This material was produced for instructors using Llaudet, Elena and Kosuke Imai.
Data Analysis for Social Science: A Friendly and Practical Introduction. (Princeton University Press)
and should not be shared beyond those who are enrolled in this class.

Predicting Course Grades Using Midterm Scores
Part IV: Quantifying Uncertainty (with Solutions)

Let's continue to analyze real, historical, student performance data from the class. The dataset we will use is in the *grades.csv* file. Table 1 shows the names and descriptions of the variables in this dataset, where the unit of observation is students.

variable	description
midterm	students' scores in the midterm (from 0 to 100 points)
final	students' scores in the final exam (from 0 to 100 points)
overall	students' scores in the class overall (from 0 to 100 points)
gradeA	identifies students who earned an A or an A minus in the class

Table 1: Variables in "grades.csv"

In this problem set, we practice fitting a line to make predictions, creating scatter plots, adding the fitted line to the scatter plot, and constructing confidence intervals for our predictions.

As always, we start by loading and looking at the data:

```
## load and look at the data
grades <- read.csv("grades.csv") # reads and stores data
head(grades) # shows first observations
##   midterm final overall gradeA
## 1   79.25 47.00    69.2      0
## 2   96.25 87.75    94.3      1
## 3   58.25 37.75    62.0      0
## 4   54.50 62.00    72.4      0
## 5   83.00 39.75    72.4      0
## 6   41.75 49.50    59.5      0
```

1. First, let's fit the linear models that we will use to make predictions.

    a. Fit the following three linear models: (i) the linear model to predict final exam scores using midterm scores, (ii) the linear model to predict overall scores in the course using midterm scores, and (iii) the linear model to predict the probability of earning an A or an A- in the course using midterm scores. To specify the dataframe where the variables are stored, use the optional argument `data`, instead of using the `$` operator for each variable. Store each fitted line in an object. Call the three objects *fit_final*, *fit_overall*, and *fit_gradeA*, respectively. Then, run the names of the objects, *fit_final*, *fit_overall*, and *fit_gradeA*, so that R will provide you with the contents of each object. Finish by writing the fitted line for each model. In other words, provide the formula $\widehat{Y} = \widehat{\alpha} + \widehat{\beta}X$ where you specify each term (i.e., substitute Y for the name of the outcome variable, substitute $\widehat{\alpha}$ for the estimated value of the intercept coefficient, substitute $\widehat{\beta}$ for the estimated value of the slope coefficient, and substitute X for the name of the predictor.) (10 points)

This material was produced for instructors using Llaudet, Elena and Kosuke Imai.
Data Analysis for Social Science: A Friendly and Practical Introduction. (Princeton University Press)
and should not be shared beyond those who are enrolled in this class.

R code:

```
## fit and store the linear models that use midterm
fit_final <- lm(final ~ midterm, data=grades) # to predict final
fit_overall <- lm(overall ~ midterm, data=grades) # to predict overall
fit_gradeA <- lm(gradeA ~midterm, data=grades) # to predict gradeA
```

```
fit_final # provides contents of object fit_final
##
## Call:
## lm(formula = final ~ midterm, data = grades)
##
## Coefficients:
## (Intercept)        midterm
##     -6.0059       0.9704
```

```
fit_overall # provides contents of object fit_overall
##
## Call:
## lm(formula = overall ~ midterm, data = grades)
##
## Coefficients:
## (Intercept)        midterm
##     29.9834       0.6565
```

```
fit_gradeA # provides contents of object fit_gradeA
##
## Call:
## lm(formula = gradeA ~midterm, data = grades)
##
## Coefficients:
## (Intercept)        midterm
##    -1.34305      0.02122
```

(Note: Remember that to fit a linear model in R we use the lm() function. This function requires an argument of the type Y ~ X. Here, *final*, *overall*, and *gradeA* are the outcome variables, Y, and *midterm* is the treatment variable, X. To specify the dataframe where the variables are stored, we can use either the $ operator for each variable or the optional argument data.} Now, to store the fitted linear model as an object named *fit_final*, we use the assignment operator <- and specify *fit_final* to its left.)

Answers: The fitted lines are: (i) $\widehat{\text{final}} = -6 + 0.97\ midterm$

(ii) $\widehat{\text{overall}} = 30 + 0.66\ midterm$

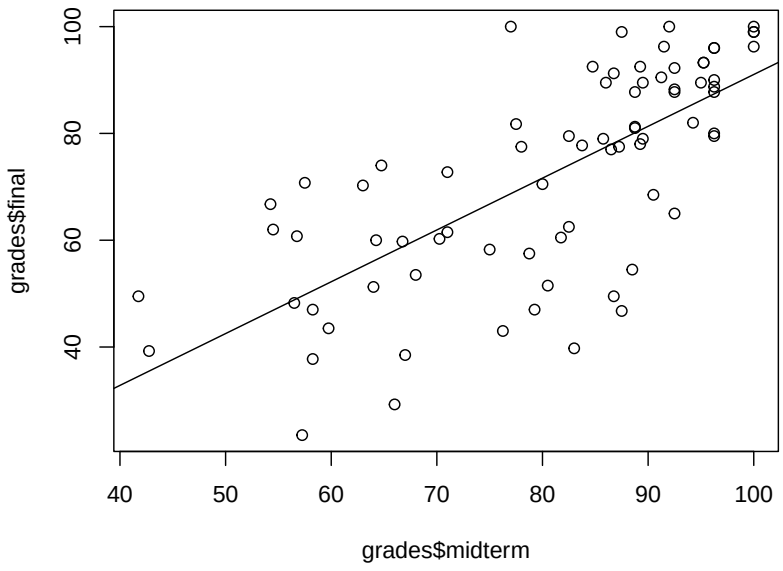
(iii) $\widehat{\text{gradeA}} = -1.34 + 0.02\ midterm$

This material was produced for instructors using Llaudet, Elena and Kosuke Imai.
Data Analysis for Social Science: A Friendly and Practical Introduction. (Princeton University Press)
and should not be shared beyond those who are enrolled in this class.

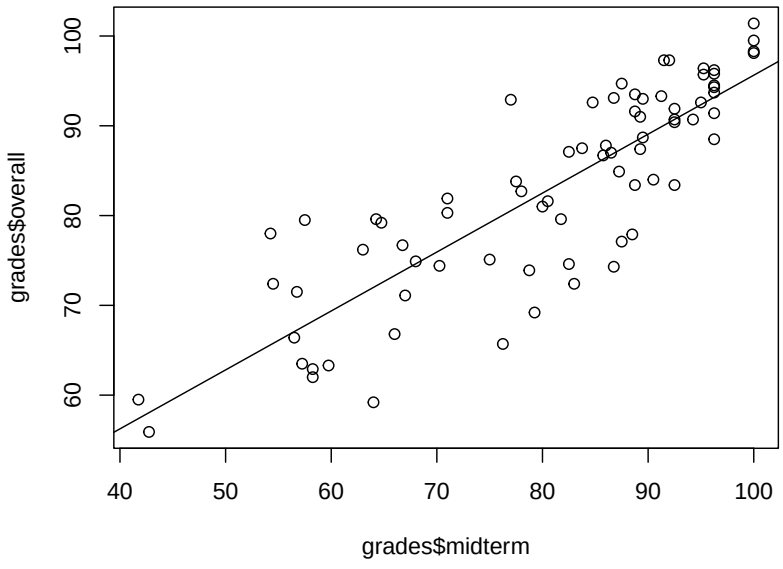
b. For each of the three linear models above, create a visualization of the relationship between X and Y and add the fitted line. (Hint: The functions plot() and abline() might be helpful here.) (R code only.) (10 points)

R code:

```
plot(grades$midterm, grades$final) # creates scatter plot
abline(fit_final) # adds fitted line
```

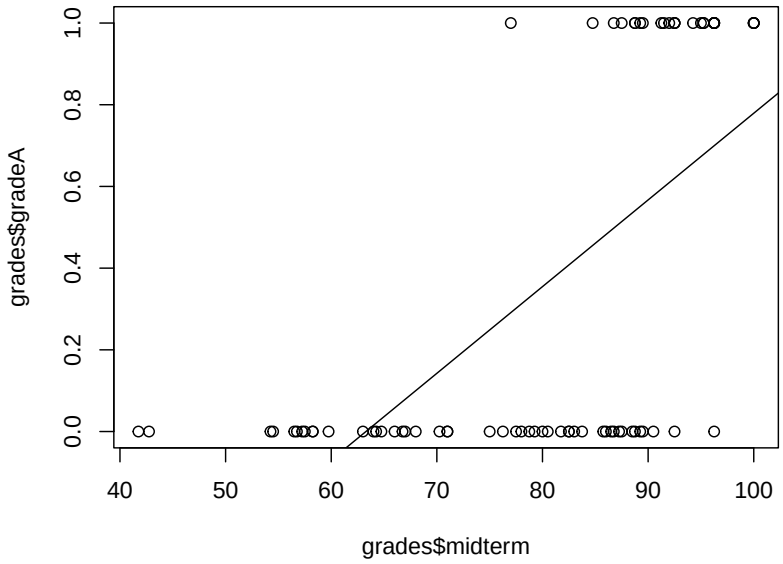



```
plot(grades$midterm, grades$overall) # creates scatter plot
abline(fit_overall) # adds fitted line
```



This material was produced for instructors using Llaudet, Elena and Kosuke Imai.
Data Analysis for Social Science: A Friendly and Practical Introduction. (Princeton University Press)
and should not be shared beyond those who are enrolled in this class.

```
plot(grades$midterm, grades$gradeA) # creates scatter plot
abline(fit_gradeA) # adds fitted line
```



(Note: To visualize the relationship between X and Y, we create a scatter plot with the function plot(). This function requires two arguments and in this order: (1) the X variable, (2) the Y variable. We always plot the X variable along the X axes, and the Y variable along the Y axes. Here, *midterm* is the X variable and *final*, *overall*, and *gradeA* are the Y variables. Alternatively, if we do not want the order of the arguments to matter, we could specify the names of the arguments, x and y, in the code. For example, plot(x=grades$midterm, y=grades$final) and plot(y=grades$final, x=grades$midterm) would produce the same scatter plot as the first one above. Now, to add the fitted line to the scatter plot, we use the function abline(). The only required argument is the name of the object where we have stored the fitted line, fit_final in this first case.)

2. Now, let's use the fitted linear models to make some predictions.

   a. Suppose that you earn 80 points in the midterm. Based on your performance in the midterm, what would be (i) your predicted final exam score, (ii) your predicted overall score, and (iii) your predicted probability of earning an A or A- in the course? Please show your calculations and then answer the question with a full sentence (including units of measurement). (5 points)

   Calculations:

$$\begin{aligned}\text{(i)}\quad \widehat{\text{final}} &= \widehat{\alpha} + \widehat{\beta}\,\text{midterm} \\ &= -6 + 0.97\,\text{midterm} \\ &= -6 + 0.97 \times 80 \quad \text{(if midterm=80)} \\ &= -6 + 77.6 = 71.6\end{aligned}$$

This material was produced for instructors using Llaudet, Elena and Kosuke Imai.
Data Analysis for Social Science: A Friendly and Practical Introduction. (Princeton University Press)
and should not be shared beyond those who are enrolled in this class.

$$\begin{aligned}\text{(ii)}\quad \widehat{\text{overall}} &= \widehat{\alpha} + \widehat{\beta}\,\text{midterm} \\ &= 30 + 0.66\,\text{midterm} \\ &= 30 + 0.66 \times 80 \quad \text{(if midterm=80)} \\ &= 30 + 52.8 = 82.8\end{aligned}$$

$$\begin{aligned}\text{(iii)}\quad \widehat{\text{gradeA}} &= \widehat{\alpha} + \widehat{\beta}\,\text{midterm} \\ &= -1.3431 + 0.0212\,\text{midterm} \\ &= -1.3431 + 0.0212 \times 80 \quad \text{(if midterm=80)} \\ &= -1.3431 + 1.696 = 0.3529\end{aligned}$$

Answer: If I earn 80 points in the midterm, I would predict that:

(i) I will earn 71.6 points in the final exam, on average.

(ii) I will earn 82.8 points overall in the class, on average.

(iii) My probability of earning an A or an A- is of 35.29%, on average.

(Note: $\widehat{Y}$ is in the same unit of measurement as $\overline{Y}$; in the first two cases, Y is non-binary and measured in points so $\overline{Y}$ and $\widehat{Y}$ are also measured in points. In the third case, Y is binary so $\overline{Y}$ and $\widehat{Y}$ are in percentages, after multiplying the outputs by 100. Recall: the mean of a binary variable should be interpreted as the proportion of the observations that have the characteristic identified by the variable. That is, it should be interpret in percentages, after multiplying the number by 100. Note: We could have arrive at the same conclusions by looking at the scatter plots with the fitted lines above. In each scatter plot, all we would need to do is: (i) find 80 on the X-axis, (ii) go up to the line, and (iii) find the value on the Y-axis associated with that point on the line.)

b. Because of potential noise in the data, there is some uncertainty around these predictions. Construct the 95% confidence interval for each of the three predictions using the function predict() and re-write your answers to the previous question accordingly. (25 points)

R code:

```
## compute 95% interval for prediction of final
predict(fit_final, # object with lm() output
    newdata=data.frame(midterm=80), #set value of X
     interval="confidence") # provide 95% confidence interval
##        fit      lwr      upr
## 1 71.62762 68.42561 74.82964
```

```
## compute 95% interval for prediction of overall
predict(fit_overall, # object with lm() output
    newdata=data.frame(midterm=80), #set value of X
     interval="confidence") # provide 95% confidence interval
##        fit      lwr      upr
## 1 82.50305 81.12341 83.88269
```

This material was produced for instructors using Llaudet, Elena and Kosuke Imai.
Data Analysis for Social Science: A Friendly and Practical Introduction. (Princeton University Press)
and should not be shared beyond those who are enrolled in this class.

```
## compute 95% interval for prediction of gradeA
predict( fit_gradeA, # object with lm() output
   newdata=data.frame(midterm=80), #set value of X
    interval ="confidence") # provide 95% confidence interval
##          fit       lwr       upr
## 1 0.3547382 0.269011 0.4404654
```

(Recall: The only required argument of the function predict() is the name of the object that contains the output of the lm() function. By default, this function produces a prediction for every observation in the dataset used to fit the linear model. To produce only one prediction based on a particular value of the predictor(s), we set the optional argument newdata to equal data.frame(), where inside the parentheses we specify the value of the predictor(s). To also produce the 95% confidence interval of that one prediction, we set the optional argument interval to equal "confidence". The first number R provides is the predicted outcome based on the specified (i) fitted linear model and (ii) value of the predictor. The next two numbers are the lower and upper limits of the 95% confidence interval.)

Answers: If I earn 80 points in the midterm, I would predict that:

(i) I am likely to earn between 68.43 and 74.83 points in the final exam, on average. (Note: The 95% confidence interval here is: [68.43, 74.83] because the lower bound is 68.43 and the upper bound is 74.83.)

(ii) I am likely to earn between 81.12 and 83.88 points overall in the class, on average. (Note: The 95% confidence interval here is: [81.12, 83.88] because the lower bound is 81.12 and the upper bound is 83.88.)

(iii) My probability of earning an A or an A- in the course is likely to be between 27% and 44%, on average. (Note: The 95% confidence interval here is: [0.27, 0.44] because the lower bound is 0.27 and the upper bound is 0.44. As we saw in the previous question, since Y is binary, $\widehat{Y}$ should be interpreted in percentages, after multiplying the output by 100.)

(Note: The predictions produced by the functions predict() above, shown as the first return values: 71.63, 82.5, and 0.35, are the exact same values we arrived at in the previous question when we did the calculations by hand using the fitted linear models. The small differences are due to our rounding to two decimals.)