

STAT135: Confidence intervals and more cars

Nicholas Horton (nhorton@amherst.edu)

March 10, 2017

Chapter 17 in a nutshell

- The distribution of sample means is less variable than the distribution of the underlying population
- The distribution of sample means is more normal than the distribution of the underlying population

Confidence intervals for a proportion

$SE(\hat{p}) =$

Constructing a confidence interval:

Interpreting a confidence interval:

Example 1: Each of the 110 students in a statistics class selects a different random sample of 35 Quiz scores from a population of 5000 scores they are given. Using their data, each student constructs a 90% confidence interval for μ , the average Quiz score of the 5000 students. Which of the following conclusions is correct?

- About 10% of the sample means will not be included in the confidence intervals.
- About 90% of the confidence intervals will contain μ .
- It is probable that 90% of the confidence intervals will be identical.
- About 10% of the raw scores in the samples will not be found in these confidence intervals.

Example 2: Suppose two researchers want to estimate the proportion of American college students who favor abolishing the penny. They both want to have about the same margin of error to estimate this proportion. However, Researcher 1 wants to estimate with 99% confidence and Researcher 2 wants to estimate with 95% confidence. Which researcher would need more students for her study in order to obtain the desired margin of error?

- Researcher 1.
- Researcher 2.
- Both researchers would need the same number of subjects.
- It is impossible to obtain the same margin of error with the two different confidence levels.

More cars

```
ds <- read_csv("carscollated2017.csv")
ds <- ds %>%
  mutate(price = readr::parse_number(price))
```

```
tally(~ year, margins=TRUE, data=ds)
```

```
## year
## 2007 2010 2011 2012 2013 2014 2015 2016 2017 Total
##      2     5    37    45   176   237   201   126     2   831
```

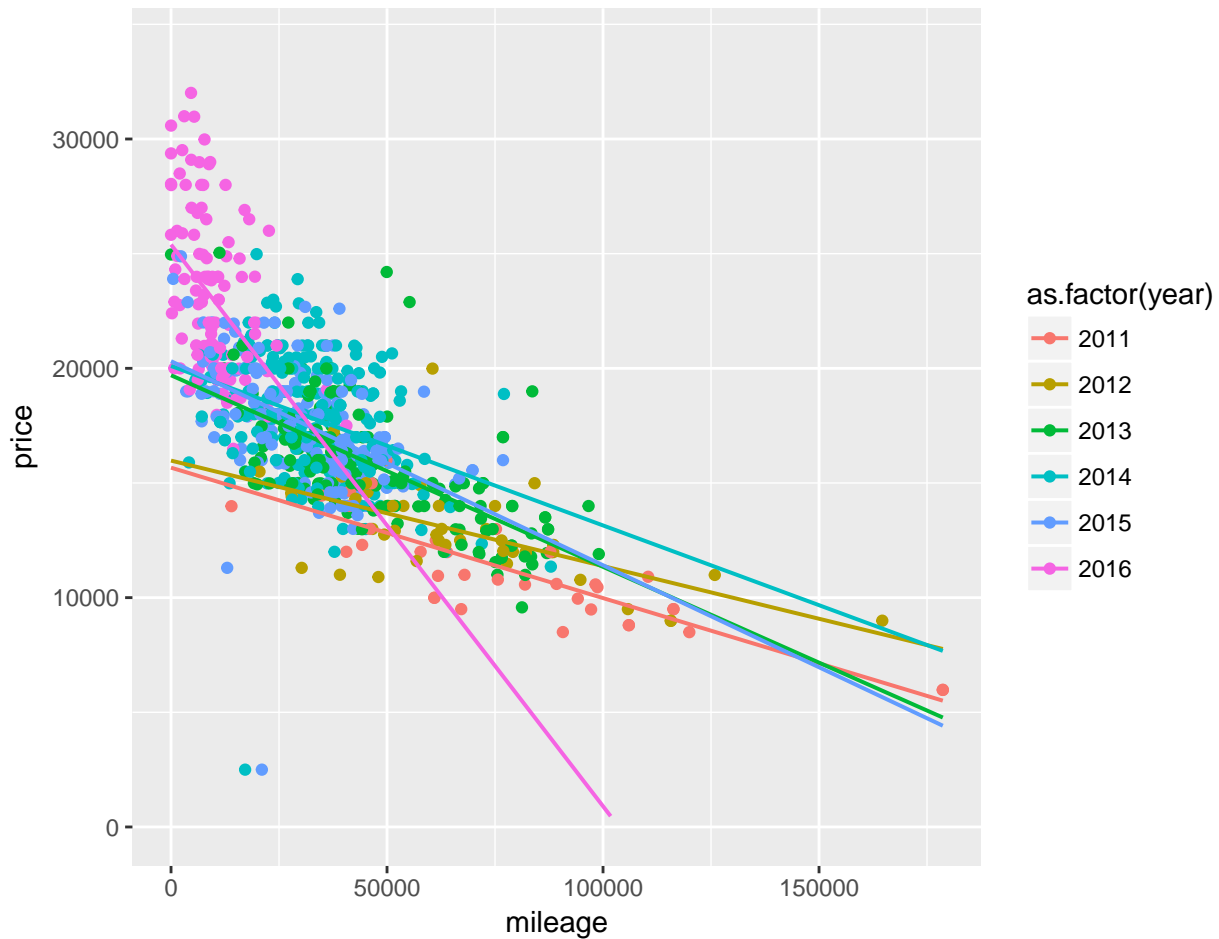
```
tally(~ location, margins=TRUE, data=ds)
```

```
## location
##      40202      Atlanta  Bangor, ME  Baton Rouge  Boston
##      40      40      40      40      40
##      Buffalo  Chicago  Cleveland  Dallas  Los Angeles
##      40      41      26      41      40
##      Minneapolis  New Orleans  NYC  Phoenix  Portland
##      59      33      40      40      40
##      Richmond Salt Lake City  San Diego  San Francisco  Seattle
##      40      33      40      39      39
##      Tampa      Total
##      40      831
```

```
ds <- filter(ds, year > 2010, year < 2017) # drop new cars and really old cars
```

```
gf_point(price ~ mileage, color = ~ as.factor(year), data = ds) %>%
  gf_lm() %>%
  gf_lims(y = c(0, 34000))
```

```
## Warning: Removed 34 rows containing missing values (geom_lm).
```



a) interpret what insights you can make from the scatterplot

SOLUTION:

```
options(scipen=5, show.signif.stars=FALSE, digits=4)
mod <- lm(price ~ location + mileage + as.factor(year) + mileage*as.factor(year), data=ds)
msummary(mod)
```

##	Estimate	Std. Error	t value	Pr(> t)
## (Intercept)	17061.06938	868.85620	19.64	< 2e-16
## locationAtlanta	-1638.41488	462.55755	-3.54	0.00042
## locationBangor, ME	-1689.69743	463.90469	-3.64	0.00029
## locationBaton Rouge	-745.21252	474.32078	-1.57	0.11656
## locationBoston	-563.64808	460.06933	-1.23	0.22089
## locationBuffalo	-581.60744	484.23517	-1.20	0.23008
## locationChicago	-2237.49897	456.49750	-4.90	1.2e-06

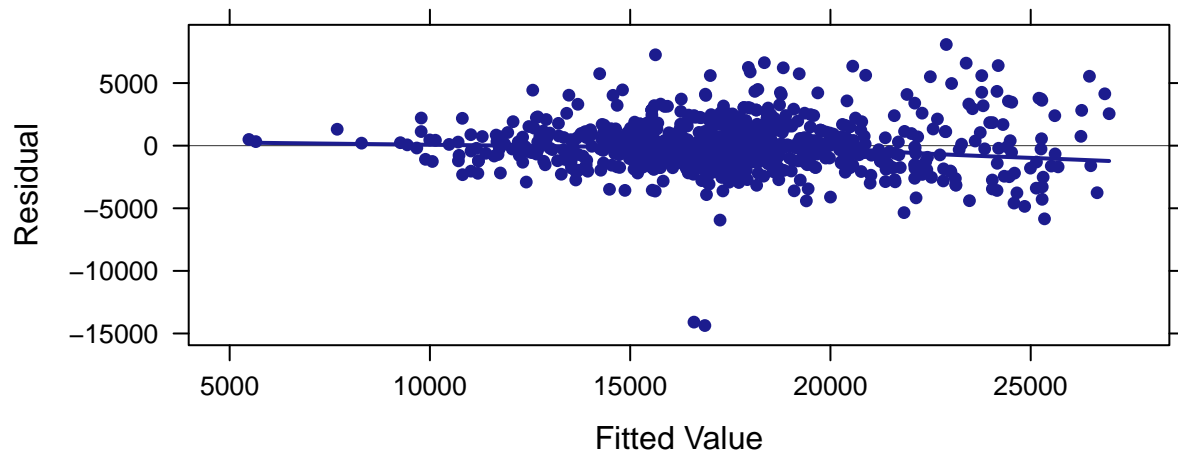
```
## locationCleveland      -1491.58656   520.87678   -2.86  0.00430
## locationDallas        -1078.11128   462.04754   -2.33  0.01988
## locationLos Angeles    2319.67933   460.04752    5.04  5.7e-07
## locationMinneapolis   -622.89582   423.72233   -1.47  0.14194
## locationNew Orleans    -573.29737   498.84387   -1.15  0.25080
## locationNYC            -594.56186   458.89342   -1.30  0.19548
## locationPhoenix        -325.96320   463.81240   -0.70  0.48239
## locationPortland        65.24543    461.66827    0.14  0.88765
## locationRichmond       -744.32172   461.18604   -1.61  0.10694
## locationSalt Lake City -1954.04693   494.67997   -3.95  8.5e-05
## locationSan Diego       257.69790   461.97731    0.56  0.57713
## locationSan Francisco  1578.28190   461.39289    3.42  0.00066
## locationSeattle        2136.54194   463.06079    4.61  4.6e-06
## locationTampa          -2152.29736   462.16712   -4.66  3.8e-06
## mileage                -0.06065     0.00950   -6.38  3.0e-10
## as.factor(year)2012     -251.31079  1135.10846   -0.22  0.82484
## as.factor(year)2013      3237.23166   894.68539    3.62  0.00032
## as.factor(year)2014      3140.19070   888.34344    3.53  0.00043
## as.factor(year)2015      3252.51391   885.30630    3.67  0.00026
## as.factor(year)2016      8208.61054   874.47684    9.39  < 2e-16
## mileage:as.factor(year)2012  0.01709     0.01436    1.19  0.23445
## mileage:as.factor(year)2013  -0.01797     0.01215   -1.48  0.13939
## mileage:as.factor(year)2014  -0.00343     0.01396   -0.25  0.80603
## mileage:as.factor(year)2015  -0.00989     0.01393   -0.71  0.47777
## mileage:as.factor(year)2016  -0.18186     0.02754   -6.60  7.3e-11
##
## Residual standard error: 2040 on 790 degrees of freedom
## Multiple R-squared:  0.736, Adjusted R-squared:  0.726
## F-statistic: 71.2 on 31 and 790 DF, p-value: <2e-16
```

b) interpret the regression results

SOLUTION:

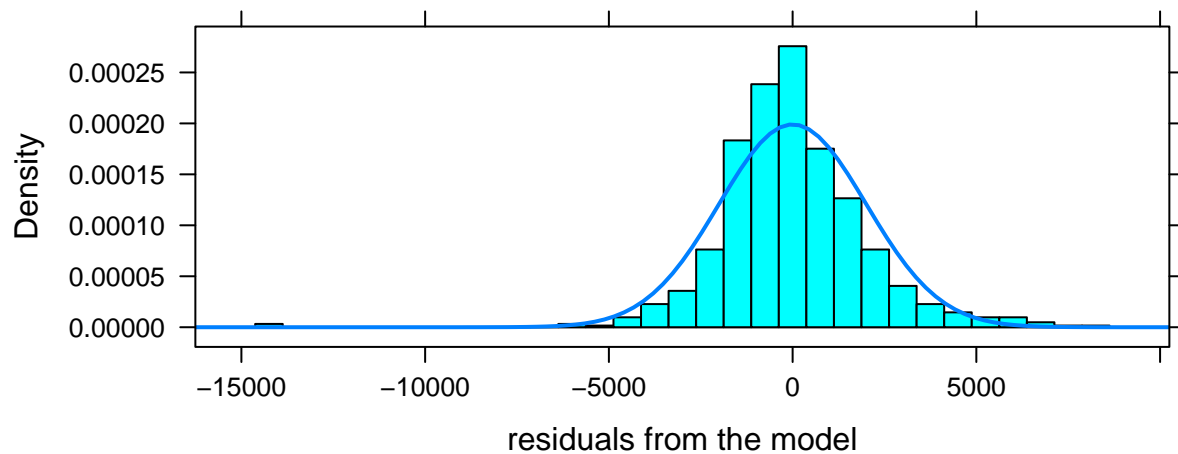
```
mplot(mod, which=1) # Figure 1
```

Residuals vs Fitted



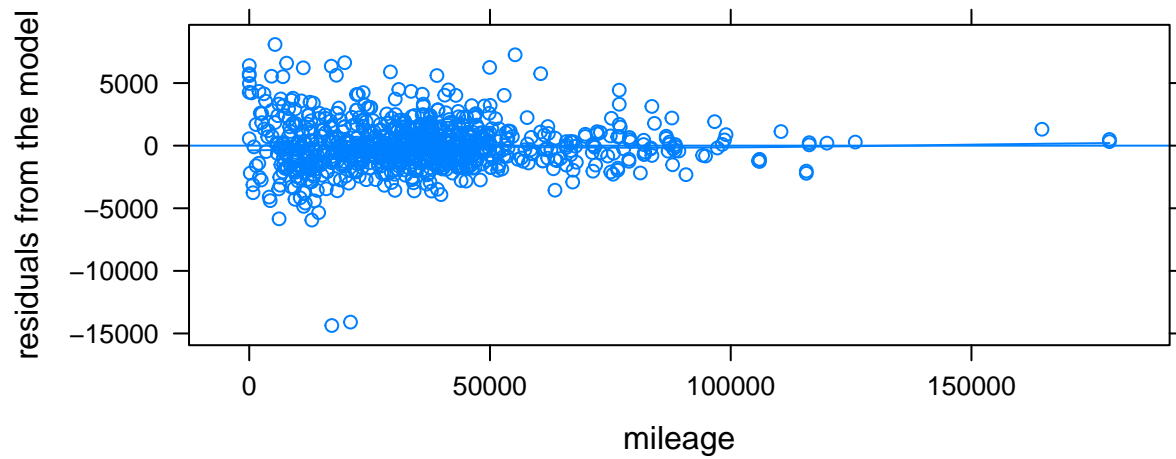
```
histogram(~ resid(mod), fit="normal", width=750, main="Figure 2", xlab="residuals from the model")
```

Figure 2



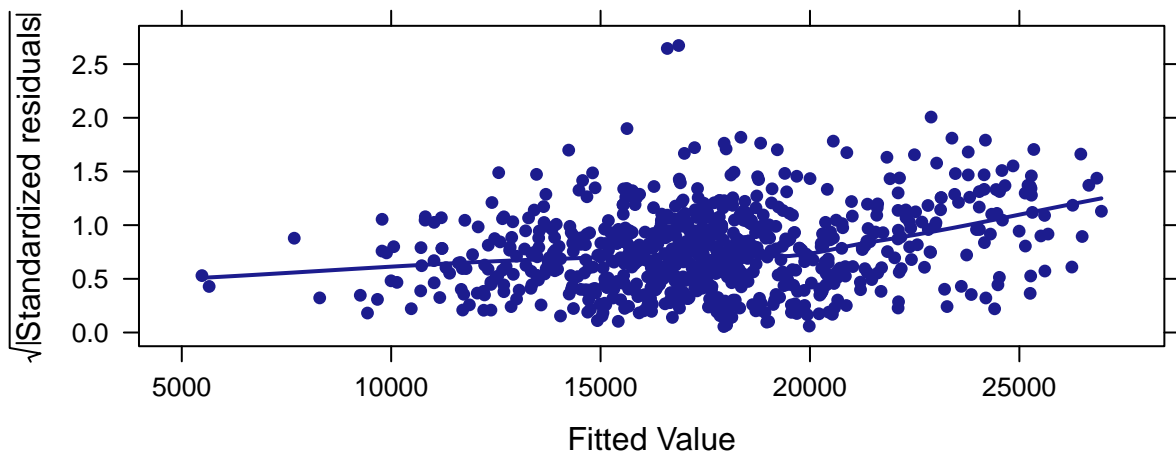
```
xyplot(resid(mod) ~ mileage, type=c("p", "r", "smooth"), ylab="residuals from the model",  
main="Figure 3", data=ds)
```

Figure 3



```
mplot(mod, which=3) # Figure 4
```

Scale-Location



c) interpret the regression diagnostics

(be sure to specify which assumption is being verified using which figure)

SOLUTION:

```
carfun <- makeFun(mod)
carfun(year=2016, mileage=1000, location='Tampa')
```

```
##      1
## 22875
```

```
carfun(year=2016, mileage=1000, location='Tampa') -
carfun(year=2016, mileage=1001, location='Tampa')
```

```
##      1
## 0.2425
```

```
carfun(year=2012, mileage=1000, location='Tampa') -
carfun(year=2012, mileage=1001, location='Tampa')
```

```
##      1
## 0.04355
```

```
ds <- mutate(ds, fitted = predict(mod), resid = resid(mod))
filter(ds, resid(mod) < -10000)
```

```
## # A tibble: 2 x 8
##       car model price  year mileage location fitted  resid
##       <chr> <chr> <dbl> <int>   <dbl>   <chr>   <dbl>   <dbl>
## 1 Toyota Prius  four  2500  2014   17152 Chicago  16865 -14365
## 2 Toyota Prius  four  2500  2015   21027 Chicago  16593 -14093
```

d) what might we conclude about the large residuals?

SOLUTION: