Predicting Elections Using Betting Markets (with Solutions)

In this exercise, we aim to predict electoral outcomes based on data from the betting markets. For this purpose, we will analyze state-level data on the 2008 U.S. presidential election where the two main candidates were Obama (the Democratic candidate) and McCain (the Republican candidate) and model the relationship between (i) the closing price the day before the election of contracts from the online betting company Intrade and (ii) the real electoral outcomes. At Intrade, people trade contracts such as 'Obama to win the electoral college votes of Florida.' The market price of each contract fluctuates based on its sales. The price of each nominee's contract can be interpreted as the vote share that the market expects that nominee to receive in that state.

The dataset we will use is in the *intrade.csv* file. Table 1 shows the names and descriptions of the variables in this dataset, where the unit of observation is states (plus DC).

variable	description
state	state's two letter abbreviation
D_expected	closing price or expected vote share of the Democratic candidate's contract in that state (in percentages)
R_expected	closing price or expected vote share of the Republican candidate's contract in that state (in percentages)
D_real	real vote share received by the Democratic candidate in that state (in percentages)
R_real	real vote share received by the Republican candidate in that state (in percentages)

Table 1: Variables in "intrade.csv"

In this problem set, we practice creating new variables, creating histograms, computing correlations, creating scatter plots, fitting linear models, making predictions with the fitted line, and computing R^2.

As always, let's start by loading and exploring the data:

```
## load and look at the data
intrade <- read.csv("intrade.csv") # reads and stores data
head(intrade) # shows first observations
##   state D_expected R_expected D_real R_real
## 1    AK        6.0       94.0     38     59
## 2    AL        3.3       95.0     39     60
## 3    AR        3.9       90.0     39     59
## 4    AZ       18.9       80.2     45     54
## 5    CA       97.9        2.1     61     37
## 6    CO       91.5       12.0     54     45
```

And, to confirm that we have data on all 50 states plus DC, we can run:

```
dim(intrade) # provides dimensions: rows, columns
## [1] 51  5
```

1. First, let's create our *Y* and *X* variables, figure out their unit of measurement, and explore whether they are moderately to strongly linearly associated with each other.

   a. Create our *X* variable, that is our predictor (i.e., what we will use as the basis of our prediction): the expected democratic margin based on the betting markets, defined as the closing price of the Democratic candidate's contract minus the closing price of the Republican candidate's contract. Call this variable *expected_margin* and make sure to store it inside the dataframe *intrade*. (R code only) (2.5 points)

   R code:

```
## create our X variable
intrade$expected_margin <- intrade$D_expected - intrade$R_expected
```

   (To store the values of the difference between *intrade$D_expected* and *intrade$R_expected* as a variable named *expected_margin* inside the dataframe *intrade*, we use the assignment operator <- and specify intrade$expected_margin to the left of the assignment operator. Without intrade$ in front of expected_margin, we would create a variable outside the dataframe, as a new object, which is not what we want.)

   b. Create our *Y* variable, that is our outcome variable (i.e., what we want to predict): the real democratic margin, defined as the real vote share received by the Democratic candidate minus the real vote share received by the Republican candidate. Call this variable *real_margin* and make sure to store it inside the dataframe *intrade*. (R code only) (2.5 points)

   R code:

```
## create our Y variable
intrade$real_margin <- intrade$D_real - intrade$R_real
```

   (See note above.)

   c. Use the function head() to look at the first few observations of *intrade* again to ensure that the two new variables were created correctly. In what unit of measurement are the two new variables, *expected_margin* and *real_margin*? (5 points)

   R code:

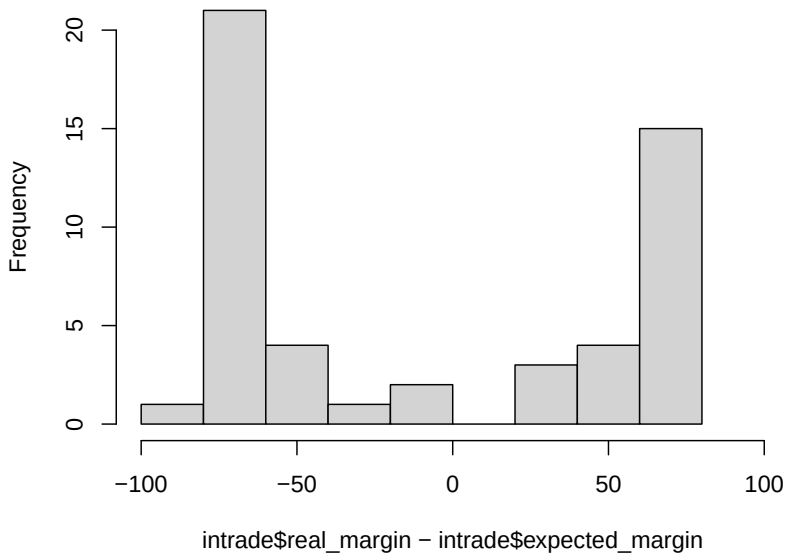
```
## look at first few observations
head(intrade)
##   state D_expected R_expected D_real R_real expected_margin real_margin
## 1    AK        6.0       94.0      38     59          -88.0         -21
## 2    AL        3.3       95.0      39     60          -91.7         -21
## 3    AR        3.9       90.0      39     59          -86.1         -20
## 4    AZ       18.9       80.2      45     54          -61.3          -9
## 5    CA       97.9        2.1      61     37           95.8          24
## 6    CO       91.5       12.0      54     45           79.5           9
```

Answer: Both *expected_margin* and *real_margin* are measured in percentage points since they are the result of computing the difference between two percentages. By looking at the first few observations above, we can confirm that the two new variables were created correctly. For example, the first value of *expected_margin* is -88 percentage points, which makes sense since 6% - 94% = -88 percentage points. And, the first value of *real_margin* is -21 percentage points, which also makes sense since 38% - 59% = -21 percentage points.

d. Do the betting markets do a good job in predicting the real democratic margin? To answer this question, create the histogram of the differences between *real_margin* and *expected_margin*. If the betting markets do a good job in predicting the real democratic margin, most of the observations in the histogram will be around 0 because the value of *real_margin* will equal the value of *expected_margin*. Is that what you observe in the histogram? A yes/no answer will suffice. (5 points)

R code:

```
## create histogram of the differences between real_margin and expected_margin
hist(intrade$real_margin-intrade$expected_margin)
```



(Recall: The function to create a histogram is hist(). The only required argument is the code identifying the variable or the formula of variables, as is the case here.)

Answer: No, most observations are far away from zero, which means that the betting markets do not do a good job in predicting the real democratic margin. So, we should not just use the value of *expected_margin* as our prediction of *real_margin*.

e. Is the relationship between *expected_margin* and *real_margin* moderate to strongly linear? If so, we could model the relationship between *expected_margin* and *real_margin* with a line and then use that line to predict *real_margin* based on the value of *expected_margin*. Please answer this question, by computing the correlation coefficient between *expected_margin* and *real_margin*. A yes/no answer will suffice. (5 points)

R code:

```
cor(intrade$expected_margin, intrade$real_margin) # computes correlation
## [1] 0.8538204
```

(Recall: The function in R to compute a correlation coefficient is cor(). The only two required arguments are the code identifying the two variables. The order of the variables does not matter since cor(X,Y) = cor(Y,X).)

Answer: Yes, the relationship between the two variables is moderate to strongly linear since their correlation coefficient is pretty high and much closer to 1 than to 0. This means that we can build a reasonably good predictive linear model using *expected_margin* to predict *real_margin*.

2. Second, let's fit the linear model that we will use to make predictions.

   a. Use the function lm() to fit a linear model to summarize the relationship between *expected_margin* and *real_margin* and store the output in an object called *fit*. Then, ask R to provide the contents of *fit* by running its name. (R code only.) (5 points)

   R code:

```
# fit linear model and store it in an object called fit
fit <- lm(intrade$real_margin ~ intrade$expected_margin)
```

```
fit # provides contents of object
##
## Call:
## lm(formula = intrade$real_margin ~ intrade$expected_margin)
##
## Coefficients:
##                (Intercept)  intrade$expected_margin
##                     1.3027                  0.2291
```

   (Recall: The function lm() fits a linear model. It requires a function of the type Y ~ X, where Y identifies the Y variable (*real_margin*, in this case) and X identifies the X variable (*expected_margin*, in this case). To specify the dataframe where the variables are stored, we can use either the $ operator (as in the code above) or the optional argument data. If we wanted to use the latter, the code to fit the linear model would be lm(real_margin ~ expected_margin, data = intrade).)

   b. What is the fitted line? 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.) (5 points)

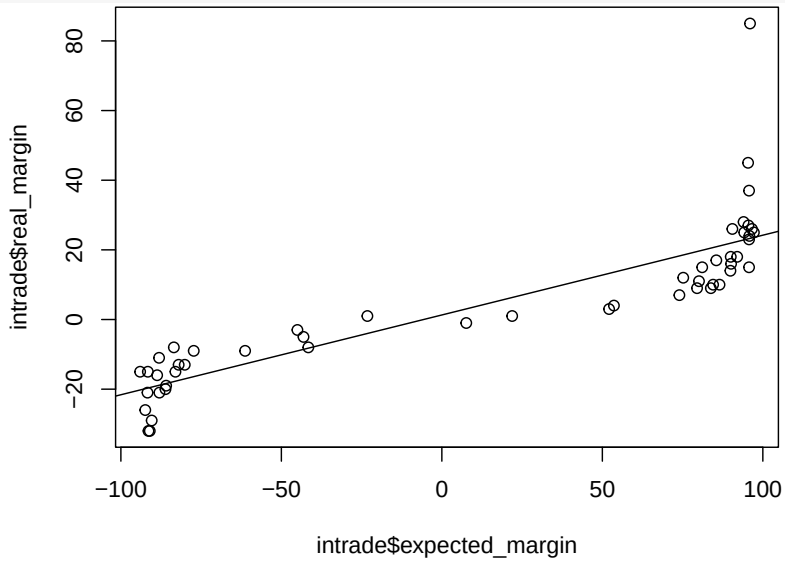
   Answer: $\widehat{\text{real\_margin}} = 1.3 + 0.23$ *expected_margin*

   (Note: Y is *real_margin*, $\widehat{\alpha}$=1.3, $\widehat{\beta}$=0.23, and X is *expected_margin*.)

c. Create a visualization of the relationship between *expected_margin* and *real_margin* and add the fitted line to the graph using the function abline(). (R code only.). (5 points)

R code:

```
## create scatter plot
plot(intrade$expected_margin, intrade$real_margin)
## add fitted line
abline(fit)
```



(Recall: A scatter plot is the graphical representation of the relationship between two variables. The function in R to create a scatter plot is plot(). It requires two arguments (separated by a comma) and in this particular order: (1) the code identifying the variable to be plotted along the x-axis, and (2) the the code identifying the variable to be plotted along the y-axis. We always plot the predictor along the X-axis and the outcome variable along the Y-axis. 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=intrade$expected_margin, y= intrade$real_margin) and plot(y= intrade$real_margin, x=intrade$expected_margin) would produce the same scatter plot as the one above. Recall: The function abline() adds lines to the most recently created graph. To add the fitted line, we specify as the main argument the name of the object where we stored the output of the lm() function, *fit* in this case.)

3. Now, let's use the fitted line to make some predictions.

a. Computing $\widehat{Y}$ based on X: Suppose the day before the next U.S. presidential election, you notice that the difference between the closing price of the Democratic candidate's contract and the closing price of the Republican candidate's contract (i.e., the expected democratic margin) in a particular state equals 20 percentage points. By how much would you predict that the Democratic candidate will win that state? (5 points)

Calculations:

$$\begin{aligned}\widehat{\text{real\_margin}} &= \widehat{\alpha} + \widehat{\beta}\,\text{expected\_margin} \\ &= 1.3 + 0.23\ \text{expected\_margin} \\ &= 1.3 + 0.23 \times 20 \ \text{(if expected\_margin=20)} \\ &= 1.3 + 4.6 = 5.9\end{aligned}$$

Answer: If the expected democratic margin in a state is of 20 percentage points the night before the election, I would predict that the real democratic margin in that state will be of 5.9 percentage points, on average. That is, I would predict that the Democratic candidate will win that state by 5.9 percentage points, on average. (Note: $\widehat{Y}$ is in the same unit of measurement as $\overline{Y}$; in this case, Y is non-binary and measured in percentage points so $\overline{Y}$ and $\widehat{Y}$ are also measured in percentage points.)

b. Computing $\triangle\widehat{Y}$ based on $\triangle X$: What is the predicted change in the real democratic margin associated with an increase in the expected democratic margin of 10 percentage points? (5 points)

Calculations:

$$\begin{aligned}\triangle\widehat{\text{real\_margin}} &= \widehat{\beta}\,\triangle\text{expected\_margin} \\ &= 0.23 \times \triangle\text{expected\_margin} \\ &= 0.23 \times 10 \ \text{(if } \triangle \text{ expected\_margin=10)} \\ &= 2.3\end{aligned}$$

Answer: An increase in the expected democratic margin of 10 percentage points is associated with a predicted increase in the real democratic margin of 2.3 percentage points, on average. (Note: $\triangle\widehat{Y}$ is in the same unit of measurement as $\triangle\overline{Y}$; in this case, Y is non-binary and measured in percentage points so $\triangle\overline{Y}$ and $\triangle\widehat{Y}$ are also measured in percentage points.)

c. To explore how good the model is at making predictions, compute the R^2 of the fitted model and interpret its value. (5 points)

R code:

```
cor(intrade$expected_margin, intrade$real_margin)^2 # computes R^2
## [1] 0.7290092
```

Answer: The linear model using *expected_margin* as a predictor explains 73% of the variation of *real_margin*, which means this linear model is relatively good at predicting electoral outcomes. (Note: R^2 measures the proportion of the variation of the outcome variable explained by the model. In the simple linear model: $R^2 = \text{cor(X,Y)}^2$. Since R^2 is relatively close to 1, it looks like a fairly good predictive model. The prediction errors—the vertical distance between the dots and the line—are relatively small.)