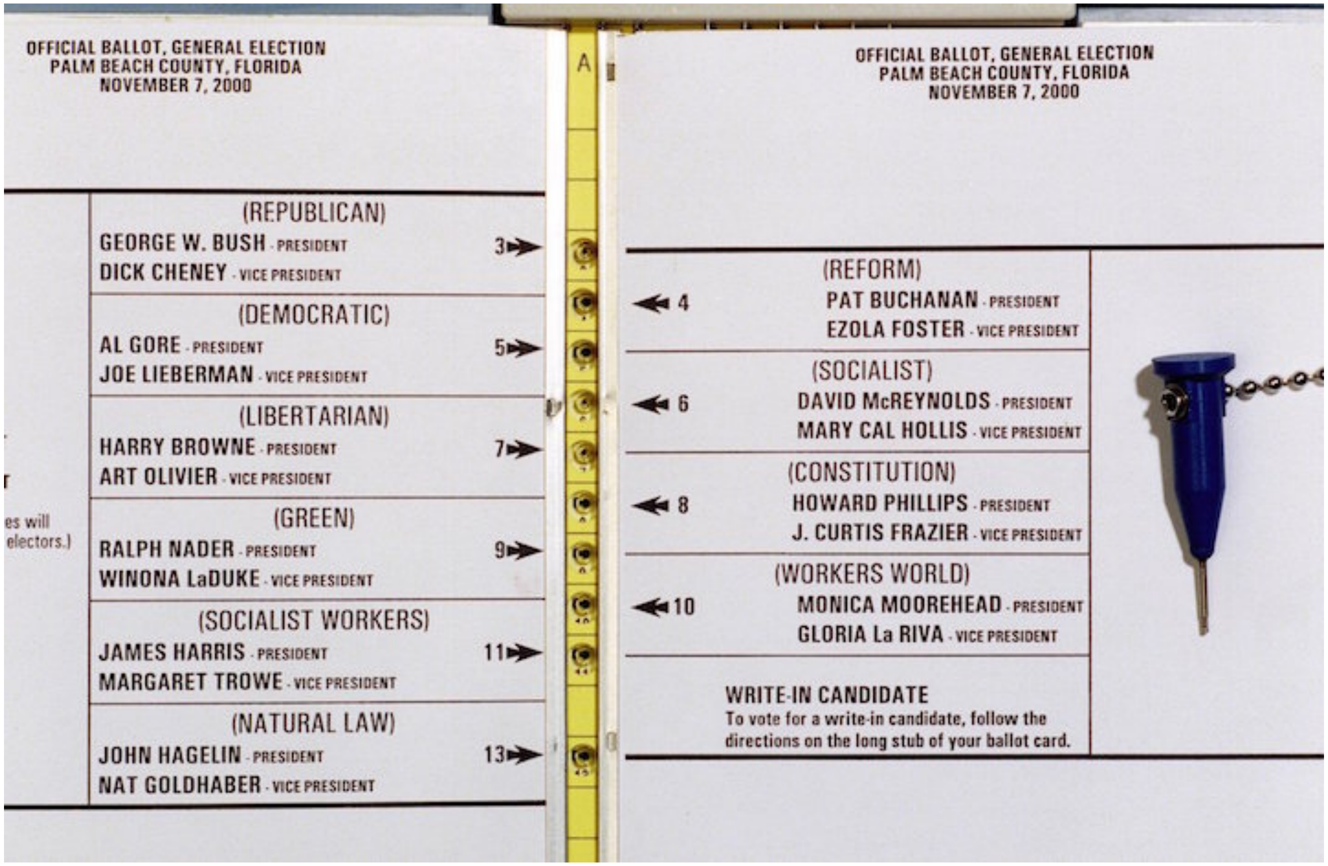
Bootstrap prediction intervals for NEW values

Procedure derives from bootstrapping residuals.

1. Compute the residuals from the fitted model
2. Bootstrap the residuals
3. Find the desired quantiles of the residuals
4. Compute prediction intervals by adding residual quantiles to fitted value

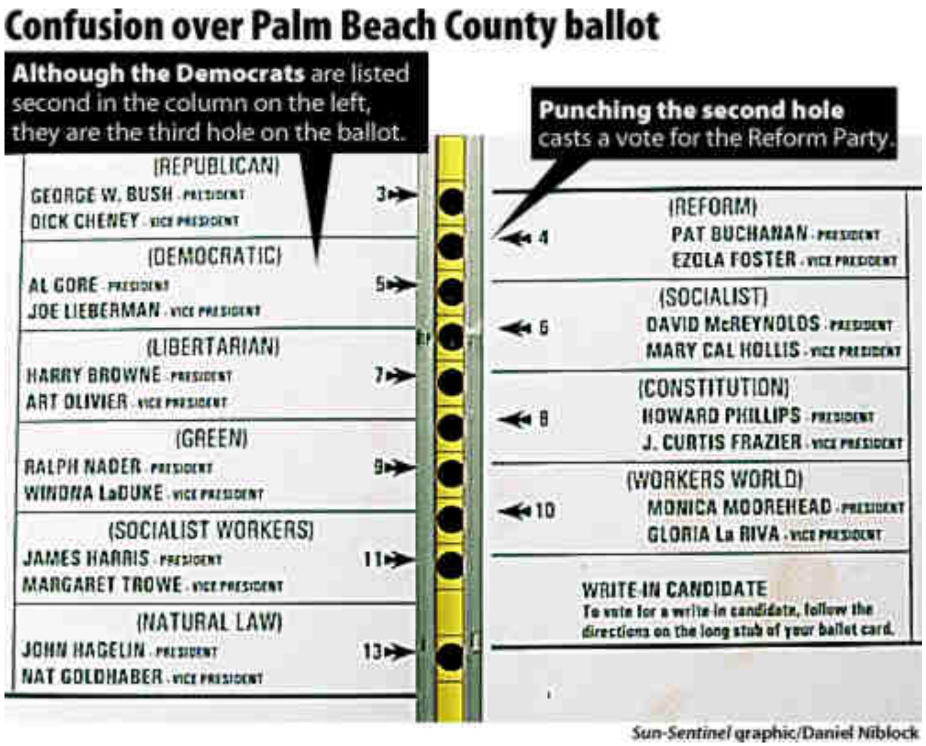

Bootstrap prediction intervals
M2012
M2008
0
1
2
0.0
0.5
1.0
1.5
2.0
Comparison with t-intervals
M2012
M2008
0
1
2
0.0
0.5
1.0
1.5
2.0

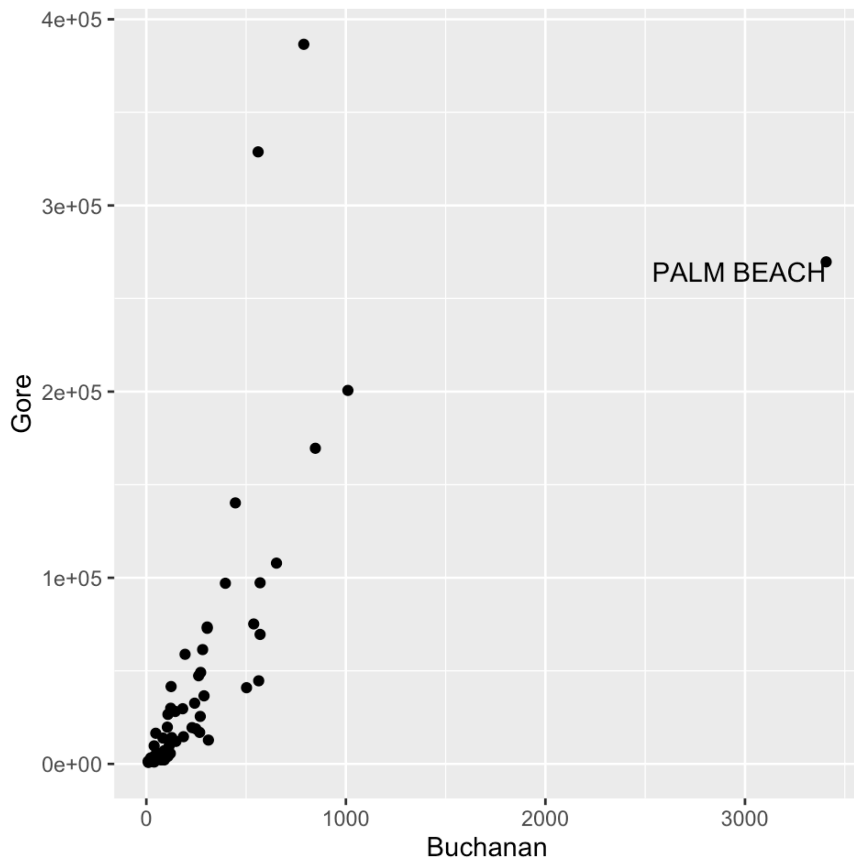
Example: 2000 US Elections



23/42

Example: Confusing?





4e+05
3e+05
2e+05
1e+05
0e+00
Gore
PALM BEACH
0
1000
2000
3000
Buchanan

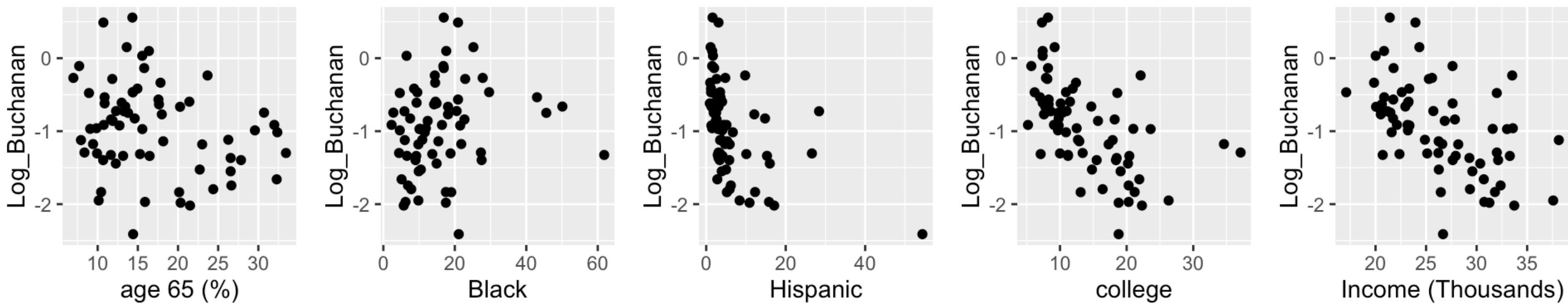
25/42

```
#> Observations: 67
#> Variables: 17
#> $ County             <chr> "ALACHUA", "BAKER", "BAY", "BRADFORD", "BRE...
#> $ Palm_Beach         <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
#> $ Population         <int> 198326, 20761, 146223, 24646, 460977, 14707...
#> $ Log_Population     <dbl> 12.198, 9.941, 11.893, 10.112, 13.041, 14.2...
#> $ Black              <dbl> 21.8, 16.8, 12.4, 22.9, 9.2, 17.5, 16.9, 4....
#> $ Hispanic           <dbl> 4.7, 1.5, 2.4, 2.6, 4.1, 10.9, 1.6, 3.4, 2....
#> $ age 65 (%)         <dbl> 9.428, 7.697, 11.882, 11.819, 16.462, 20.32...
#> $ college            <dbl> 34.6, 5.7, 15.7, 8.1, 20.4, 18.8, 8.2, 13.4...
#> $ Income (Thousands) <dbl> 26.60, 27.61, 26.85, 25.28, 33.28, 31.26, 2...
#> $ Income (Dollars)   <int> 26597, 27614, 26846, 25277, 33284, 31264, 2...
#> $ Age 65 (total)     <int> 18698, 1598, 17374, 2913, 75888, 298900, 17...
#> $ Gore               <int> 47365, 2392, 18850, 3075, 97318, 386561, 21...
#> $ Bush               <int> 34124, 5610, 38637, 5414, 115185, 177323, 2...
#> $ Buchanan           <int> 262, 73, 248, 65, 570, 789, 90, 182, 270, 1...
#> $ Nader              <int> 3215, 53, 828, 84, 4470, 7099, 39, 1462, 13...
#> $ Total Votes        <int> 84966, 8128, 58563, 8638, 217543, 571772, 5...
#> $ Log_Buchanan       <dbl> -1.1765, -0.1074, -0.8593, -0.2844, -1.3393...
```

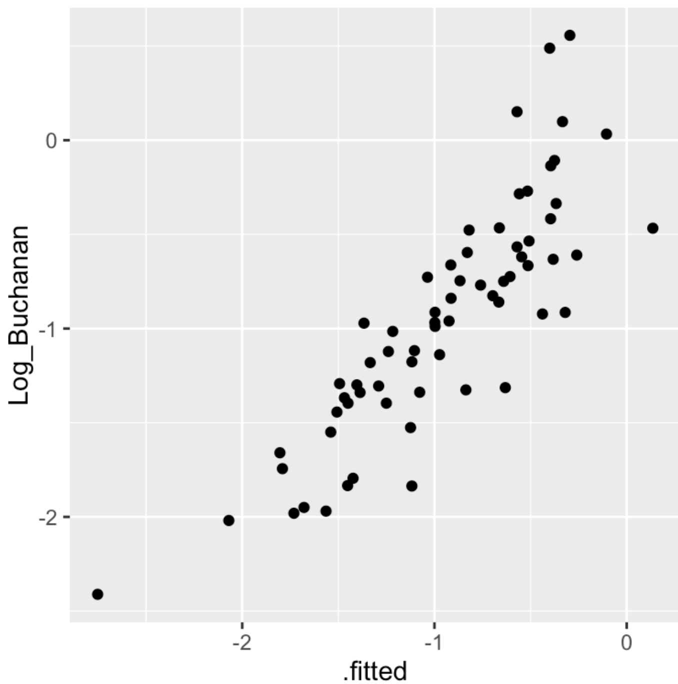
Fit model

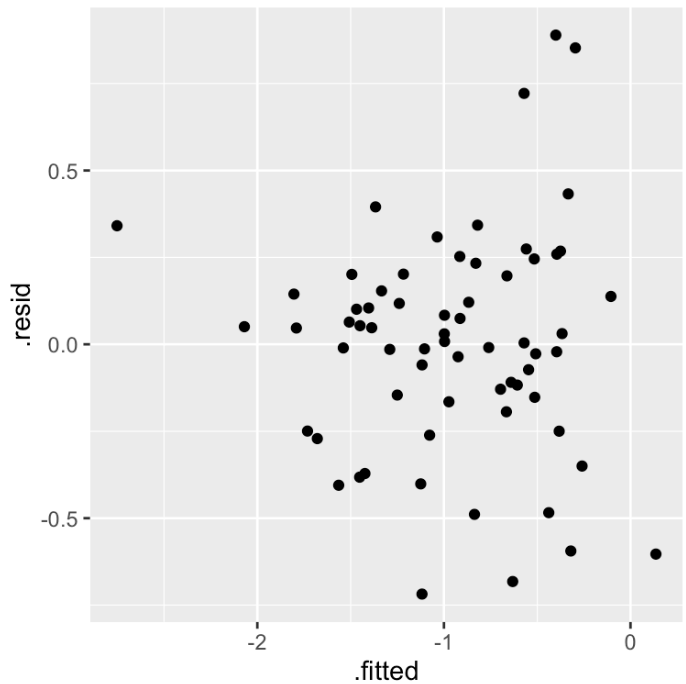
term	estimate	std.error	statistic	p.value
(Intercept)	2.1465	0.3955	5.428	0.0000
`age 65 (%)`	-0.0415	0.0070	-5.939	0.0000
Black	-0.0132	0.0046	-2.884	0.0054
Hispanic	-0.0350	0.0051	-6.807	0.0000
college	-0.0193	0.0097	-1.991	0.0510
`Income (Thousands)`	-0.0658	0.0144	-4.582	0.0000

Predictors


Log_Buchanan
0
-1
-2
10 15 20 25 30
age 65 (%)
0 20 40 60
Black
0 20 40
Hispanic
10 20 30
college
20 25 30 35
Income (Thousands)

Check model



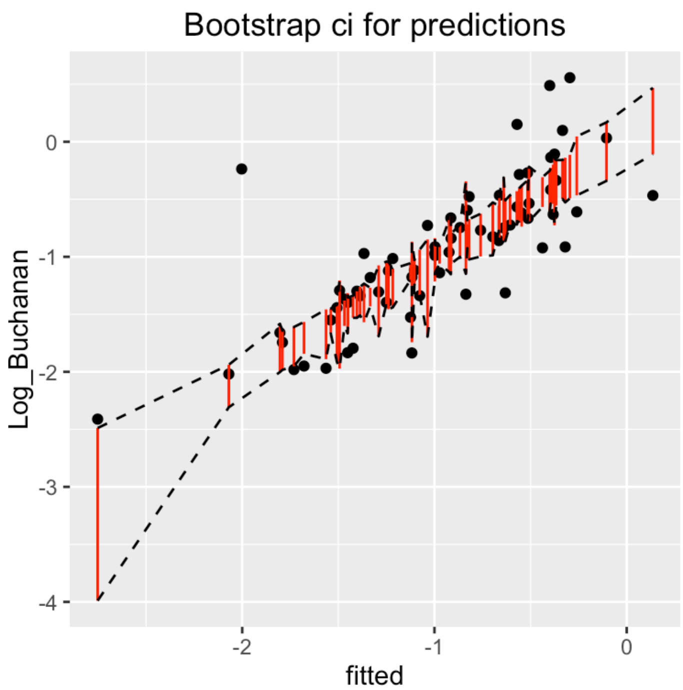


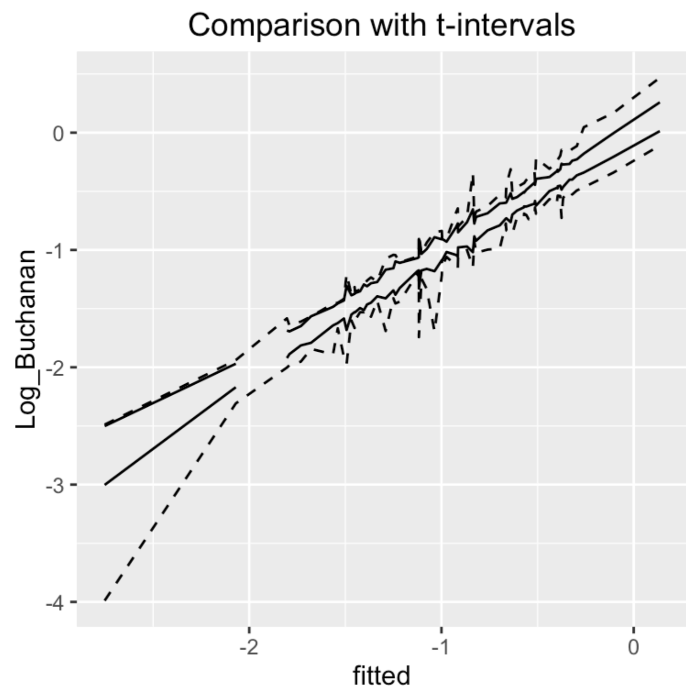
Predict Palm Beach

```
pb <- florida %>% filter(County=="PALM BEACH")
pb_p <- predict(florida_lm, pb)
pb_e <- pb$Log_Buchanan - pb_p
kable(cbind(pb$Log_Buchanan, pb_p, pb_e))
```

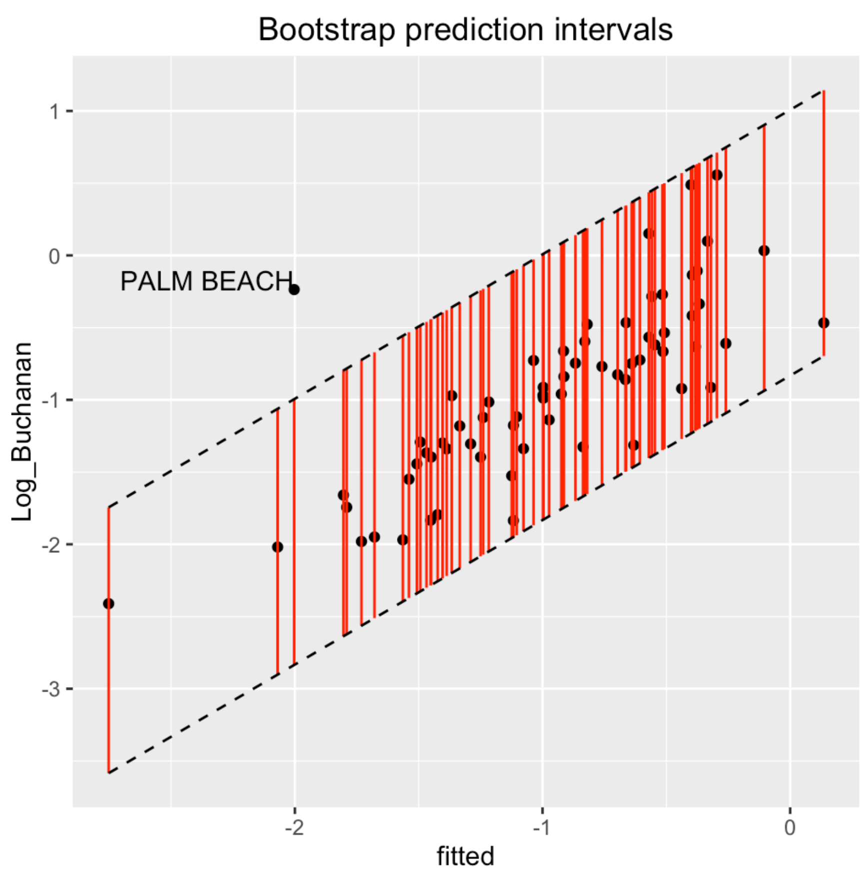
	pb_p	pb_e
-0.2365	-2.003	1.766

Bootstrap confidence for predictions





Bootstrap prediction intervals



Summary

- The number of votes for Buchanan in Palm Beach County were much higher than could be expected given the demographic composition of the locaiton.
- This is evidence that the butterfly ballot may have caused some confusion, and error in voting intention.

Resources

- Statistics online textbook, Diez, Barr, Cetinkaya-Rundel
- Mike Akritas PSU lecture notes
- Nice example for automotive costs
- 2000 US Election Florida undercount

Share and share alike