Data606 Lab8 - Introduction to linear regression

Lab Completed by Chad Bailey

Batter up

The movie Moneyball focuses on the "quest for the secret of success in baseball". It follows a low-budget team, the Oakland Athletics, who believed that underused statistics, such as a player's ability to get on base, betterpredict the ability to score runs than typical statistics like home runs, RBIs (runs batted in), and batting average. Obtaining players who excelled in these underused statistics turned out to be much more affordable for the team.

In this lab we'll be looking at data from all 30 Major League Baseball teams and examining the linear relationship between runs scored in a season and a number of other player statistics. Our aim will be to summarize these relationships both graphically and numerically in order to find which variable, if any, helps us best predict a team's runs scored in a season.

The data

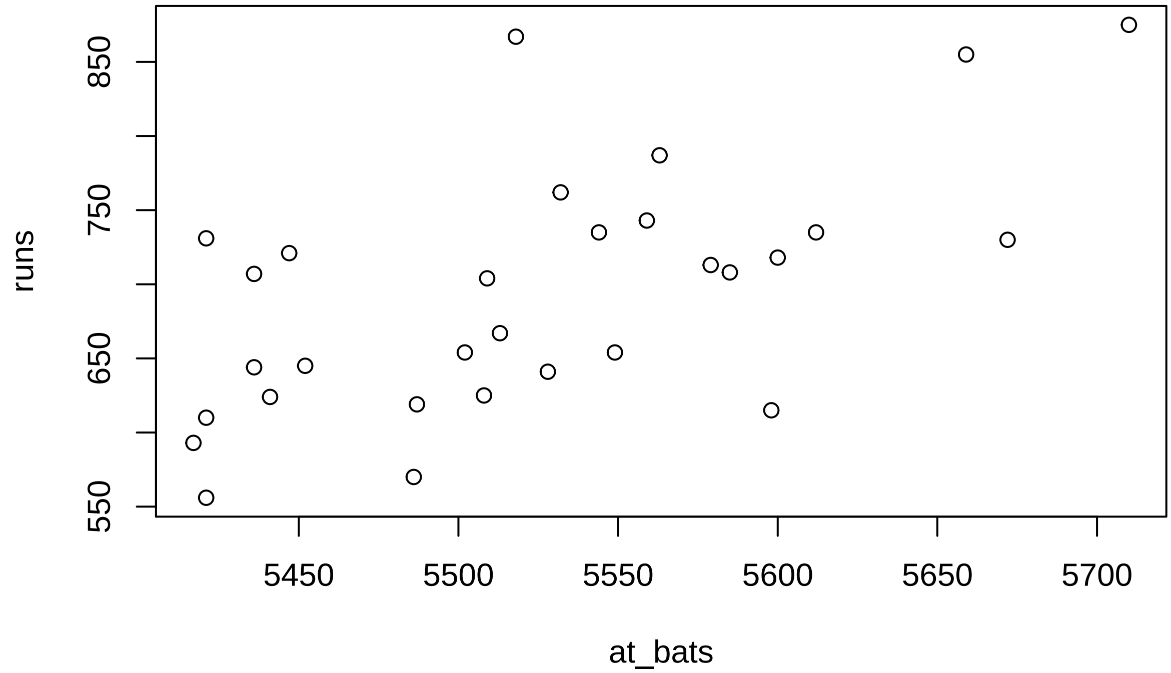
Let's load up the data for the 2011 season.

```
load("more/mlb11.RData")
```

In addition to runs scored, there are seven traditionally used variables in the data set: at-bats, hits, home runs, batting average, strikeouts, stolen bases, and wins. There are also three newer variables: on-base percentage, slugging percentage, and on-base plus slugging. For the first portion of the analysis we'll consider the seven traditional variables. At the end of the lab, you'll work with the newer variables on your own.

1. What type of plot would you use to display the relationship between `runs` and one of the other numerical variables? Plot this relationship using the variable `at_bats` as the predictor. Does the relationship look linear? If you knew a team's `at_bats`, would you be comfortable using a linear model to predict the number of runs?

```
## Student response to exercise 1

##  The appropriate display would be a scatter plot
    plot(data = mlb11, runs ~ at_bats)
```



```
##  The scatterplot appears to show a modest linear relationshp.
##  Yes, I would be comfortable using a linear model to predict runs using
##  at_bats. However, given the modest relationship, I would continue to
##  look for other variables with possibly stronger relationships.
```

If the relationship looks linear, we can quantify the strength of the relationship with the correlation coefficient.

```
cor(mlb11$runs, mlb11$at_bats)
```

```
## [1] 0.610627
```

Sum of squared residuals

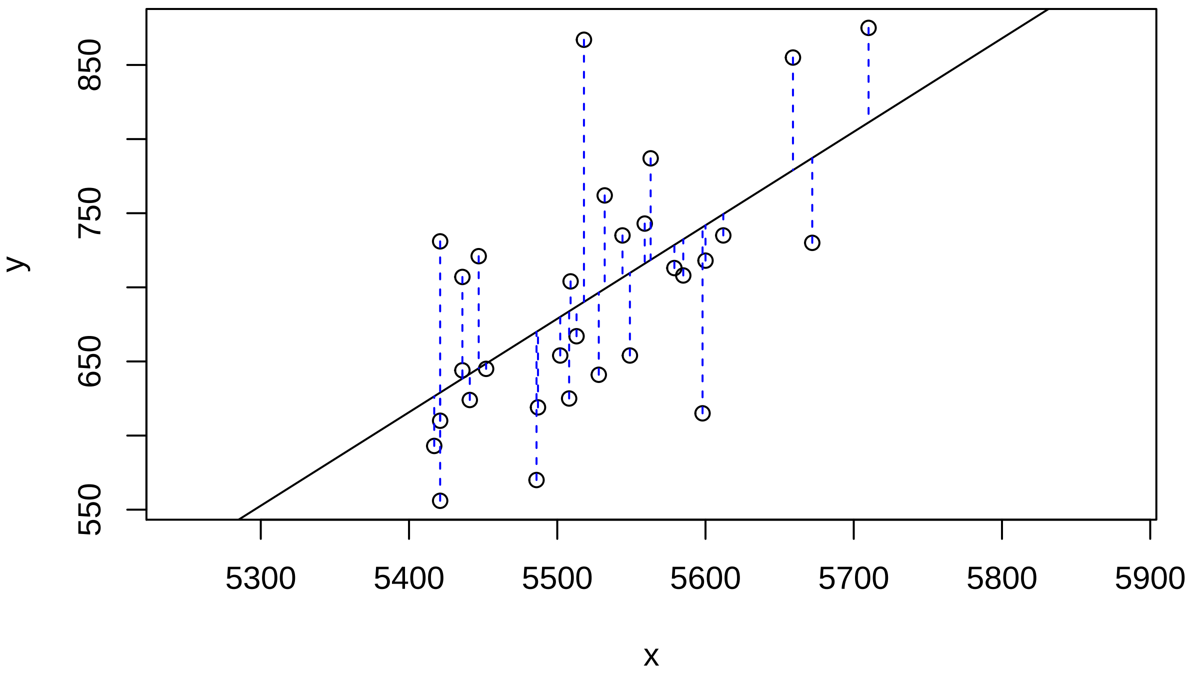
Think back to the way that we described the distribution of a single variable. Recall that we discussed characteristics such as center, spread, and shape. It's also useful to be able to describe the relationship of two numerical variables, such as `runs` and `at_bats` above.

2. Looking at your plot from the previous exercise, describe the relationship between these two variables. Make sure to discuss the form, direction, and strength of the relationship as well as any unusual observations.

```
##  Student response to exercise 2

##  The scatter displays a moderate positive correlation and has a least one
##  possible outlier in the positive direction.
```

Just as we used the mean and standard deviation to summarize a single variable, we can summarize the relationship between these two variables by finding the line that best follows their association. Use the following interactive function to select the line that you think does the best job of going through the cloud of points.

```
plot_ss(x = mlb11$at_bats, y = mlb11$runs)
```



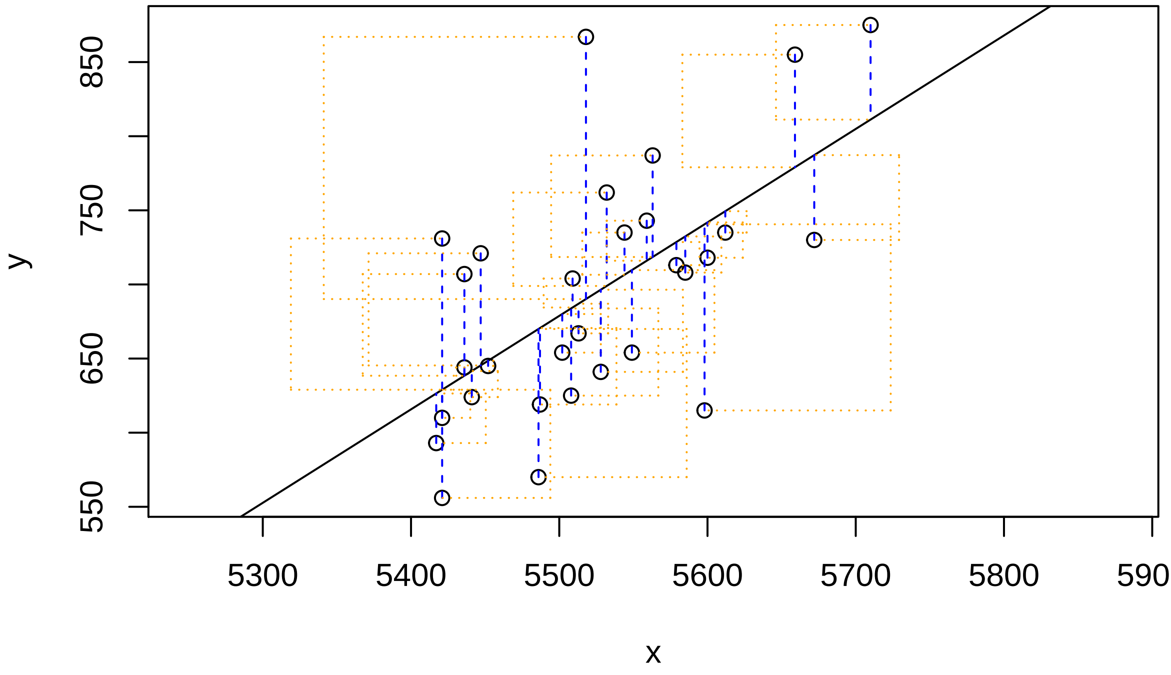
```
## Click two points to make a line.

## Call:
## lm(formula = y ~ x, data = pts)
##
## Coefficients:
## (Intercept)            x
##  -2789.2429       0.6305
##
## Sum of Squares:  123721.9
```

After running this command, you'll be prompted to click two points on the plot to define a line. Once you've done that, the line you specified will be shown in black and the residuals in blue. Note that there are 30 residuals, one for each of the 30 observations. Recall that the residuals are the difference between the observed values and the values predicted by the line:

$$e_i = y_i - \hat{y}_i$$

The most common way to do linear regression is to select the line that minimizes the sum of squared residuals. To visualize the squared residuals, you can rerun the plot command and add the argument `showSquares = TRUE`.

```
plot_ss(x = mlb11$at_bats, y = mlb11$runs, showSquares = TRUE)
```



```
## Click two points to make a line.

## Call:
## lm(formula = y ~ x, data = pts)
##
## Coefficients:
## (Intercept)            x
##  -2789.2429       0.6305
##
## Sum of Squares:  123721.9
```

Note that the output from the `plot_ss` function provides you with the slope and intercept of your line as well as the sum of squares.

3. Using `plot_ss`, choose a line that does a good job of minimizing the sum of squares. Run the function several times. What was the smallest sum of squares that you got? How does it compare to your neighbors?

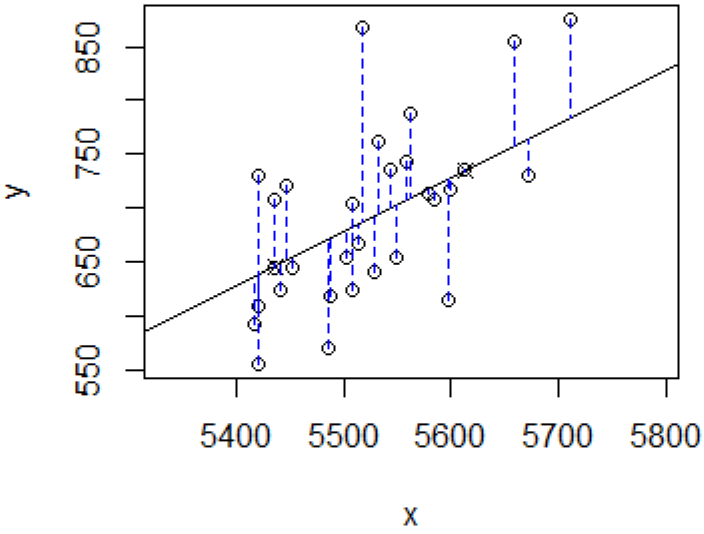


Figure 1: Image of plot_ss

```
##  Student response to exercise 3

##  After multiple selections of two observed points in the data set, the smallest
##  observed sum of squares was 127142.3
```

The linear model

It is rather cumbersome to try to get the correct least squares line, i.e. the line that minimizes the sum of squared residuals, through trial and error. Instead we can use the `lm` function in R to fit the linear model (a.k.a. regression line).

```
m1 <- lm(runs ~ at_bats, data = mlb11)
```

The first argument in the function `lm` is a formula that takes the form `y ~ x`. Here it can be read that we want to make a linear model of `runs` as a function of `at_bats`. The second argument specifies that R should look in the `mlb11` data frame to find the `runs` and `at_bats` variables.

The output of `lm` is an object that contains all of the information we need about the linear model that was just fit. We can access this information using the summary function.

```
summary(m1)
```

```
##
## Call:
## lm(formula = runs ~ at_bats, data = mlb11)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -125.58  -47.05  -16.59   54.40  176.87
##
```

```
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2789.2429   853.6957  -3.267 0.002871 **
## at_bats         0.6305     0.1545   4.080 0.000339 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 66.47 on 28 degrees of freedom
## Multiple R-squared:  0.3729, Adjusted R-squared:  0.3505
## F-statistic: 16.65 on 1 and 28 DF,  p-value: 0.0003388
```

Let's consider this output piece by piece. First, the formula used to describe the model is shown at the top. After the formula you find the five-number summary of the residuals. The "Coefficients" table shown next is key; its first column displays the linear model's y-intercept and the coefficient of `at_bats`. With this table, we can write down the least squares regression line for the linear model:

$$\hat{y} = -2789.2429 + 0.6305 * atbats$$

One last piece of information we will discuss from the summary output is the Multiple R-squared, or more simply, R^2. The R^2 value represents the proportion of variability in the response variable that is explained by the explanatory variable. For this model, 37.3% of the variability in runs is explained by at-bats.

4. Fit a new model that uses `homeruns` to predict `runs`. Using the estimates from the R output, write the equation of the regression line. What does the slope tell us in the context of the relationship between success of a team and its home runs?

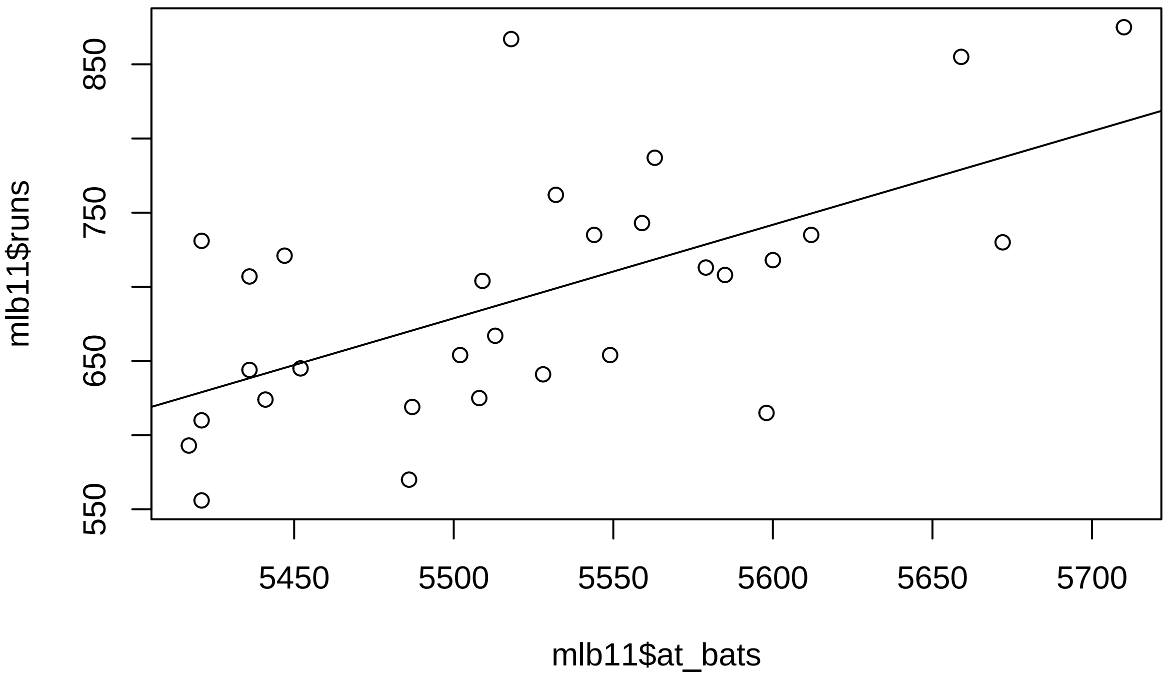
```
##  Student response to exercise4

    m2 <- lm(runs ~ homeruns, data = mlb11)
    summary(m2)
```

```
##
## Call:
## lm(formula = runs ~ homeruns, data = mlb11)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -91.615 -33.410   3.231  24.292 104.631
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 415.2389    41.6779   9.963 1.04e-10 ***
## homeruns      1.8345     0.2677   6.854 1.90e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 51.29 on 28 degrees of freedom
## Multiple R-squared:  0.6266, Adjusted R-squared:  0.6132
## F-statistic: 46.98 on 1 and 28 DF,  p-value: 1.9e-07
##  The optimized linear model for 'homeruns' predicting 'runs' is:
##  y = 415.2389 + 1.8345 * x

##  The model's slope (1.8345) shows a positive relationship between total runs
##  and home runs.
```

Prediction and prediction errors

Let's create a scatterplot with the least squares line laid on top.

```
plot(mlb11$runs ~ mlb11$at_bats)
abline(m1)
```



The function `abline` plots a line based on its slope and intercept. Here, we used a shortcut by providing the model `m1`, which contains both parameter estimates. This line can be used to predict y at any value of x. When predictions are made for values of x that are beyond the range of the observed data, it is referred to as *extrapolation* and is not usually recommended. However, predictions made within the range of the data are more reliable. They're also used to compute the residuals.

5. If a team manager saw the least squares regression line and not the actual data, how many runs would he or she predict for a team with 5,579 at-bats? Is this an overestimate or an underestimate, and by how much? In other words, what is the residual for this prediction?

```
##  Student response to exercise 5

    mlb11[which(mlb11$at_bats==5579), ]
```

```
##                      team runs at_bats hits homeruns bat_avg strikeouts
## 16 Philadelphia Phillies  713    5579 1409      153   0.253       1024
##    stolen_bases wins new_onbase new_slug new_obs
## 16           96  102      0.323    0.395   0.717
```

```
    -2789.2429 + 0.6305 * 5579
```

```
## [1] 728.3166
```

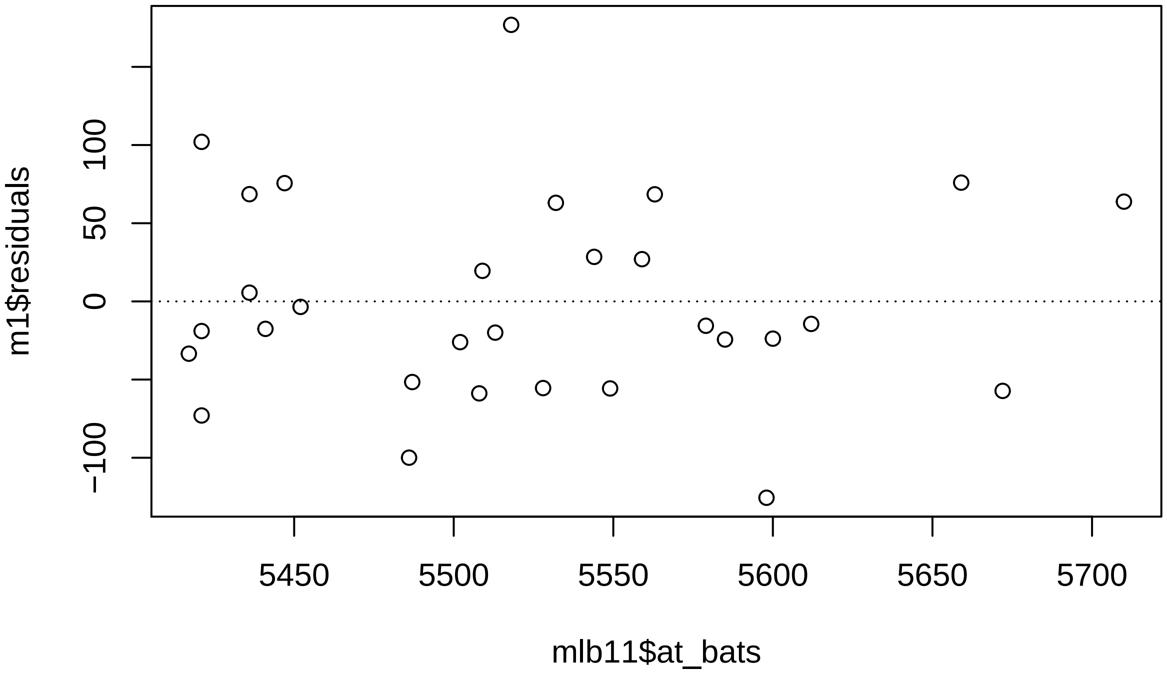
```
##  If the team manager only had the regression line they would predict 728.3166
##  runs for 5,579 at-bats. This would be a slight over-estimation since the observed
##  number of runs was 713. This gives a residual of 15.3166.
```

Model diagnostics

To assess whether the linear model is reliable, we need to check for (1) linearity, (2) nearly normal residuals, and (3) constant variability.

Linearity: You already checked if the relationship between runs and at-bats is linear using a scatterplot. We should also verify this condition with a plot of the residuals vs. at-bats. Recall that any code following a # is intended to be a comment that helps understand the code but is ignored by R.

```
plot(m1$residuals ~ mlb11$at_bats)
abline(h = 0, lty = 3)  # adds a horizontal dashed line at y = 0
```

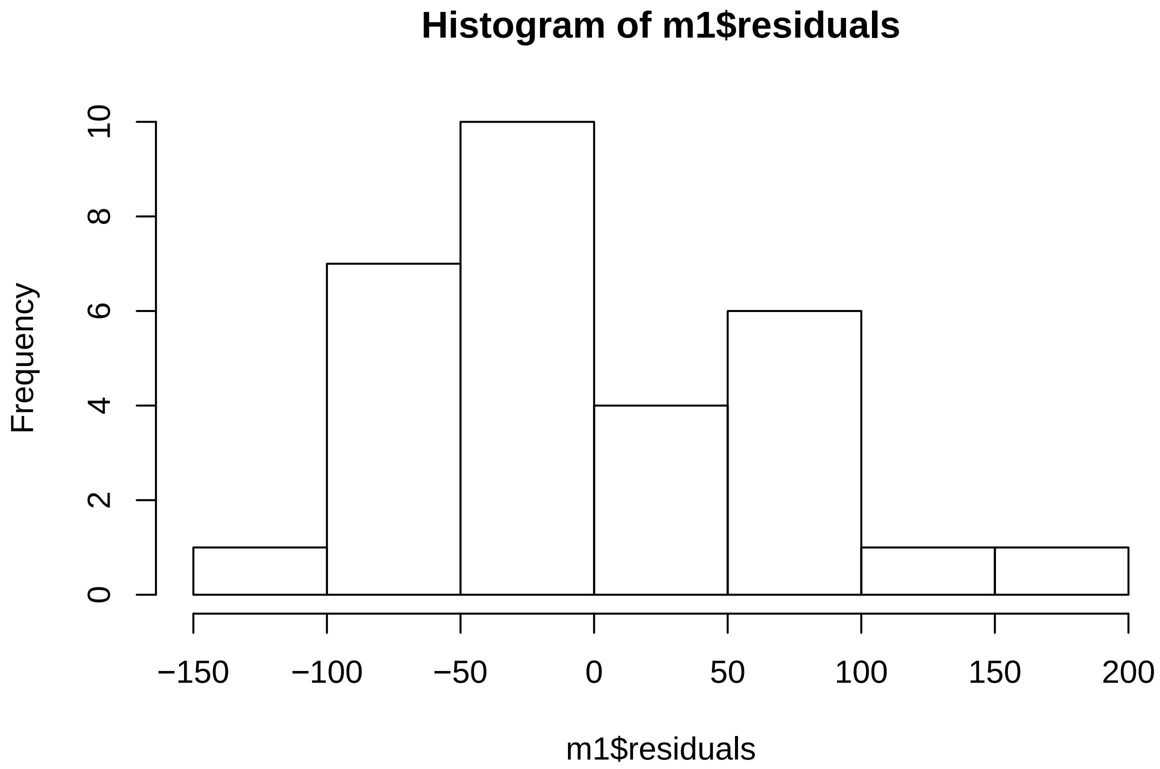


6. Is there any apparent pattern in the residuals plot? What does this indicate about the linearity of the relationship between runs and at-bats?

```
##  Student response to exercise 6

##  No, the residuals are mostly spreadout across the x-axis (at-bats). This
##  supports the position of a linear relationship.
```

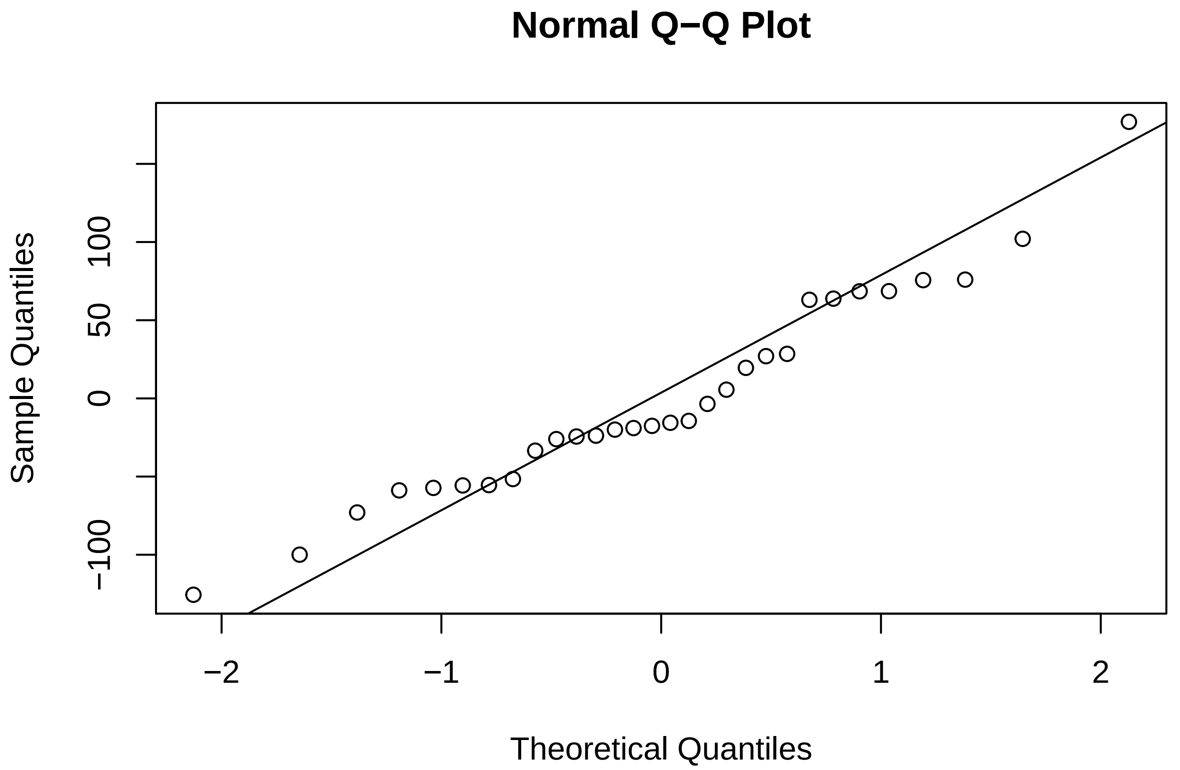
Nearly normal residuals: To check this condition, we can look at a histogram

```
hist(m1$residuals)
```



or a normal probability plot of the residuals.

```
qqnorm(m1$residuals)
qqline(m1$residuals)  # adds diagonal line to the normal prob plot
```



7. Based on the histogram and the normal probability plot, does the nearly normal residuals condition appear to be met?

```
##  Student response to exercise 7

##  Yes, based on the histogram and the normal probability plot, the nearly
##  normal residuals condition appear to have been met.
```

Constant variability:

8. Based on the plot in (1), does the constant variability condition appear to be met?

```
##  Student response to exercise8

##  Yes, based on the residual plot, the variablity around the least squares line
##  remains roughly constant across values of 'at_bats'.
```

On Your Own

- Choose another traditional variable from `mlb11` that you think might be a good predictor of `runs`. Produce a scatterplot of the two variables and fit a linear model. At a glance, does there seem to be a linear relationship?

- How does this relationship compare to the relationship between `runs` and `at_bats`? Use the R^2 values from the two model summaries to compare. Does your variable seem to predict `runs` better than `at_bats`? How can you tell?

- Now that you can summarize the linear relationship between two variables, investigate the relationships between `runs` and each of the other five traditional variables. Which variable best predicts `runs`? Support your conclusion using the graphical and numerical methods we've discussed (for the sake of conciseness, only include output for the best variable, not all five).

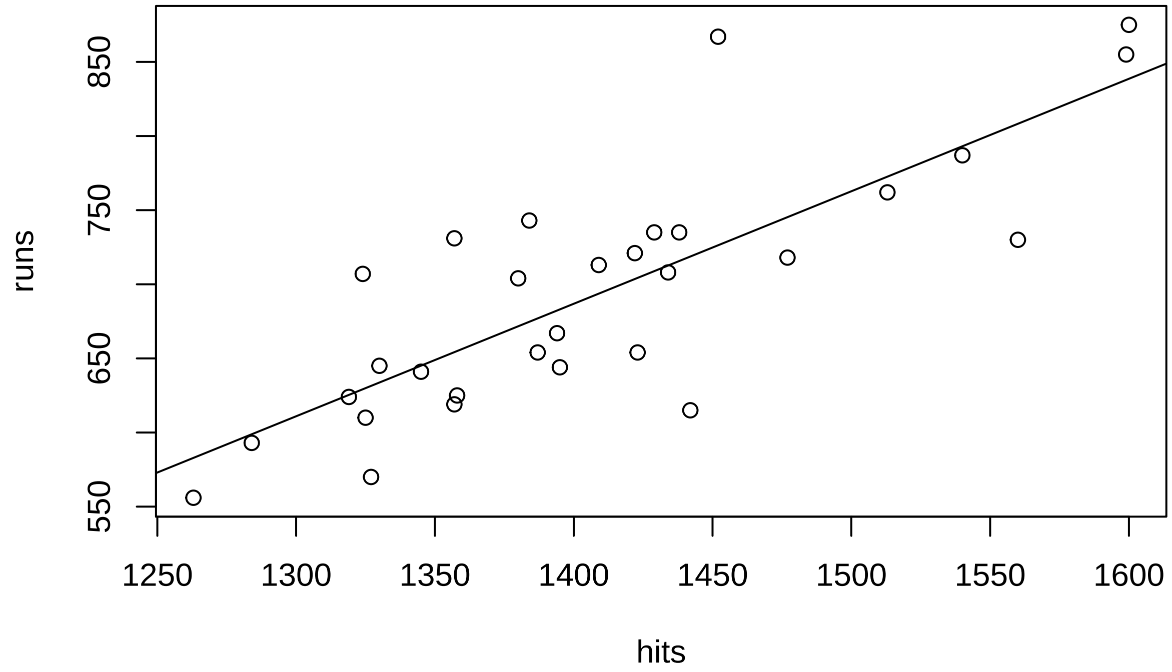
- Now examine the three newer variables. These are the statistics used by the author of *Moneyball* to predict a teams success. In general, are they more or less effective at predicting runs than the old variables? Explain using appropriate graphical and numerical evidence. Of all ten variables we've analyzed, which seems to be the best predictor of `runs`? Using the limited (or not so limited) information you know about these baseball statistics, does your result make sense?

- Check the model diagnostics for the regression model with the variable you decided was the best predictor for runs.

```
##  Student response to On Your Own 1

    head(mlb11)
```

```
##                    team runs at_bats hits homeruns bat_avg strikeouts
## 1        Texas Rangers  855    5659 1599      210   0.283        930
## 2      Boston Red Sox  875    5710 1600      203   0.280       1108
## 3      Detroit Tigers  787    5563 1540      169   0.277       1143
## 4  Kansas City Royals  730    5672 1560      129   0.275       1006
## 5 St. Louis Cardinals  762    5532 1513      162   0.273        978
## 6       New York Mets  718    5600 1477      108   0.264       1085
##   stolen_bases wins new_onbase new_slug new_obs
## 1          143   96      0.340    0.460   0.800
## 2          102   90      0.349    0.461   0.810
## 3           49   95      0.340    0.434   0.773
## 4          153   71      0.329    0.415   0.744
## 5           57   90      0.341    0.425   0.766
## 6          130   77      0.335    0.391   0.725
```

```
    mhits1 <- lm(runs ~ hits, data = mlb11)
    plot(runs ~ hits, data = mlb11)
    abline(mhits1)
```



```
##  The new predictor variable used is 'hits'.
##  Yes, visually there appears to be a positive linear relationship between
##  'hits' and 'runs'.
```

```
##  Student response to On Your Own 2

    summary(m1)
```

```
##
## Call:
## lm(formula = runs ~ at_bats, data = mlb11)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -125.58  -47.05  -16.59   54.40  176.87
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2789.2429   853.6957  -3.267 0.002871 **
## at_bats         0.6305     0.1545   4.080 0.000339 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 66.47 on 28 degrees of freedom
## Multiple R-squared:  0.3729, Adjusted R-squared:  0.3505
## F-statistic: 16.65 on 1 and 28 DF,  p-value: 0.0003388
```

```
    summary(mhits1)
```

```
##
## Call:
## lm(formula = runs ~ hits, data = mlb11)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -103.718  -27.179   -5.233   19.322  140.693
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -375.5600   151.1806  -2.484   0.0192 *
## hits           0.7589     0.1071   7.085 1.04e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 50.23 on 28 degrees of freedom
## Multiple R-squared:  0.6419, Adjusted R-squared:  0.6292
## F-statistic:  50.2 on 1 and 28 DF,  p-value: 1.043e-07
```

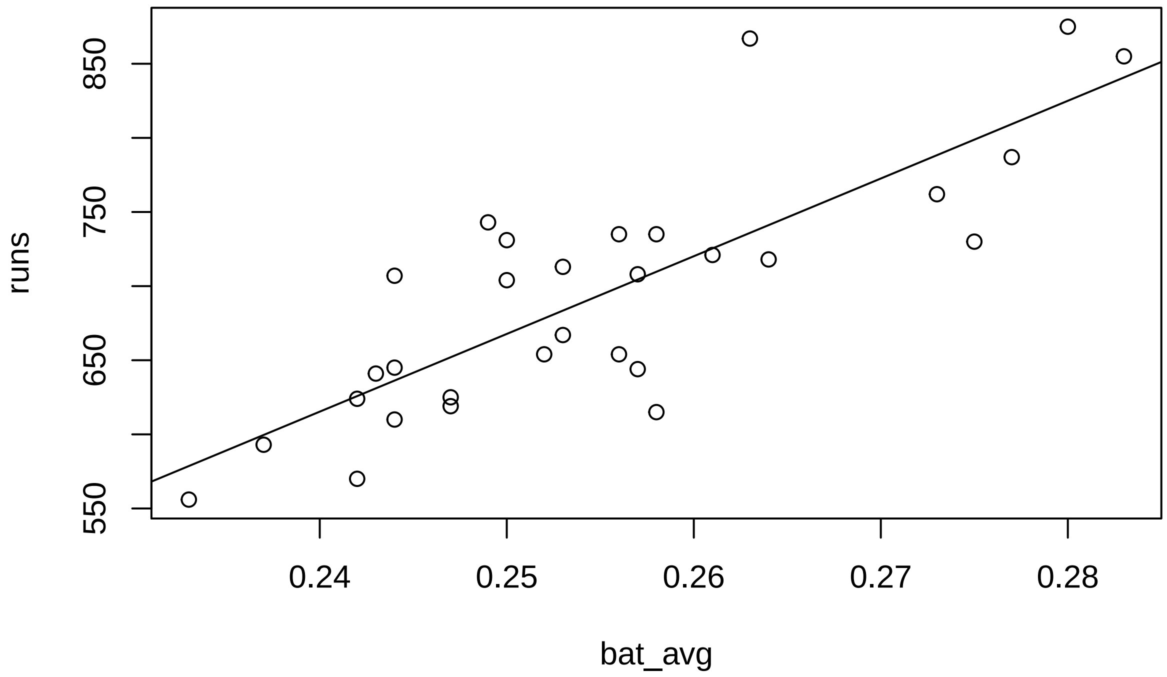
```
##  The variable 'hits' has a strong relationship with 'runs' than was seen with
##  'at_bats' (R^2 values of 0.6419 and 0.3729 respectively). The variable 'hits'
##  seems to be a better predictor of 'runs' than 'at_bats'. This can be seen (1)
##  by the higher value of R^2 and also visually by a tigher clustering of the
##  data around the regression line.
```

```
##  Student response to On Your Own 3

    head(mlb11)
```

```
##                  team runs at_bats hits homeruns bat_avg strikeouts
## 1       Texas Rangers  855    5659 1599      210   0.283        930
## 2      Boston Red Sox  875    5710 1600      203   0.280       1108
## 3      Detroit Tigers  787    5563 1540      169   0.277       1143
## 4  Kansas City Royals  730    5672 1560      129   0.275       1006
## 5 St. Louis Cardinals  762    5532 1513      162   0.273        978
## 6       New York Mets  718    5600 1477      108   0.264       1085
##   stolen_bases wins new_onbase new_slug new_obs
## 1          143   96      0.340    0.460   0.800
## 2          102   90      0.349    0.461   0.810
## 3           49   95      0.340    0.434   0.773
## 4          153   71      0.329    0.415   0.744
## 5           57   90      0.341    0.425   0.766
## 6          130   77      0.335    0.391   0.725
```

```
    ##  mhomeruns1 <- lm(runs ~ homeruns, data = mlb11)
    mbat_avg1 <- lm(runs ~ bat_avg, data = mlb11)
    ##  mstrikeouts1 <- lm(runs ~ strikeouts, data = mlb11)
    ##  mstolen_bases1 <- lm(runs ~ stolen_bases, data = mlb11)
    ##  mwins1 <- lm(runs ~ wins, data = mlb11)
```

```
##  plot(runs ~ at_bats, data = mlb11); abline(m1)
##  plot(runs ~ hits, data = mlb11); abline(mhits1)
##  plot(runs ~ homeruns, data = mlb11); abline(mhomeruns1)
plot(runs ~ bat_avg, data = mlb11); abline(mbat_avg1)
```



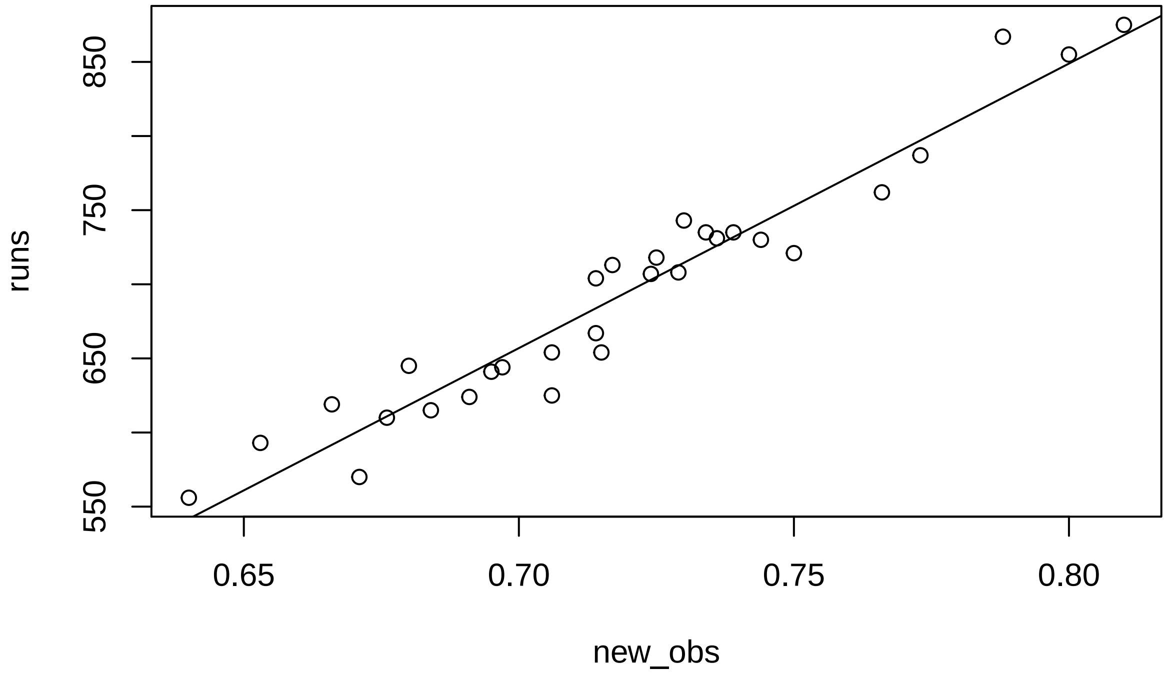
```
##  plot(runs ~ strikeouts, data = mlb11); abline(mstrikeouts1)
##  plot(runs ~ stolen_bases, data = mlb11); abline(mstolen_bases1)
##  plot(runs ~ wins, data = mlb11); abline(mwins1)


##  summary(m1)   ## at_bats
##  summary(mhits1)
##  summary(mhomeruns1)
summary(mbat_avg1)
```

```
##
## Call:
## lm(formula = runs ~ bat_avg, data = mlb11)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -94.676 -26.303  -5.496  28.482 131.113
##
## Coefficients:
```

```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -642.8      183.1  -3.511  0.00153 **
## bat_avg       5242.2      717.3   7.308 5.88e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 49.23 on 28 degrees of freedom
## Multiple R-squared:  0.6561, Adjusted R-squared:  0.6438
## F-statistic: 53.41 on 1 and 28 DF,  p-value: 5.877e-08
```

```
    ##  summary(mstrikeouts1)
    ##  summary(mstolen_bases1)
    ##  summary(mwins1)


    ##  Of the seven traditional variables, 'bat_avg' is the strongest predictor
    ##  of 'runs' for this dataset. This can be seen visually as 'bat_avg' has
    ##  the tighest cluster of data near the regression line in the scatter plot.
    ##  This is also seen by 'bat_avg' having the highest value of R^2 (0.6561).
```

```
##  Student response to On Your Own 4

    ##  head(mlb11)

    ##  mnew_onbase1 <- lm(runs ~ new_onbase, data = mlb11)
    ##  mnew_slug1 <- lm(runs ~ new_slug, data = mlb11)
    mnew_obs1 <- lm(runs ~ new_obs, data = mlb11)


    ##  plot(runs ~ new_onbase, data = mlb11); abline(mnew_onbase1)
    ##  plot(runs ~ new_slug, data = mlb11); abline(mnew_slug1)
    plot(runs ~ new_obs, data = mlb11); abline(mnew_obs1)
```

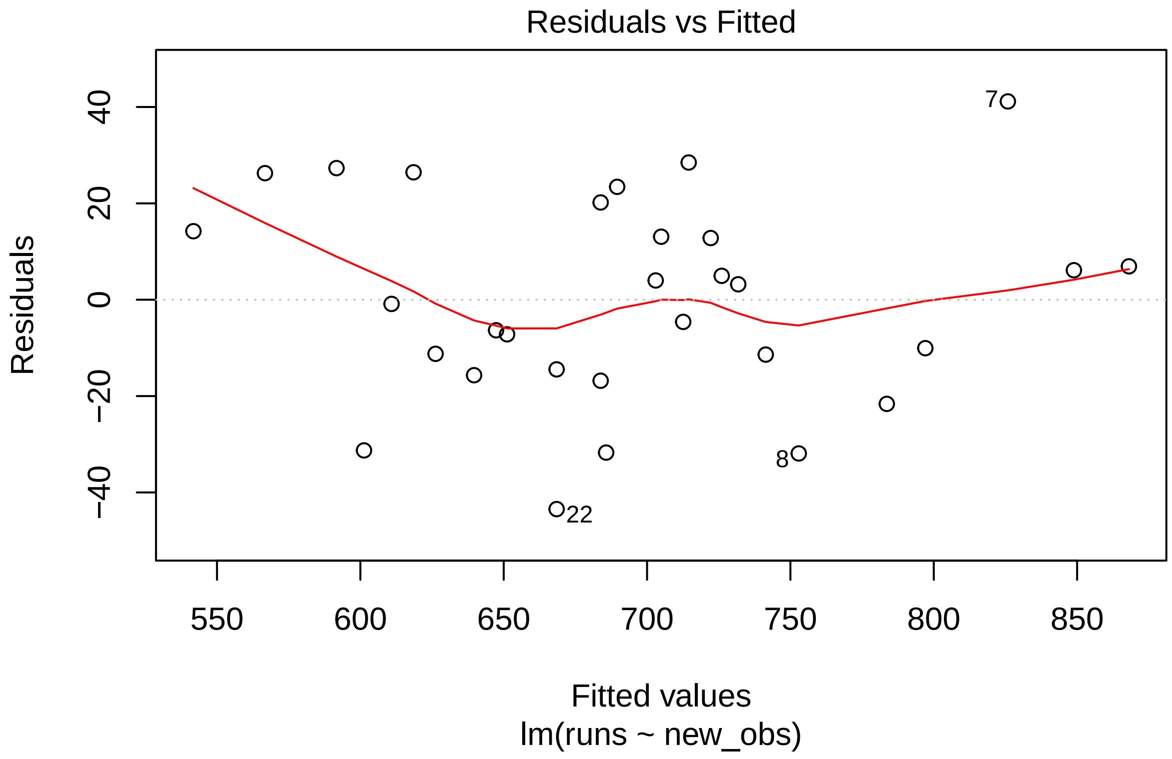


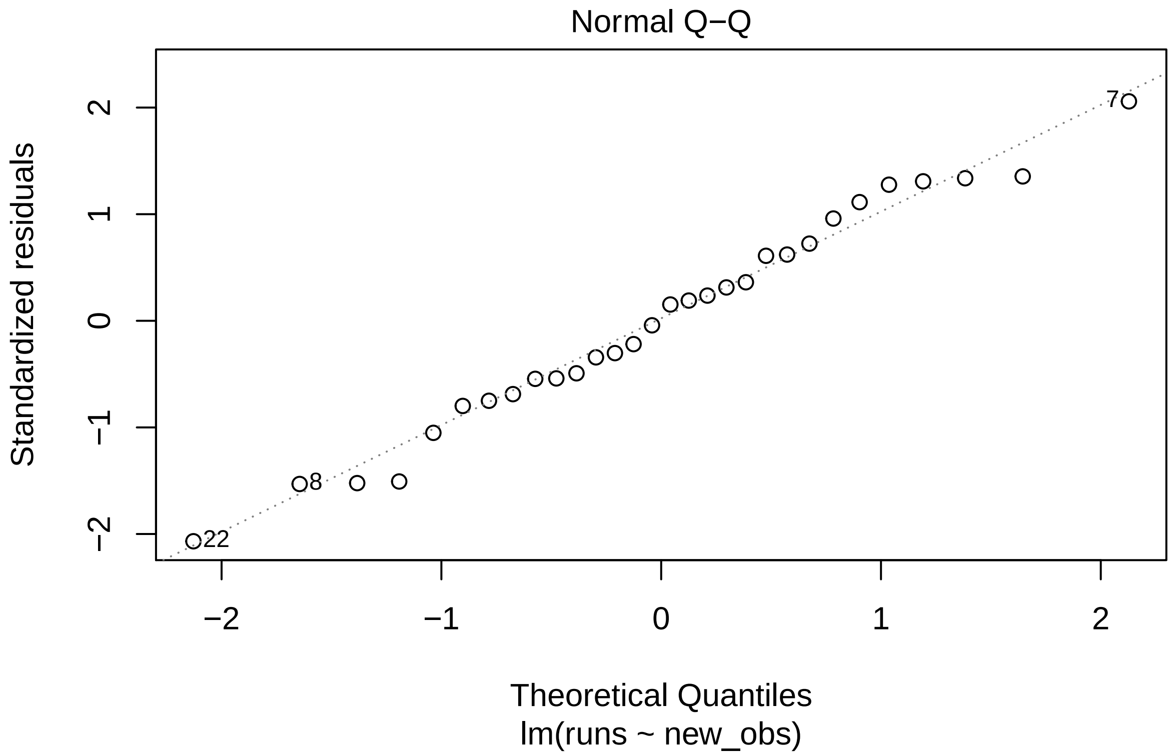
```r
##  summary(mnew_onbase1)
##  summary(mnew_slug1)
summary(mnew_obs1)
```

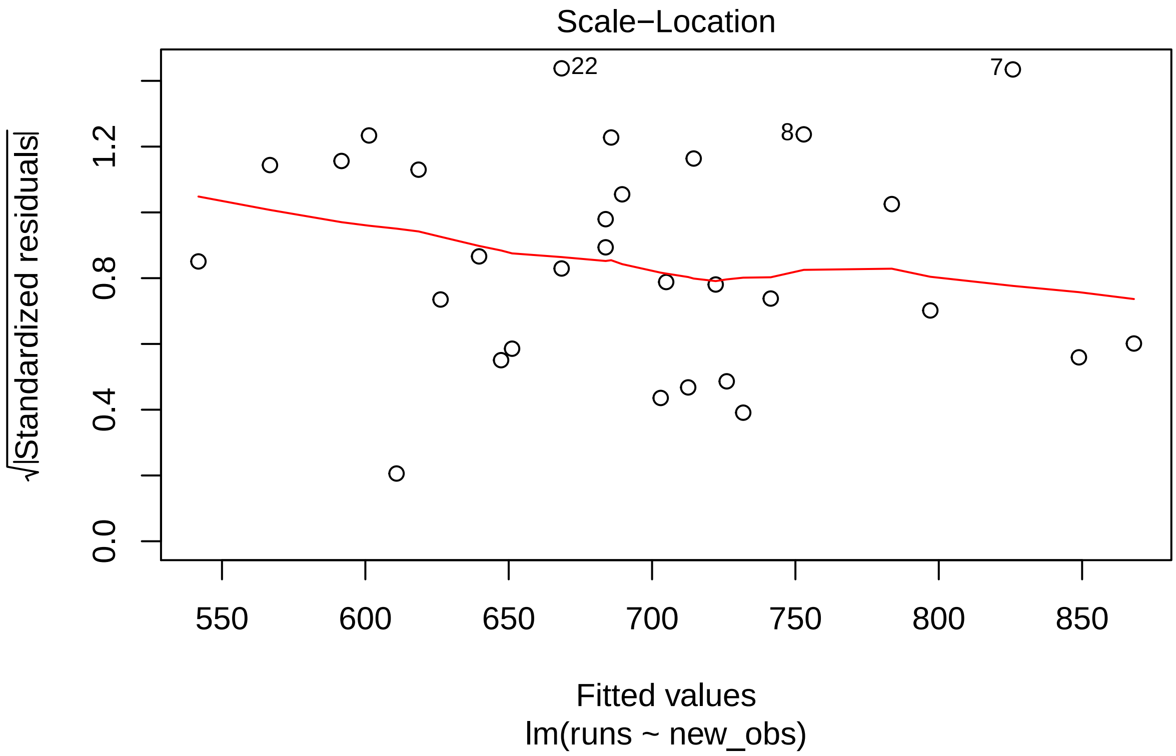
```
##
## Call:
## lm(formula = runs ~ new_obs, data = mlb11)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -43.456 -13.690   1.165  13.935  41.156
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -686.61      68.93  -9.962 1.05e-10 ***
## new_obs      1919.36      95.70  20.057  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 21.41 on 28 degrees of freedom
## Multiple R-squared:  0.9349, Adjusted R-squared:  0.9326
## F-statistic: 402.3 on 1 and 28 DF,  p-value: < 2.2e-16
```

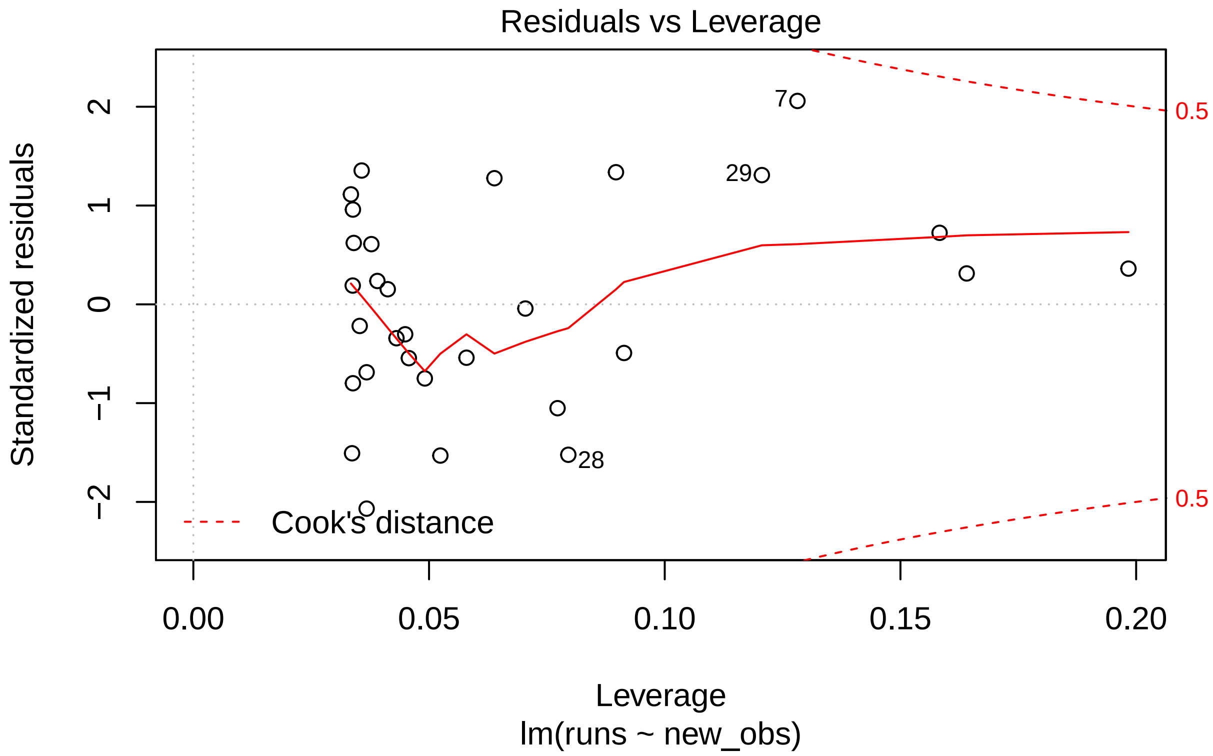
```r
##  Each of the new variables is a better predictor of 'runs' than any of the
##  traditional variables witihn this dataset. Of all ten variables, 'new_obs'
##  is the strongest predictor. This is determined visually as its data is
##  very tightly clustered around the reresssion line. Numerically, this
##  variable has the highest R-squared value at 0.9349.

##  Given a quick search for the terms "baseball statistics obs" returns that
##  this variable is on-base plus slugging (which measure the ability to get
##  on base and hitting power). With this information, yes, it makes sense that
##  this would be a very powerful predictor of 'runs'.
```

```
##  Student response to On Your Own 5

    plot(mnew_obs1)
```




Normal Q−Q
Standardized residuals
2
1
0
−1
−2
−2
−1
0
1
2
Theoretical Quantiles
lm(runs ~ new_obs)
7
8
22


Scale−Location
√|Standardized residuals|
0.0
0.4
0.8
1.2
550
600
650
700
750
800
850
22
8
7
Fitted values
lm(runs ~ new_obs)



```
##  Linearity:  Residuals vs Fitted; the data appear to have a mostly linear
##              relationship

##  Nearly normal residuals:  Normal Q-Q; the data appear to be tighly
##                            clustered around the normal line, in most cases

##  Constant variability: Scale-Location; the data appear mostly evenly spread
##                        across the x-axis and the regression line is nearly
##                        horizontal (which is desired).

##  Independent observations: Residuals vs Leverage; none of the variables
##                            meet the thresholds for "influential values"
##                            which indicates observations are likely
##                            independent.
```