(You should start with the ".Rmd" file that produced what you're reading: hw4.Rmd (hw4.Rmd). But this HTML version is easier to read.)

# STAT 327 Homework 4

We'll grade your homework by

- opening your "hw4.Rmd" file in RStudio in a directory (folder) containing the data file(s)
- clicking "Knit HTML"
- reading the HTML output
- reading your "hw4.Rmd"

You should write R code anywhere you see an empty R code chunk. You should write English text (surrounded by doubled asterisks, `**...**` , to get **boldface type**) anywhere you see "...".

The HTML version of these instructions is easier to read than this plain text ".Rmd" file because the math notation is rendered nicely. So start by clicking "Knit HTML". (Then read this ".Rmd" file, and you'll see that it's not hard to make nice mathematical notation in R Markdown.)

Include reasonable titles and labels with each of your graphs.

Name: NAIQING CAI

Email: ncai5@wisc.edu (mailto:ncai5@wisc.edu)

# Part 1: Statistical tests and confidence intervals

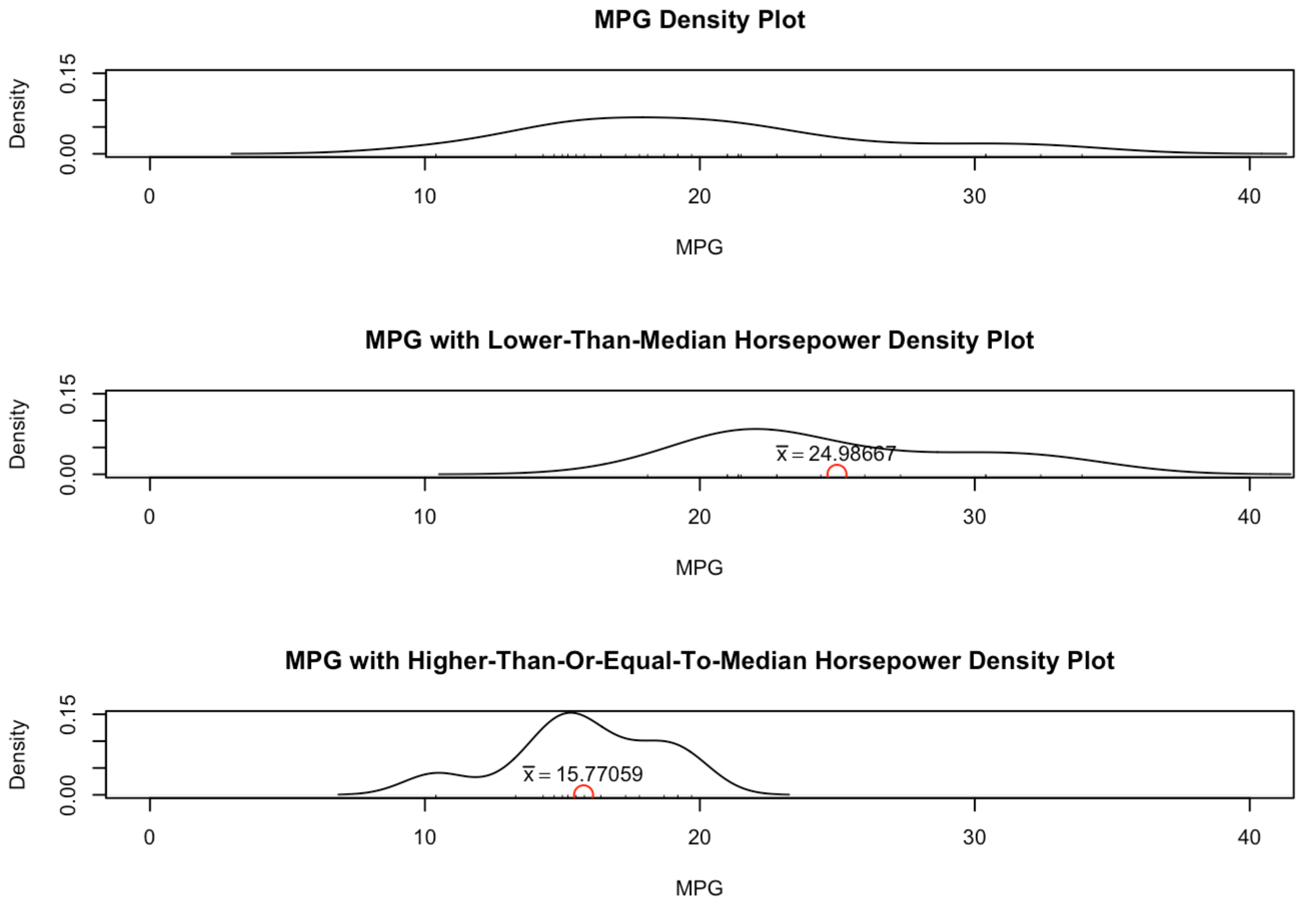
## Difference of two means

Regarding the "mtcars" data frame, let's investigate whether engine horsepower influences gas mileage. (For the sake of this exercise, suppose that the assumptions of the difference-of-two-means test are met. In fact, they probably are not met.)

## Make one graph consisting of three rows and one column, using the same $x$-limits, c(0, 40), and the same $y$-limits, c(0, 0.15), each time.

- On top, make a density plot with rug of mpg.
- In the middle,
  - make a density plot with rug of mpg for those cars with lower-than-median horsepower
  - add a solid red circle, twice as large as the default size, at the location of the mean of these data
  - add a label "$\bar{x} = 0$" (replacing "0" with the correct value) just above the red circle (hint: see ?plotmath for the bar on $x$, ?text to create a graph label, and this hint on labels (label.html))
- At the bottom,
  - make a density plot with rug of mpg for those cars with higher-than-or-equal-to-median horsepower
  - add a solid red circle, twice as large as the default size, at the location of the mean of these data
  - add a label "$\bar{x} = 0$" (replacing "0" with the correct value) (hint on labels (label.html))

```
m = matrix(data = c(1,2,3), nrow = 3, ncol = 1, byrow = TRUE)
layout(m)
# top
plot(density(mtcars$mpg), xlim = c(0,40), ylim = c(0,0.15), main = "MPG Density Plot"
, xlab = "MPG")
rug(mtcars$mpg)
# middle
plot(density(mtcars$mpg[mtcars$hp < median(mtcars$hp)]), main = "MPG with Lower-Than-
Median Horsepower Density Plot", xlab = "MPG", xlim = c(0,40), ylim = c(0,0.15))
rug(mtcars$mpg[mtcars$hp < median(mtcars$hp)])
x = mtcars$mpg[mtcars$hp < median(mtcars$hp)]
points(mean(x), 0, col = "red", cex = 2)
xbar = mean(x)
text(mean(x), .04, labels = bquote(bar(x) == .(xbar)))
# bottom
plot(density(mtcars$mpg[mtcars$hp >= median(mtcars$hp)]), xlim = c(0,40), ylim = c(0,
0.15), main = "MPG with Higher-Than-Or-Equal-To-Median Horsepower Density Plot", xlab
 = "MPG")
rug(mtcars$mpg[mtcars$hp >= median(mtcars$hp)])
y = mtcars$mpg[mtcars$hp >= median(mtcars$hp)]
points(mean(y), 0, col = "red", cex = 2)
text(mean(y), .04, labels = bquote(bar(x) ==  .(mean(y))))
```



# Judging only from the graph of the two samples, describe at least two differences in the corresponding populations.

**The mean MPG for the Lower-than-Median graph is significantly higher than the mean of that of the Higher-than-Or-Equal-To-Median graph, 24.98 vs.15.77 respectively.**

**The Lower-Than-Median plot is much more spread out, as values range from 10-40, compared to the Higher-Than-Or-Equal-To-Median range from roughly 7-22.**

## Find the P-value for testing $H_0 : \mu_{\text{mpg for low-power cars}} - \mu_{\text{mpg for high-power cars}} = 0$ against the alternative $H_1 : \mu_{\text{mpg for low-power cars}} - \mu_{\text{mpg for high-power cars}} > 0$.

```
result = t.test(x,y, alternative = "greater")
result$p.value
```

```
## [1] 8.082542e-07
```

What conclusion do you draw?

**The p value is much less than 0.05, which means I should reject the null assumption and support the alternative that the MPG for low-power cars is greater than that of high-power cars.**

# Part 2: Regression models for the price of beef

## Read the data

- Read beef.txt (beef.txt) into a data frame. (Hint: it requires one line–see ?read.table. You may read it from a file saved in the same directory as your script, or you may read it directly from the class website.)
- Display the data frame’s structure
- Display the data frame’s summary

```
t = read.table("beef.txt", header = T, comment.char = "%")
str(t)
```

```
## 'data.frame':    17 obs. of  10 variables:
##  $ YEAR : int  1925 1926 1927 1928 1929 1930 1931 1932 1933 1934 ...
##  $ PBE  : num  59.7 59.7 63 71 71 74.2 72.1 79 73.1 70.2 ...
##  $ CBE  : num  58.6 59.4 53.7 48.1 49 48.2 47.9 46 50.8 55.2 ...
##  $ PPO  : num  60.5 63.3 59.9 56.3 55 59.6 57 49.5 47.3 56.6 ...
##  $ CPO  : num  65.8 63.3 66.8 69.9 68.7 66.1 67.4 69.7 68.7 62.2 ...
##  $ PFO  : num  65.8 68 65.5 64.8 65.6 62.4 51.4 42.8 41.6 46.4 ...
##  $ DINC : num  51.4 52.6 52.1 52.7 55.1 48.8 41.5 31.4 29.4 33.2 ...
##  $ CFO  : num  90.9 92.1 90.9 90.9 91.1 90.7 90 87.8 88 89.1 ...
##  $ RDINC: num  68.5 69.6 70.2 71.9 75.2 68.3 64 53.9 53.2 58 ...
##  $ RFP  : int  877 899 883 884 895 874 791 733 752 811 ...
```

```
summary(t)
```

```
##       YEAR           PBE             CBE             PPO
##  Min.   :1925   Min.   :56.00   Min.   :46.00   Min.   :41.50
##  1st Qu.:1929   1st Qu.:63.40   1st Qu.:49.00   1st Qu.:51.00
##  Median :1933   Median :70.20   Median :53.70   Median :57.00
##  Mean   :1933   Mean   :69.06   Mean   :53.09   Mean   :56.58
##  3rd Qu.:1937   3rd Qu.:73.00   3rd Qu.:55.20   3rd Qu.:60.50
##  Max.   :1941   Max.   :82.20   Max.   :60.00   Max.   :73.90
##       CPO             PFO             DINC             CFO
##  Min.   :47.70   Min.   :41.60   Min.   :29.40   Min.   :87.30
##  1st Qu.:62.20   1st Qu.:47.80   1st Qu.:40.80   1st Qu.:90.00
##  Median :66.10   Median :51.40   Median :44.50   Median :90.70
##  Mean   :63.93   Mean   :54.22   Mean   :44.62   Mean   :91.01
##  3rd Qu.:68.70   3rd Qu.:64.80   3rd Qu.:52.10   3rd Qu.:91.10
##  Max.   :72.40   Max.   :68.00   Max.   :56.30   Max.   :97.50
##      RDINC            RFP
##  Min.   :53.20   Min.   :733.0
##  1st Qu.:64.00   1st Qu.:798.0
##  Median :69.60   Median :845.0
##  Mean   :68.65   Mean   :833.2
##  3rd Qu.:72.50   3rd Qu.:877.0
##  Max.   :89.50   Max.   :899.0
```

# A first step after reading a data set into an R data frame is to check whether the categorical variables are encoded as factors. Does the beef data set have any categorical variables that should be encoded as factors?

**No, there is no any categorical variables that should be encoded as factors.**

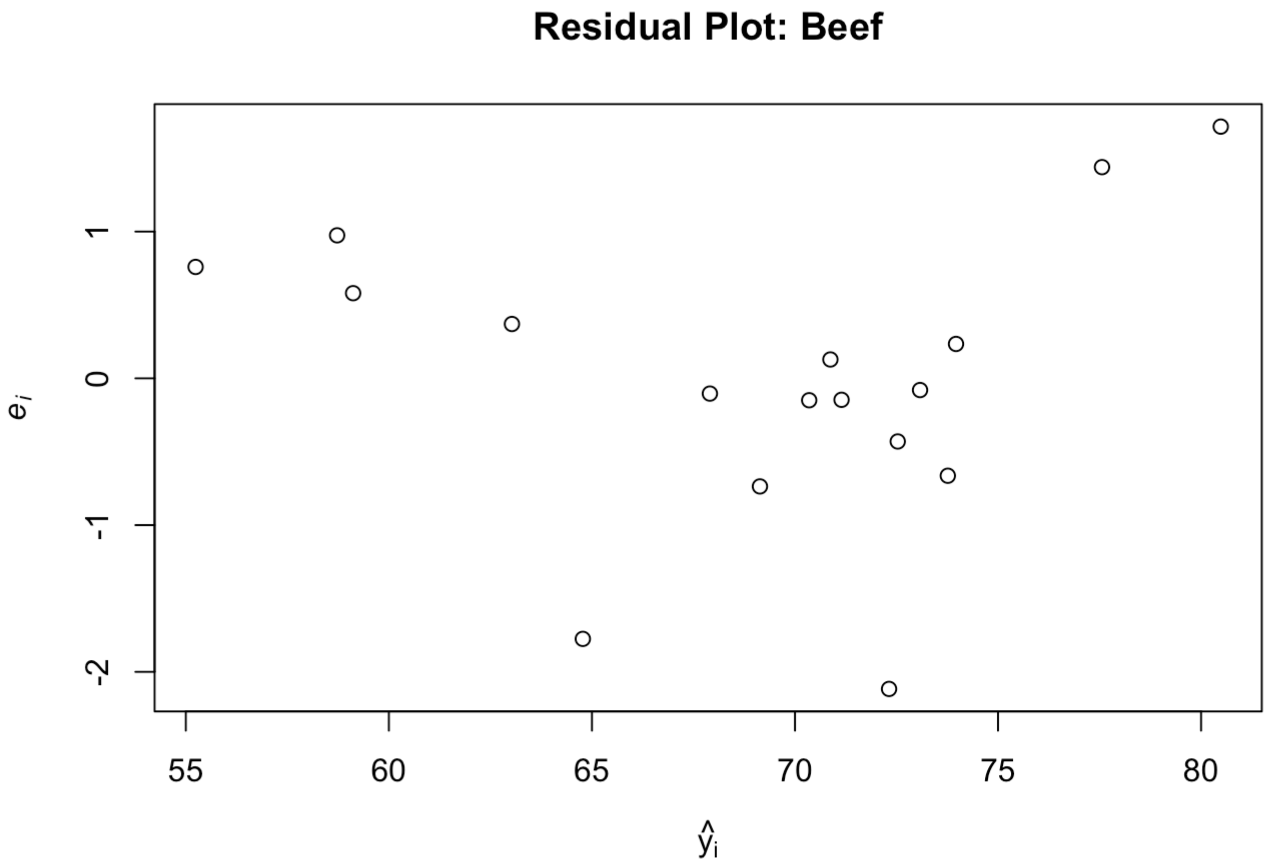
## Multiple regression

- Make a multiple linear regression model for PBE (price of beef) depending on the other variables (including YEAR).
- Display the model summary.
- Make a residual plot of residuals vs. fitted values (both of which are in your model–you don't need to do calculations):
    - Plot points of the form $(\hat{y}_i, e_i)$
    - Include the title, "Beef: residual plot"
    - Include the $y$-axis label $e_i$ (hint: search ?plotmath for how to get the subscripted $i$)
    - Include the $x$-axis label $\hat{y}_i$ (search ?plotmath for how to get the hat on the $y$)
- Make a single plot consisting of nine residual plots in a 3x3 arrangement. Each of these nine should have points of the form $(x_{ji}, e_i)$, where $x_{ji}$ is the $i^{\text{th}}$ observation of the $j^{\text{th}}$ independent variable. All variables other than PBE are independent variables here. None of these plots requires a main title, but each should have a $y$-axis label "$e_i$" and an $x$-axis label consisting of the independent variable's name.

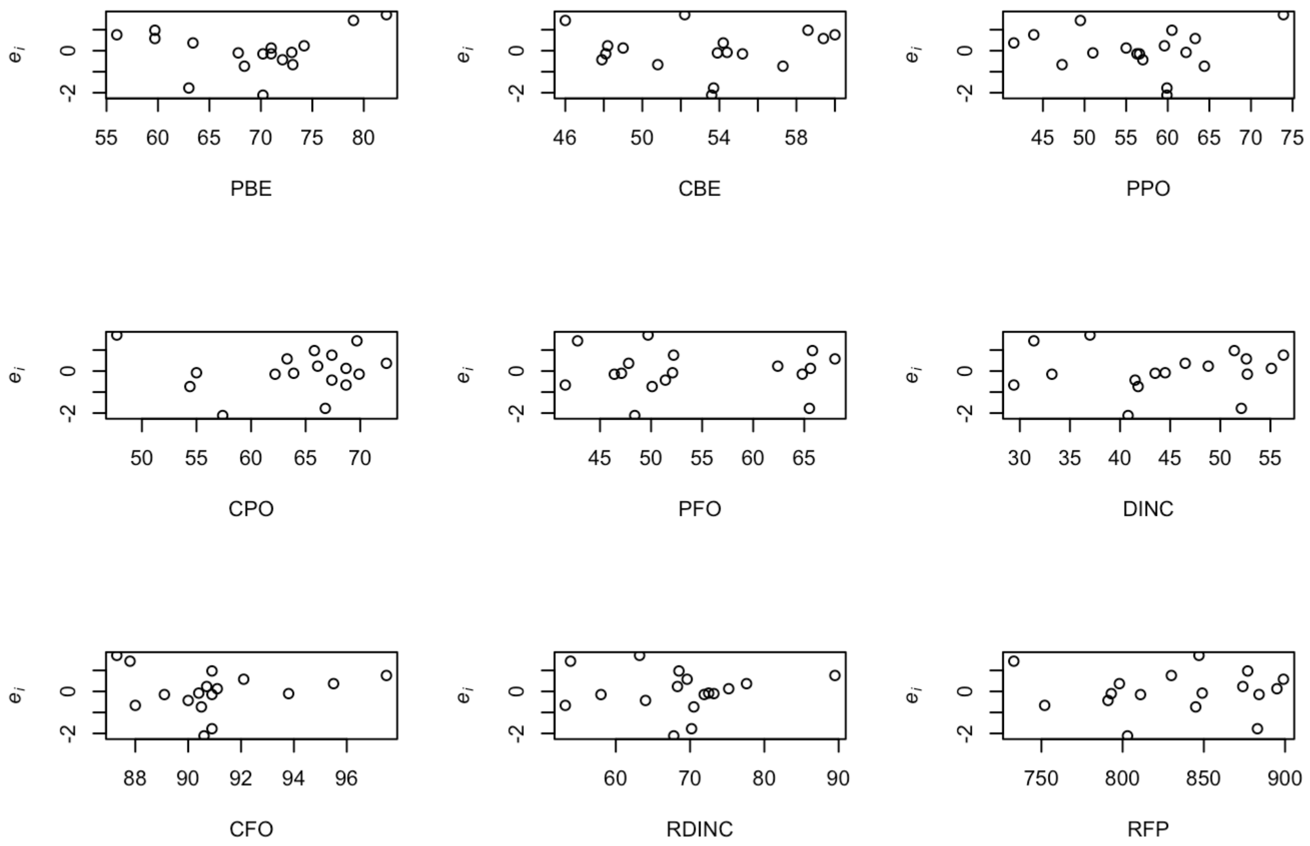
```
a = lm(t$PBE ~ t$CBE + t$PPO + t$CPO + t$PFO + t$DINC + t$CFO + t$RDINC + t$RFP)
summary(a)
```

```
##
## Call:
## lm(formula = t$PBE ~ t$CBE + t$PPO + t$CPO + t$PFO + t$DINC +
##     t$CFO + t$RDINC + t$RFP)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.11631 -0.43013 -0.07982  0.58022  1.71517
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 225.769097  76.551742   2.949   0.0184 *
## t$CBE        -1.415606   0.180946  -7.823 5.13e-05 ***
## t$PPO        -0.476230   0.330411  -1.441   0.1875
## t$CPO        -0.905734   0.401141  -2.258   0.0539 .
## t$PFO         0.853582   2.225781   0.383   0.7113
## t$DINC       -1.275225   2.870046  -0.444   0.6686
## t$CFO        -0.215570   0.936563  -0.230   0.8237
## t$RDINC       0.531204   1.890445   0.281   0.7858
## t$RFP        -0.003525   0.128698  -0.027   0.9788
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.412 on 8 degrees of freedom
## Multiple R-squared:  0.9792, Adjusted R-squared:  0.9584
## F-statistic: 47.03 on 8 and 8 DF,  p-value: 6.256e-06
```

```
plot(a$fitted.values, a$residuals, main = "Residual Plot: Beef", xlab = expression(ha
t(y[i])), ylab = expression(italic(e[i])))
```



```
m = matrix(data = c(1,2,3,4,5,6,7,8,9), nrow = 3, ncol = 3, byrow = TRUE)
layout(m)
plot(t$PBE, a$residuals,xlab = "PBE", ylab = expression(italic(e[i])))
plot(t$CBE, a$residuals,xlab = "CBE", ylab = expression(italic(e[i])))
plot(t$PPO, a$residuals,xlab = "PPO", ylab = expression(italic(e[i])))
plot(t$CPO, a$residuals,xlab = "CPO", ylab = expression(italic(e[i])))
plot(t$PFO, a$residuals,xlab = "PFO", ylab = expression(italic(e[i])))
plot(t$DINC, a$residuals,xlab = "DINC", ylab = expression(italic(e[i])))
plot(t$CFO, a$residuals,xlab = "CFO", ylab = expression(italic(e[i])))
plot(t$RDINC, a$residuals,xlab = "RDINC", ylab = expression(italic(e[i])))
plot(t$RFP, a$residuals,xlab = "RFP", ylab = expression(italic(e[i])))
```

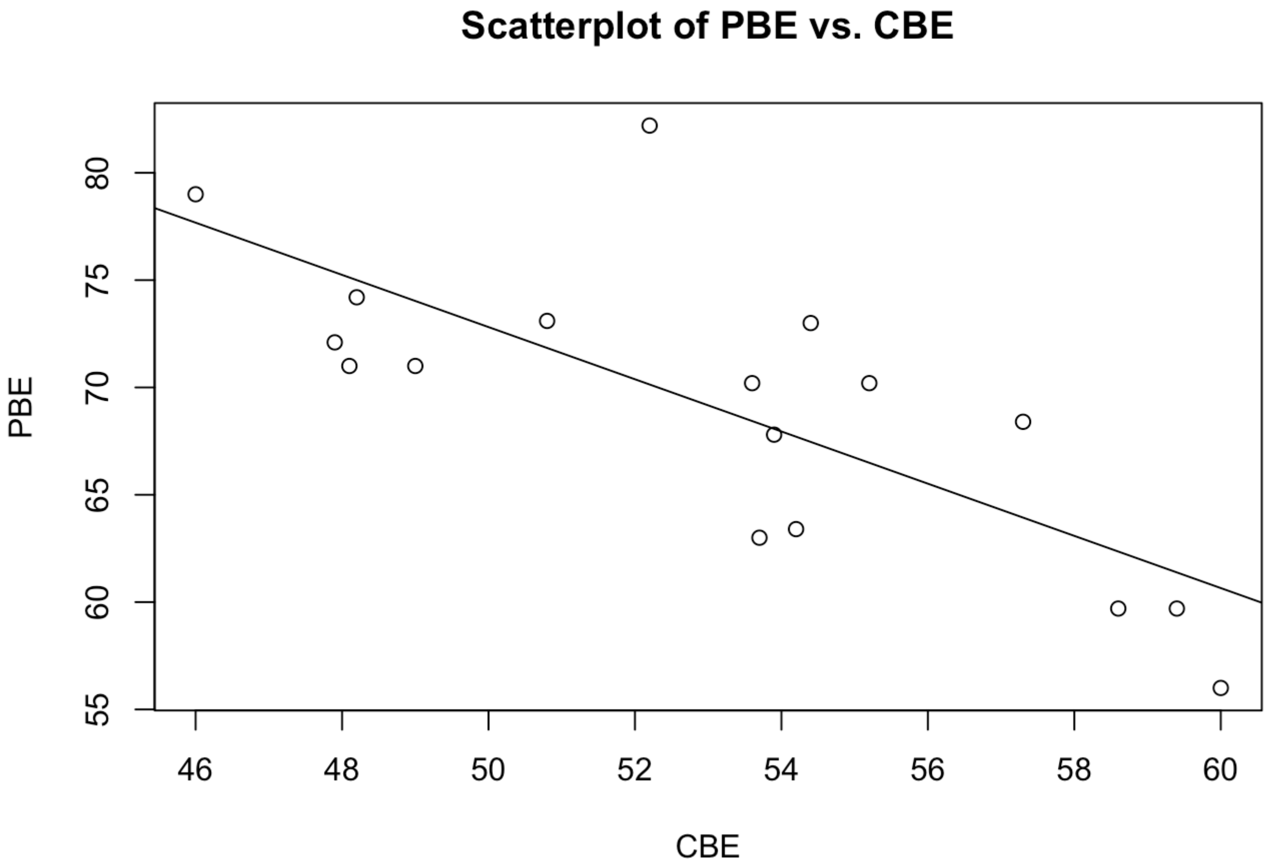


# Simple linear regression

- Look at your model summary to find the x variable whose model coefficient is most significantly different from 0. (You don’t have to write R code to find this other variable–just read your model summary.)
- Make a simple linear regression model for PBE vs. this x.
- Make a scatterplot of PBE vs. this x.
  - Add the simple regression line to your scatterplot.
  - Include a reasonable title and axis labels.

**CBE is the x variable whose model coefficient is most significantly different from 0.**

```
m = matrix(data = c(1), nrow = 1, ncol = 1)
layout(m)
simple = lm(t$PBE ~ t$CBE)
plot(t$PBE ~ t$CBE, main = "Scatterplot of PBE vs. CBE", xlab = "CBE", ylab = "PBE")
abline(reg = simple)
```



```
summary(simple)
```

```
##
## Call:
## lm(formula = t$PBE ~ t$CBE)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.3149 -3.2682 -0.8034  1.7635 12.0610
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 133.6190    14.6366   9.129 1.63e-07 ***
## t$CBE        -1.2161     0.2749  -4.424 0.000492 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.706 on 15 degrees of freedom
## Multiple R-squared:  0.5662, Adjusted R-squared:  0.5372
## F-statistic: 19.57 on 1 and 15 DF,  p-value: 0.0004923
```

## Are the coefficients (y-intercept and slope in the x direction) the same for this second simple linear regression model as they are in the first multiple regression model?

**The coefficients (y-intercept and slope in the x direction) are not the same for this second simple linear regression model as they are in the first multiple regression model.**

**In the first multiple regression model, slope is -1.415606 and y-intercept is 225.769097.**

**In the second simple linear regression model, slope is -1.2161 and y-intercept is 133.6190.**

# Part 3: Regression model including confidence bands

## Create a simulated bivariate data set consisting of n=100 $(x_i, y_i)$ pairs:

- Generate n random $x$-coordinates $x_i$ from $N(0, 1)$.
- Generate n random errors, $\epsilon_i$, from $N(0, \sigma)$, using $\sigma = 4$.
- Set $y_i = \beta_0 + \beta_1 x_i + \epsilon_i$, where $\beta_0 = 2, \beta_1 = 3$, and $\epsilon_i \sim N(0, 4)$. (That is, $y$ is a linear function of $x$, plus some random noise.)

(Now we have simulated data. We'll pretend that we don't know the true y-intercept $\beta_0 = 2$, the true slope $\beta_1 = 3$, the true $\sigma = 4$, or the true errors $\epsilon_i$. All we know are the data, $(x_i, y_i)$. We'll let linear regression estimate the coefficients.)

## Make a graph of the data and model:

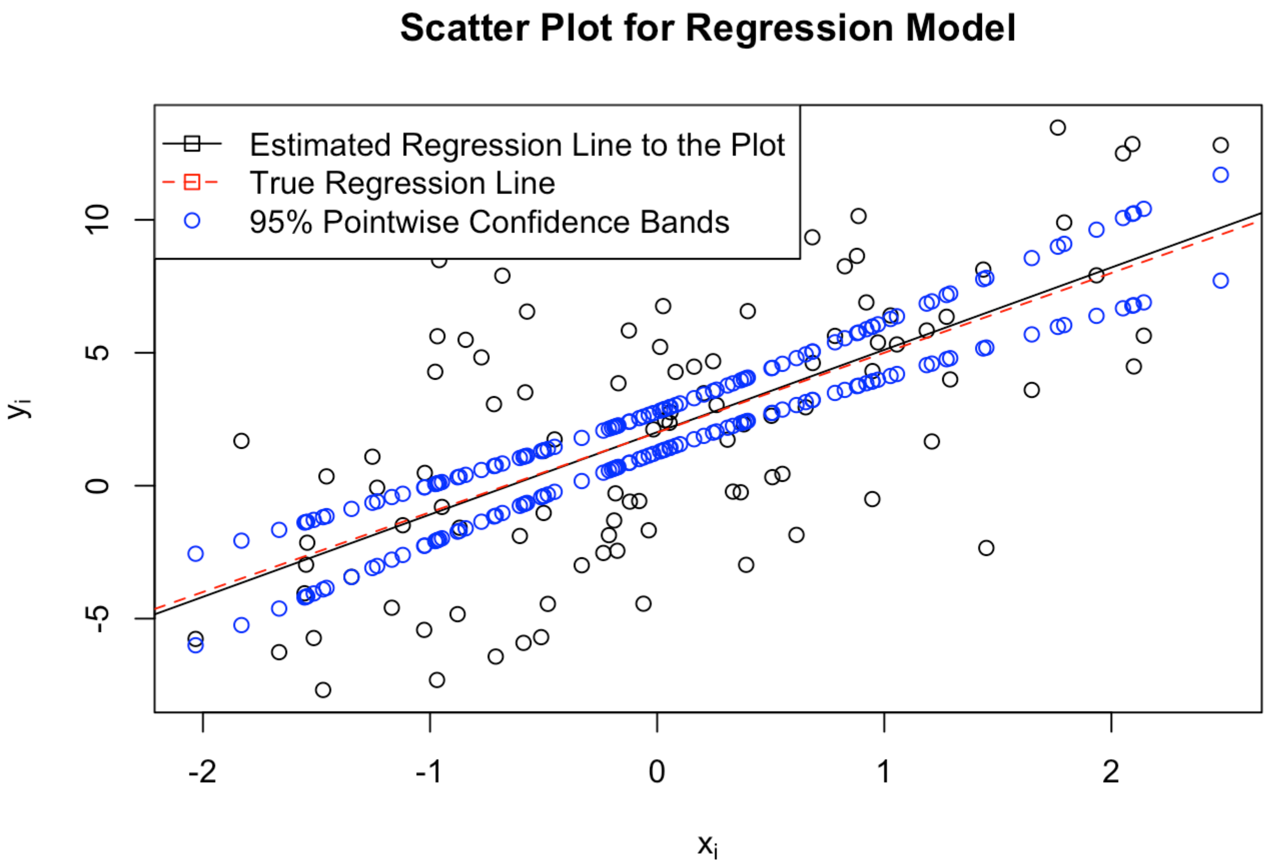
- Make a scatterplot of the data, $(x_i, y_i)$.
- Estimate a linear regression model of the form $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i$.
- Display a summary of the model; check that the estimated coefficients are near the true $\beta_0 = 2$ and $\beta_1 = 3$.
- Add a solid black estimated regression line to the plot.
- Add a dashed red true line (y = 2 + 3x) to the plot.
- Add dotted blue 95% pointwise confidence bands that consist, for each prediction $(x_i, \hat{y}_i)$, of a vertical confidence interval around $\hat{y}_i$ centered at $(x_i, \hat{y}_i)$; the formula is $\hat{y}_i \pm t_{n-2,\alpha/2} s_{\hat{y_i}}$, where
    - $\hat{y}_i$ is the predicted $y$ at $x = x_i$ (this is available in the model you calculated)
    - $e_i = y_i - \hat{y}_i$, the $i^{\text{th}}$ residual (this estimate of $\epsilon_i$ is available in the model you calculated)
    - $s = \sqrt{\frac{\sum_{i=1}^{n} e_i^2}{n-2}}$ (this is an estimate of $\sigma$)
    - $s_{\hat{y_i}} = s\sqrt{\frac{1}{n} + \frac{(x_i - \bar{x})^2}{\sum_{i=1}^{n} (x_i - \bar{x})^2}}$
    - $t_{n-2,\alpha/2}$ is the number that cuts off a right-tail area .025 from a Student's $t$ distribution with $n - 2$ degrees of freedom
- Add a legend identifying each of the black, red, and blue lines.

Hint: These calculations might look hard, but they go quickly with R. See Quiz 2's questions 8-11 for examples of efficiently translating sums into R code.

```
n = 100
x = rnorm(n = n)
errors = rnorm(n = 100, sd = 4)
y = 2 + 3*x + errors
plot(x,y, main = "Scatter Plot for Regression Model", xlab = expression(x[i]), ylab =
 expression(y[i]))
m1 = lm(y ~ x)
summary(m1)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.8486 -3.1450  0.1336  2.8736  9.4441
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.0158     0.3860   5.223 9.90e-07 ***
## x             3.0984     0.3773   8.211 8.96e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.858 on 98 degrees of freedom
## Multiple R-squared:  0.4076, Adjusted R-squared:  0.4016
## F-statistic: 67.43 on 1 and 98 DF,  p-value: 8.959e-13
```

```
abline(reg = m1, col = "black")
abline(a = 2, b = 3, col = "red", lty = 2)
t = qt(.025, df = n - 2, lower.tail = FALSE)
s = sqrt(sum(m1$residuals^2)/(n-2))
s.yhat = s * sqrt((1/n) + ((x-mean(x))^2/sum((x-mean(x))^2)))
points(x, m1$fitted.value + t*s.yhat, col = "blue")
points(x, m1$fitted.value - t*s.yhat, col = "blue")
legend("topleft", legend = c("Estimated Regression Line to the Plot", "True Regressio
n Line", "95% Pointwise Confidence Bands"), col = c("black", "red", "blue"), lty = c(
1,2,0), pch = c(0,0,1))
```


Scatter Plot for Regression Model
Estimated Regression Line to the Plot
True Regression Line
95% Pointwise Confidence Bands
$y_i$
$x_i$
-5
0
5
10
-2
-1
0
1
2