Moneyball - Linear Regression Continued

Part 1

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, better predict 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 exercise 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

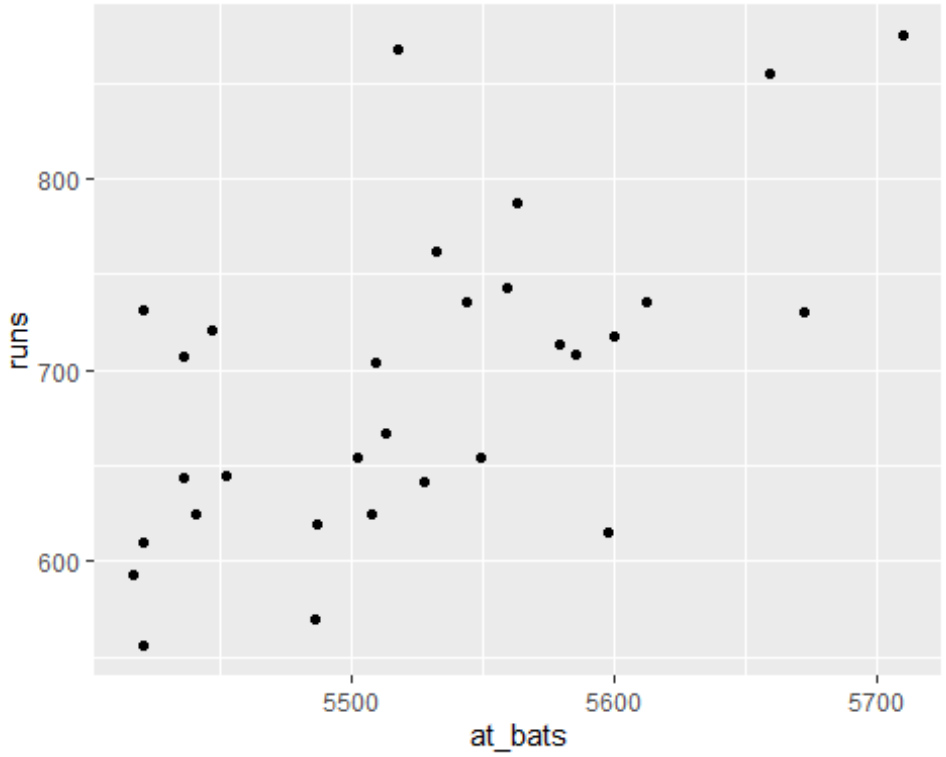
Let's load up the data for the 2011 season (and load up `mosaic` while we're at it!).

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? Using the variable `at_bats`, create such a plot between `runs` and `at_bats`. 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?

SOLUTION:

```
graph1 <- ggplot(data=mlb11, aes(x=at_bats, y=runs)) + geom_point()
graph1
```



Scatter plot, since it would show if there is a trend in the data relationship and also shows the outliers. The relation between them is linear based on the scatter plot. As the relation looks linear, we can use at_bats to predict the number of runs.

2. If the relationship looks linear, we can quantify the strength of the relationship with the correlation coefficient. Find the correlation coefficient between `runs` and `at_bats`.

SOLUTION:

```
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 between two quantitative variables, such as `runs` and `at_bats` above.

3. Looking at your plot from the previous exercise, describe the relationship between `runs` and `at_bats`. Make sure to discuss the form, direction, and strength of the relationship as well as any unusual observations.

SOLUTION: The relationship between runs and at_bats is positive and linear. The strength of their relationship is moderate to strong. There are some outliers on this plot but it is not an unusual observation.

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 represents 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)
```

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. What are residuals?

The most common way to do linear regression is to select the **line that minimizes the sum of squared residuals**.

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 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)
sum(residuals(m1)^2)
```

4. With this table, what is the least squares regression line?

SOLUTION:

runs = −2789.2429 + 0.6305 ∗ atbats

5. 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?

SOLUTION:

```
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
```

runs = 415.2389 + 1.8345 ∗ homeruns This tells us that 1.8345 runs increase for every 1 homerun increase. It shows that that the more home runs the team hits the better. As it equivalents to more runs which is an important factor for a team to succeed.

Prediction and prediction errors

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

```
xyplot(runs ~ at_bats, data=mlb11, type=c("p", "r"))
```

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.

6. 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? (Hint: Calculate the residual for the point)

SOLUTION:

```
atbats <- 5579
yPredicted <- -2789.2429 + 0.6305 * atbats
yPredicted

## [1] 728.3166

observed_df = filter(mlb11, at_bats == 5579)
yObserved = observed_df$runs
residual = yObserved - yPredicted
residual

## [1] -15.3166
```

The prediction would be 728.3166 runs for 5579 at-bats. This is an overestimate by -15.3166.

Model Diagnostics

To assess whether the linear model is reliable, we need to check for Linearity, Independence, Normal errors, and Equal Variance.

- **Linearity**: You already checked if the relationship between runs and at-bats is linear using a scatterplot.
- **Equal Variance**: We want look at a plot of the residuals against the fitted values. If we see a change in the spread of the residuals for larger values of the fitted values, we would be uncomfortable with using a linear regression model as is.

```
xyplot(resid(m1) ~ fitted(m1), data=mlb11, type=c("p", "r"))
```

7. Based on this, does the equal variance condition appear to be met?

SOLUTION: As the spread of the residuals is not in any pattern and are roughly consistent across all levels, we can say that the equal variance condition is met.

- **Independence**: We don't really have independence here, since each observation in the dataset reflects on how a particular team did when playing against the other teams. We'll overlook this bit.
- **Normal errors**: To check this condition, we can look at a histogram of the residuals or a normal quantile plot:

```
histogram(~residuals(m1), width=50)

qqmath(~resid(m1))
ladd(panel.qqmathline(resid(m1)))
```

8. Do the residuals appear normally distributed?

SOLUTION: The residuals seem to be normally distributed though there are a few small deviations from the line.

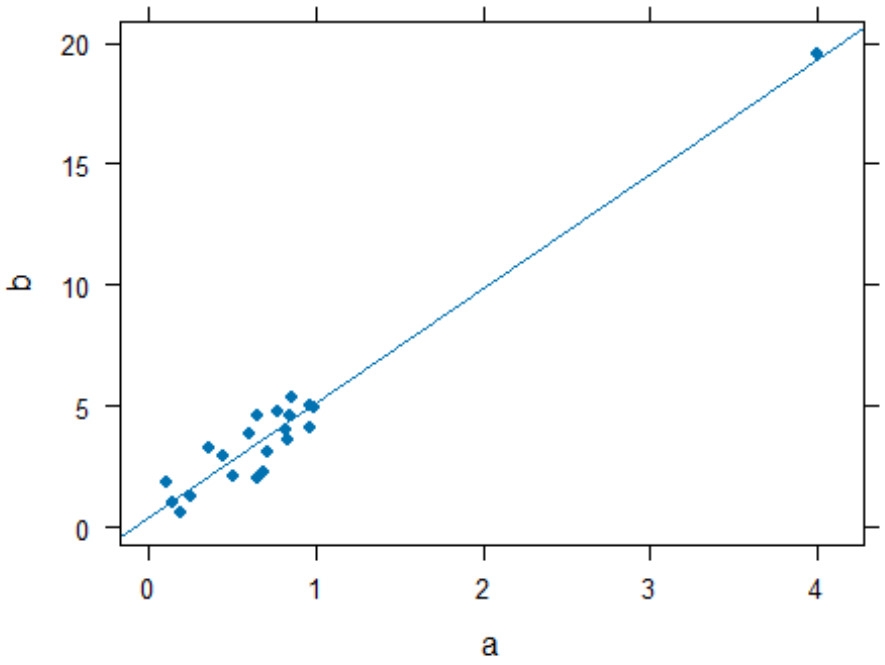
9. Do there appear to be any outliers?

SOLUTION: There is an outlier when the qnorm is 2 and when it is less than -2.

Unusual/Outlying points such as this can be:

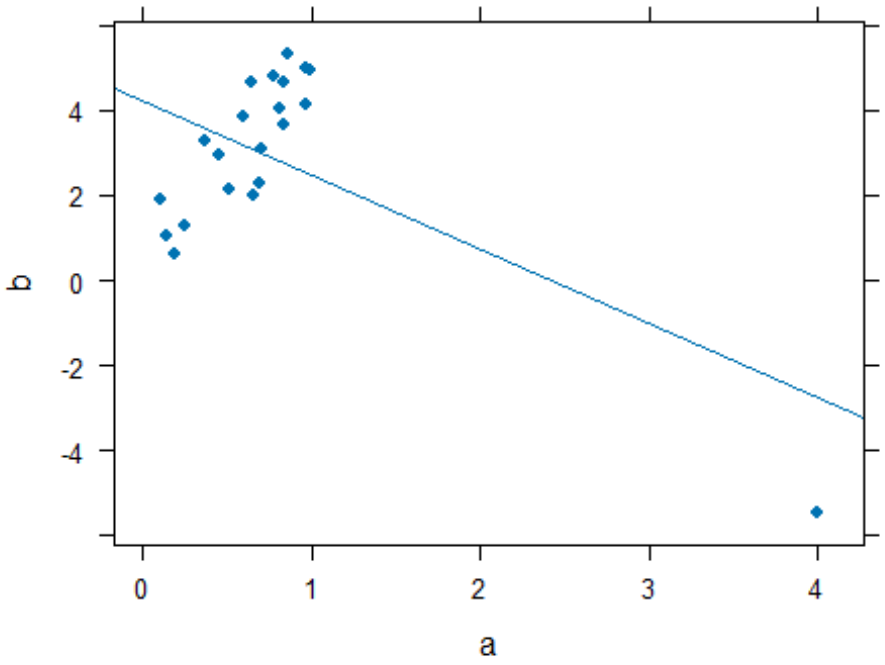
- high leverage: the observation is far away from the other points in the x-direction.

```
# Example for high leverage, high influence and neither
set.seed(15)
a <- c(runif(20), 4)
b <- a*5+rnorm(21)
xyplot(b~a, pch=16, type=c("p", "r"))
```



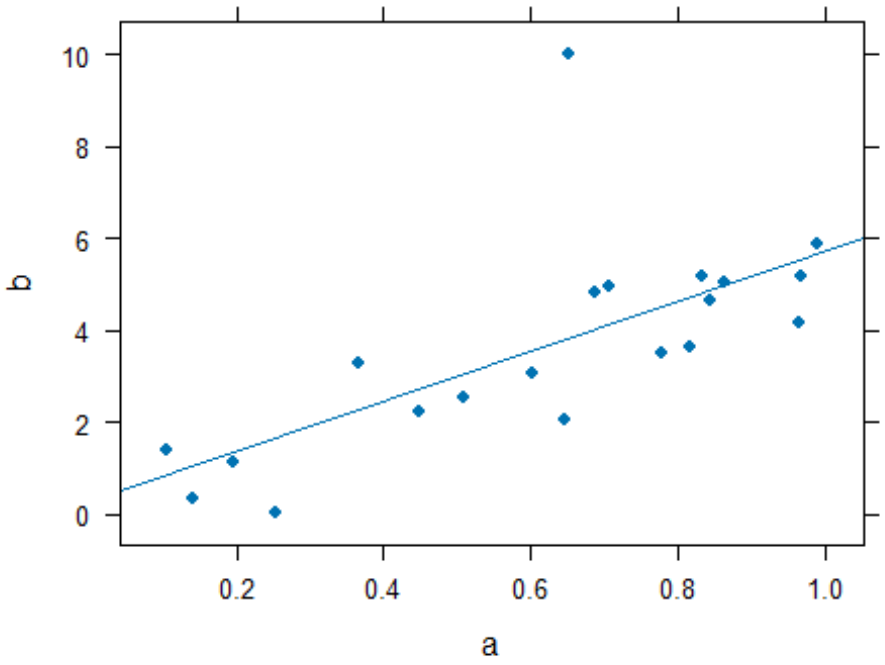
- high influence: the observation, when included, will significantly move the regression line.

```
# Example for high leverage, high influence and neither
b[21] <- b[21] -25
xyplot(b~a, pch=16, type=c("p", "r"))
```



- neither: the observation appears to be really far away from the rest of the observations, but due to its position, is unlikely to move the regression line substantially.

```
# Example for high leverage, high influence and neither
set.seed(15)
a <- sort(runif(20))
b <- a*5 + rnorm(20)
b[10] <- 10
xyplot(b~a, pch=16, type=c("p", "r"))
```



10. Is the point that you've identified in the question above (Question 9) high leverage, high influence, or both?

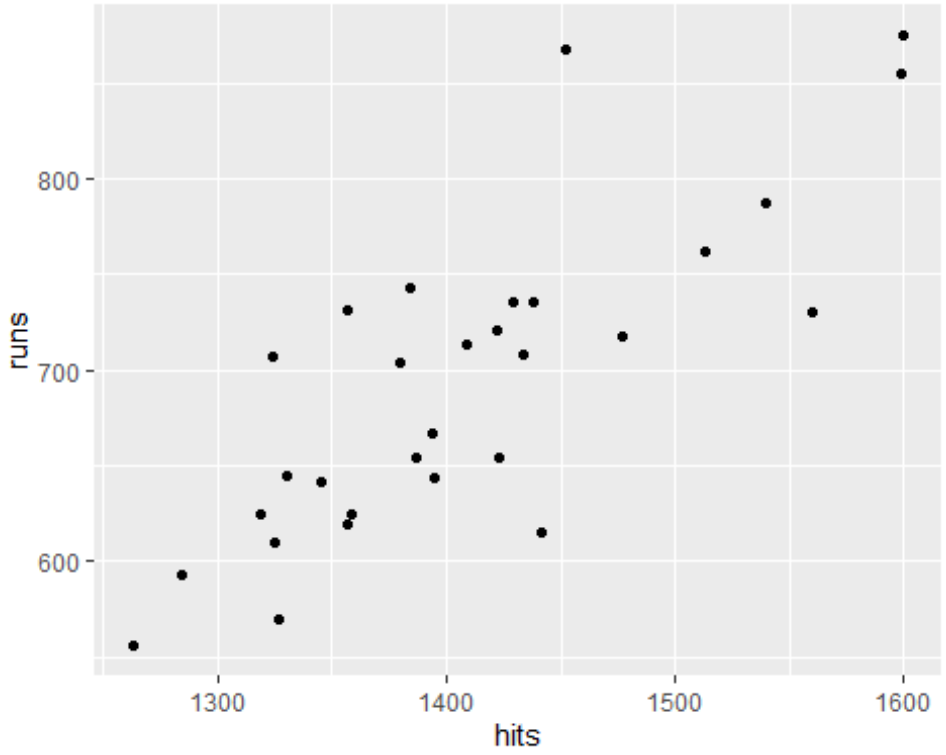
SOLUTION: The plot in question 9 has a high leverage outliers since the outliers are far away from the other x-axis points in comparision.

Part 2

1. 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?

SOLUTION:

```
graph3 = ggplot(data=mlb11, aes(x=hits, y=runs)) + geom_point()
graph3
```



This seems to be a linear relationship between hits and runs.

2. 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?

SOLUTION:

```
m1 = lm(runs ~ at_bats,      data = mlb11)
m3 = lm(runs ~ hits, data = mlb11)

rSquaredAtBats  = summary(m1)$r.squared
rSquaredHits = summary(m3)$r.squared

rSquaredAtBats

## [1] 0.3728654

rSquaredHits

## [1] 0.6419388
```

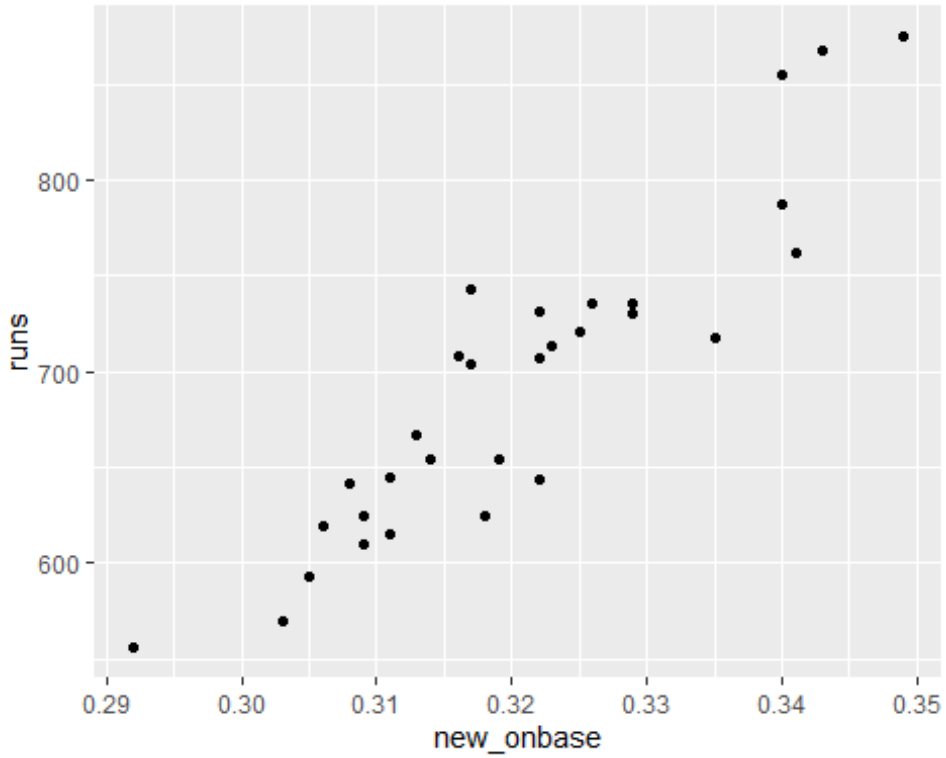
The R^2 value of at_bats is 0.373 and the R^2 value of hits is 0.642. Since the R^2 of hits is way higher than at_bats, it is a better predictor than at_bats.

3. Now examine the three newer variables. These are the statistics used by the author of *Moneyball* to predict a team's success. In general, are they more or less effective at predicting runs that the old variables? Explain using appropriate graphical and

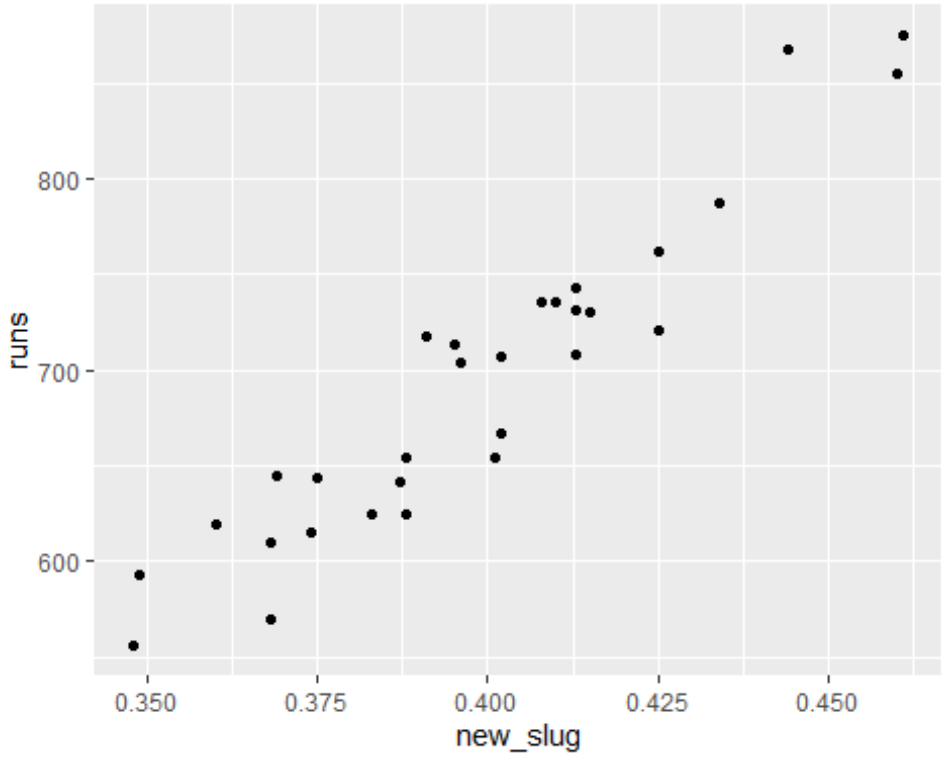
numerical evidence. Of all ten variables we've analyzed, which seems to be the best predictor of `runs`? Using the limited (or not so limited) information you know about these baseball statistics, does your result make sense?

SOLUTION:

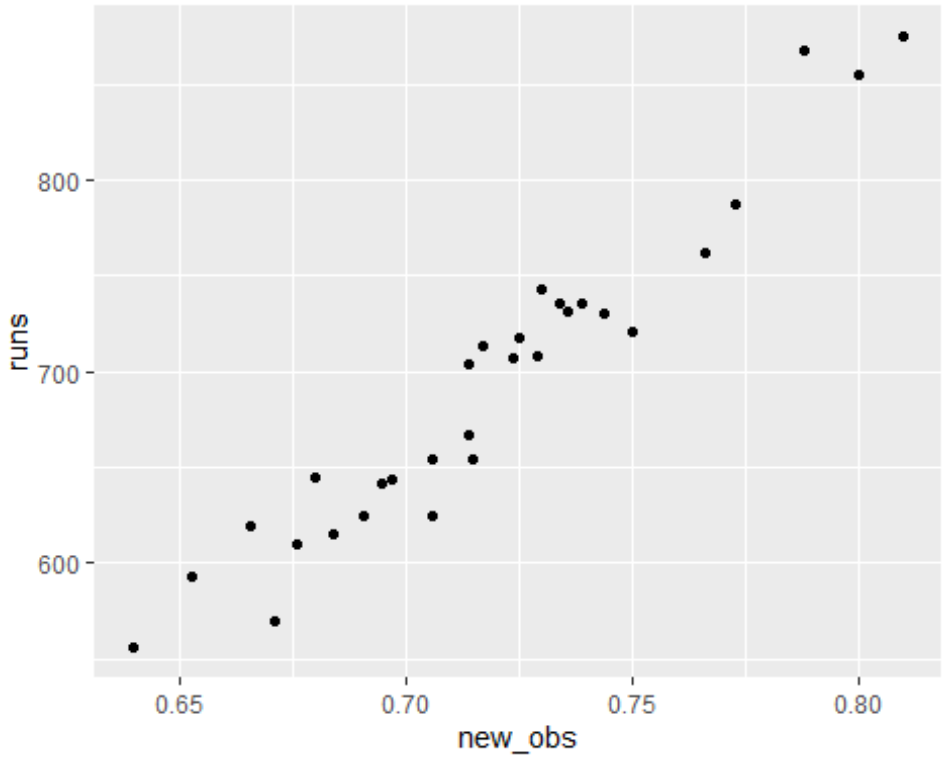
```
graph8 = ggplot(data=mlb11, aes(x=new_onbase, y=runs)) + geom_point()
graph8
```




```
graph9 = ggplot(data=mlb11, aes(x=new_slug, y=runs)) + geom_point()
graph9
```




```
graph10 = ggplot(data=mlb11, aes(x=new_obs, y=runs)) + geom_point()
graph10
```



R-squared values:

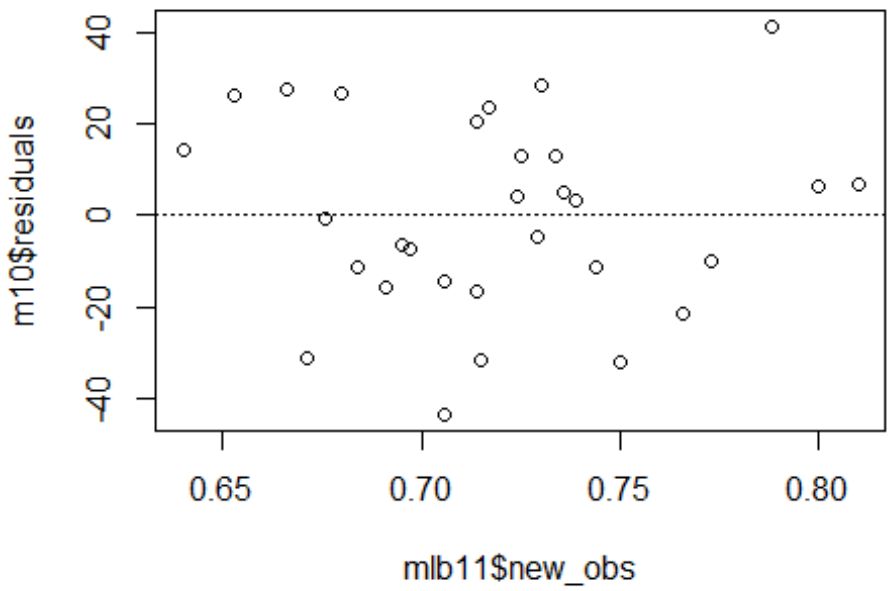
```
m1  = lm(runs ~ at_bats,         data = mlb11)
m2  = lm(runs ~ hits,            data = mlb11)
m3  = lm(runs ~ homeruns,        data = mlb11)
m4  = lm(runs ~ bat_avg,         data = mlb11)
m5  = lm(runs ~ strikeouts,      data = mlb11)
m6  = lm(runs ~ stolen_bases,    data = mlb11)
m7  = lm(runs ~ wins,            data = mlb11)
m8  = lm(runs ~ new_onbase,      data = mlb11)
m9  = lm(runs ~ new_slug,        data = mlb11)
m10 = lm(runs ~ new_obs,         data = mlb11)

rSquared1  = summary(m1)$r.squared
rSquared2  = summary(m2)$r.squared
rSquared3  = summary(m3)$r.squared
rSquared4  = summary(m4)$r.squared
rSquared5  = summary(m5)$r.squared
rSquared6  = summary(m6)$r.squared
rSquared7  = summary(m7)$r.squared
rSquared8  = summary(m8)$r.squared
rSquared9  = summary(m9)$r.squared
rSquared10 = summary(m10)$r.squared

r_squared = c(rSquared1, rSquared2, rSquared3, rSquared4, rSquared5,
rSquared6, rSquared7, rSquared8, rSquared9, rSquared10)
r_squared

##  [1] 0.372865390 0.641938767 0.626563570 0.656077135 0.169357932
0.002913993
##  [7] 0.360971179 0.849105251 0.896870368 0.934927126
```

From the scatter plot graphs of the new variables, we can see all of them show a positive and linear relationship with runs. The new variables (new_onbase, new_slug, new_obs) are way more efficient to use than the old variables since their R-squared values are way higher than the old variables. They can be strong predictors for runs, as new_obs has the highest r-squared value, new_obs should be the best predictor.

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

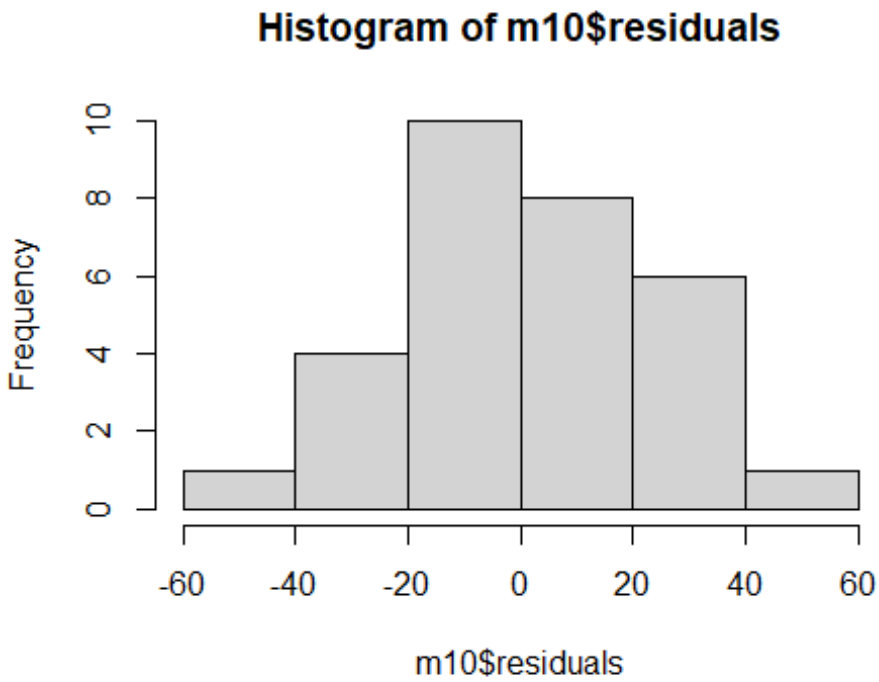
SOLUTION:

```
m10 = lm(runs ~ new_obs,         data = mlb11)
plot(m10$residuals ~ mlb11$new_obs)
abline(h = 0, lty = 3)
```

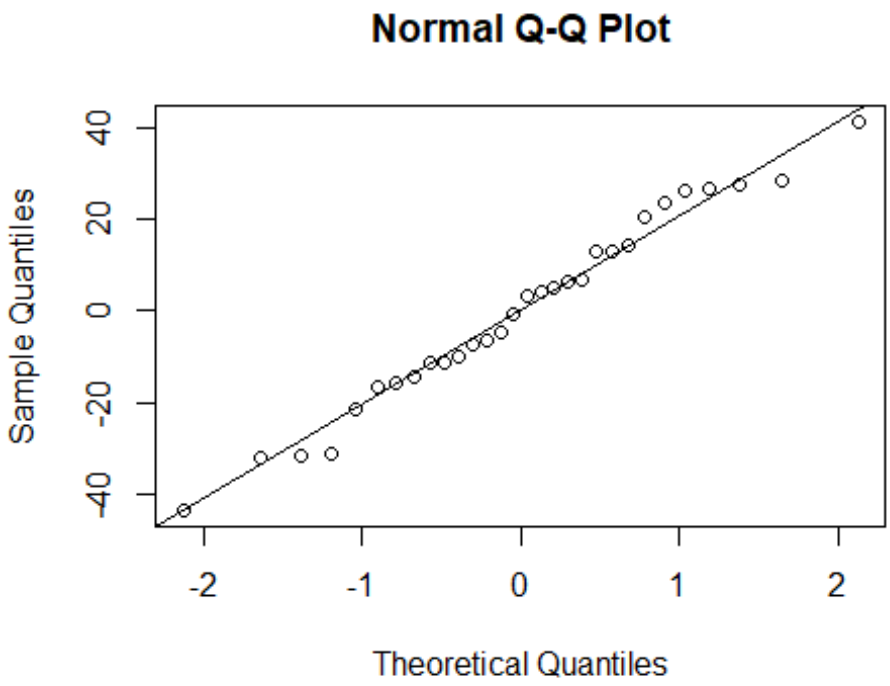


This shows equal variance as there is no pattern and all of the residuals are consistently spread on the plot.

```
hist(m10$residuals)
```



```
qqnorm(m10$residuals)
qqline(m10$residuals)
```



From the histogram and the q-q plot, we can say that there is a normal distribution of the points.

So, choosing new_obs as the best predictor makes sense as it it has a very strong relation to runs.