Introduction to linear regression

Student Name : Sachid Deshmukh

Date : 12/05/2018

- GitHub Location for rmd file
- GitHub Location for pdf file
- RPubs location of published file

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

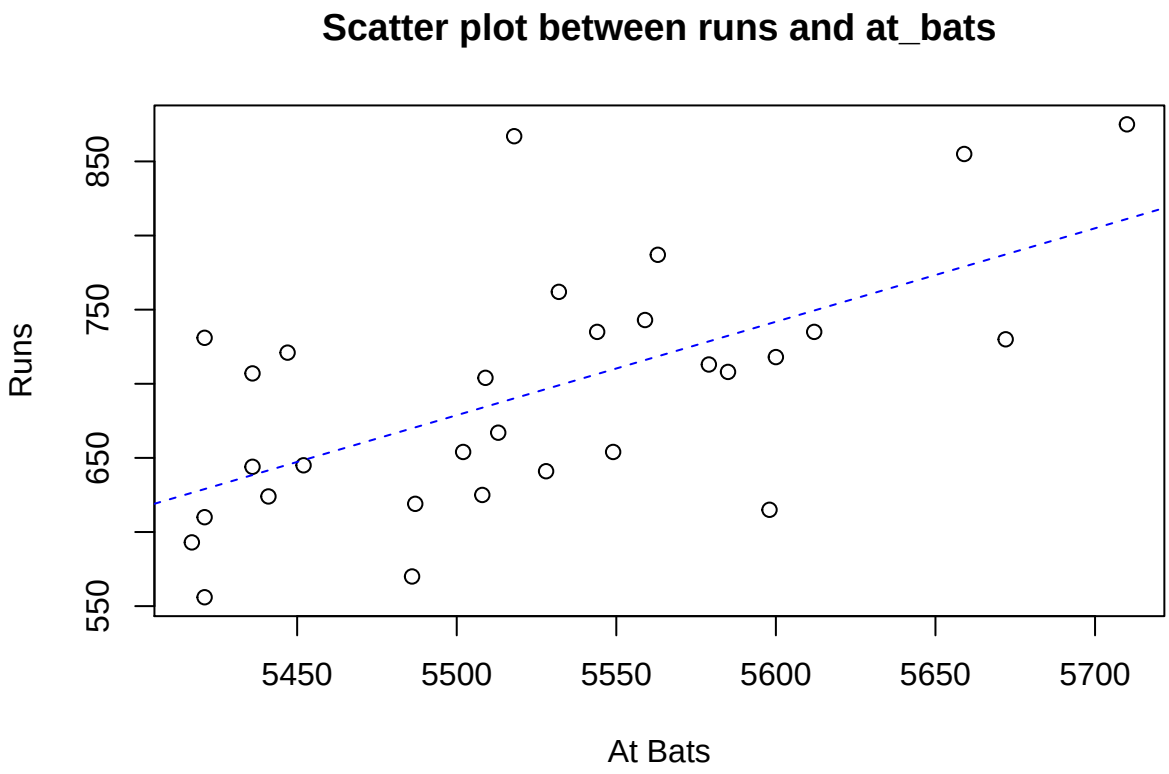
Let's load up the data for the 2011 season.

```
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?

```
fit = lm(mlb11$runs~mlb11$at_bats)
plot(mlb11$at_bats, mlb11$runs, xlab="At Bats", ylab="Runs", main="Scatter plot between runs and at_bats
abline(coefficients(fit), lty=2, col="blue")
```



Answer: We can use scatterplot to identify relationship between two numerical variables From the scatter plot above between runs and at_bats variables, it can be seen that relationship between them is linear. Since linear model assumes linear relationship between predictor and outcome variable, it would be ideal to predict team's runs based on at_bats variable using linear model

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.

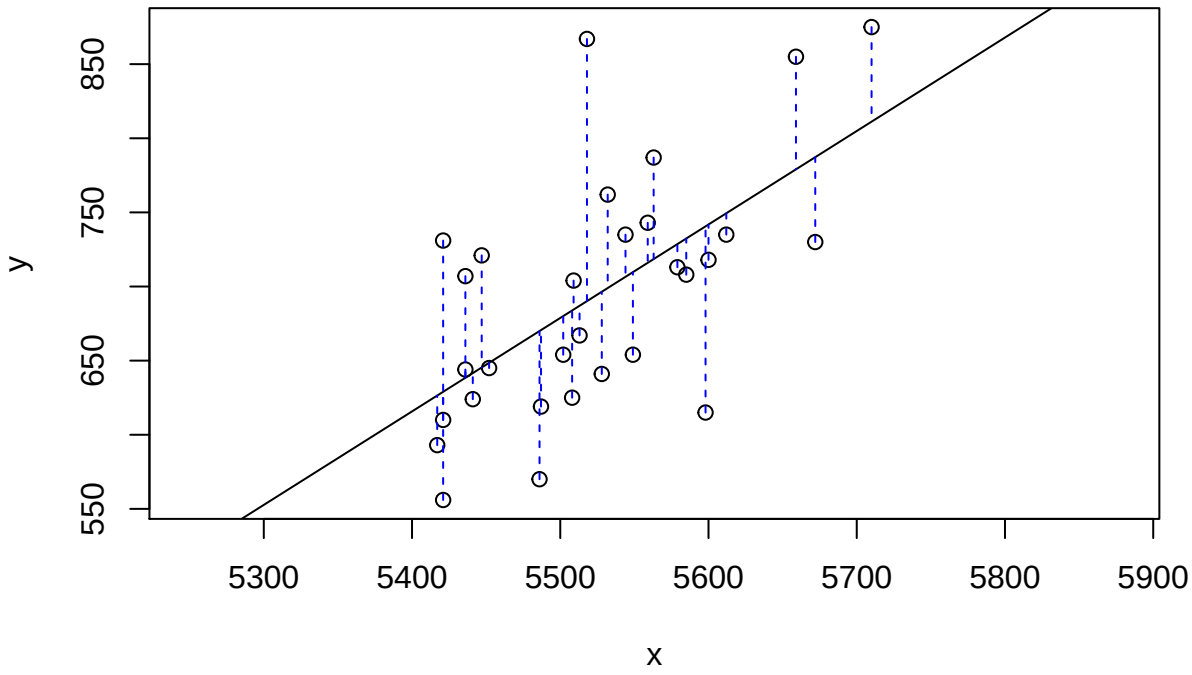
2. Looking at your plot from the previous exercise, describe the relationship between these two variables. Make sure to discuss the form, direction, and strength of the relationship as well as any unusual observations.

From the scatter plot above, we can conclude that there is a linear relationship between runs and at_bats variables. It can be seen that at_bats increases as runs increases which suggests a positive correlation between at_bats and runs. From the correlation function above, it can be seen that at_bats and runs are correlated positively and strength of the correlation is 0.61. Scatter plot above also shows some unusual values for at_bats (very low) when runs are in the range of 700-750. This is an indication of high leverage points in the data. There is one point

in particular where at_bats is unusually high for runs = 850 indicating a possible outlier in the data

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)
```



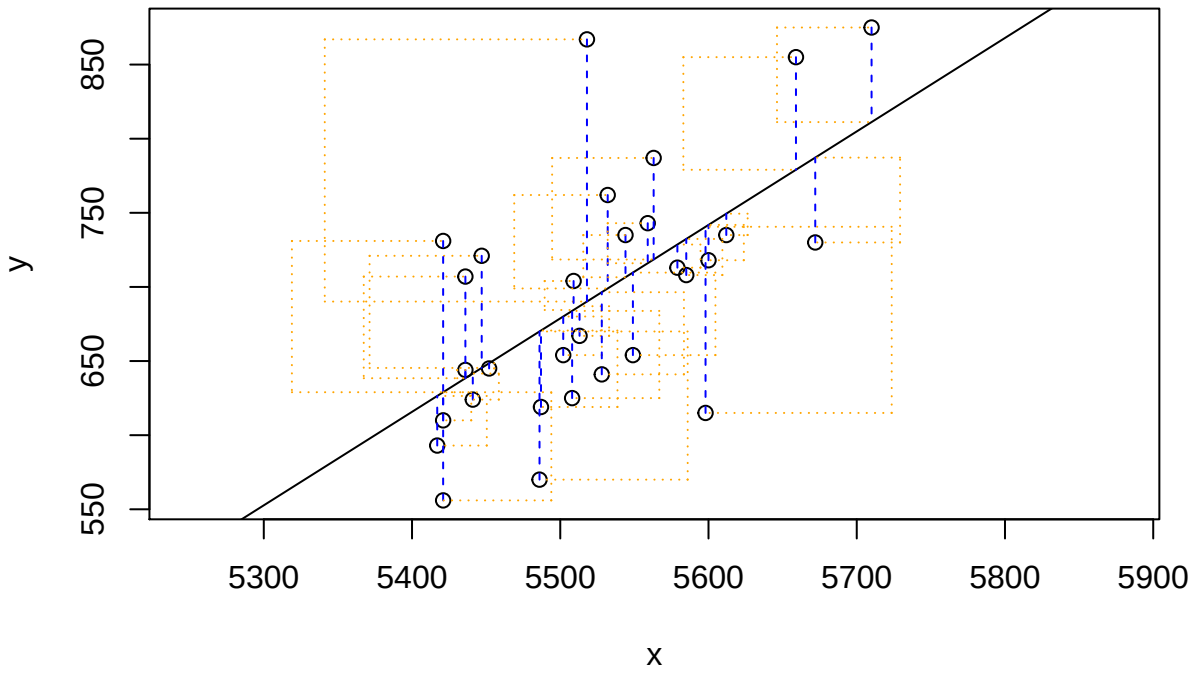

```
## Click two points to make a line.

## 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?

Answer : The smallest sum of squares I got is 133555.9

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?

```
fit = lm(runs~homeruns, data=mlb11)
summary(fit)
```

```
##
## Call:
## lm(formula = runs ~ homeruns, data = mlb11)
```

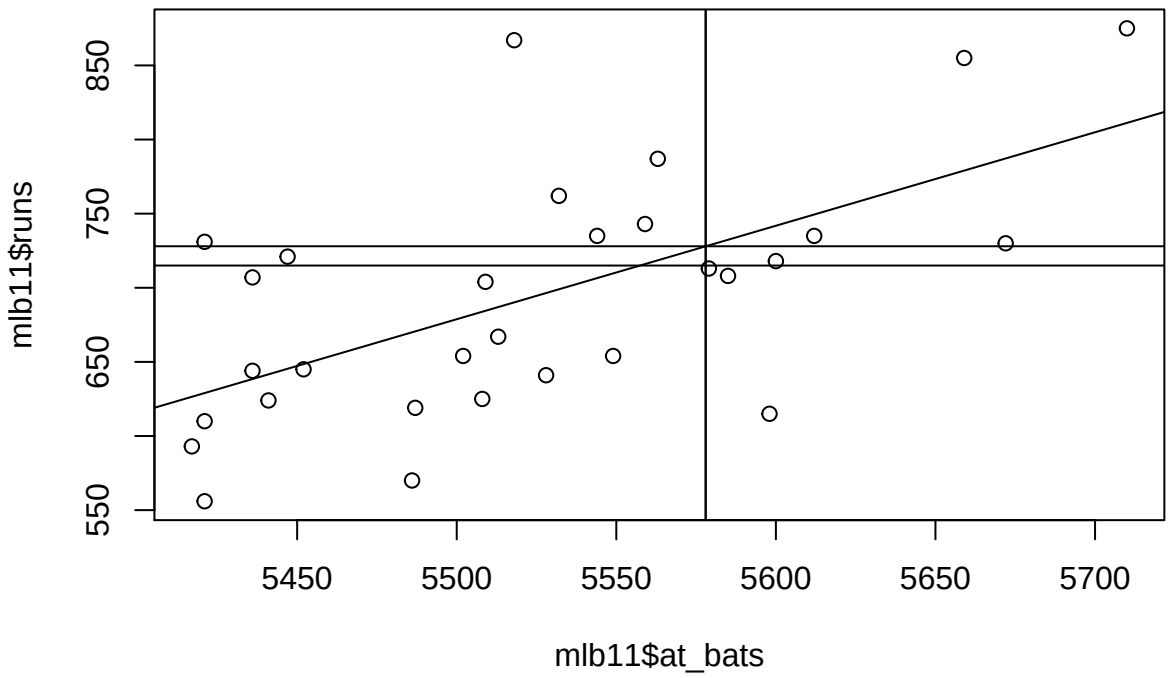
```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -91.615 -33.410   3.231  24.292 104.631
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 415.2389    41.6779   9.963 1.04e-10 ***
## homeruns      1.8345     0.2677   6.854 1.90e-07 ***
## ---
## 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
```

Answer: Equation of the regression line: runs = 415.23 + 1.8345 * homeruns Slope is positive indicating positive relationship between success of a team and its home runs. It can be concluded that each new home run results into 1.8345 more runs for a given team

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)
abline(0,0,728,5578)
abline(0,0,715,5578)
```



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 5,578 at-bats? Is this an overestimate or an underestimate, and by how much? In other words, what is the residual for this prediction?

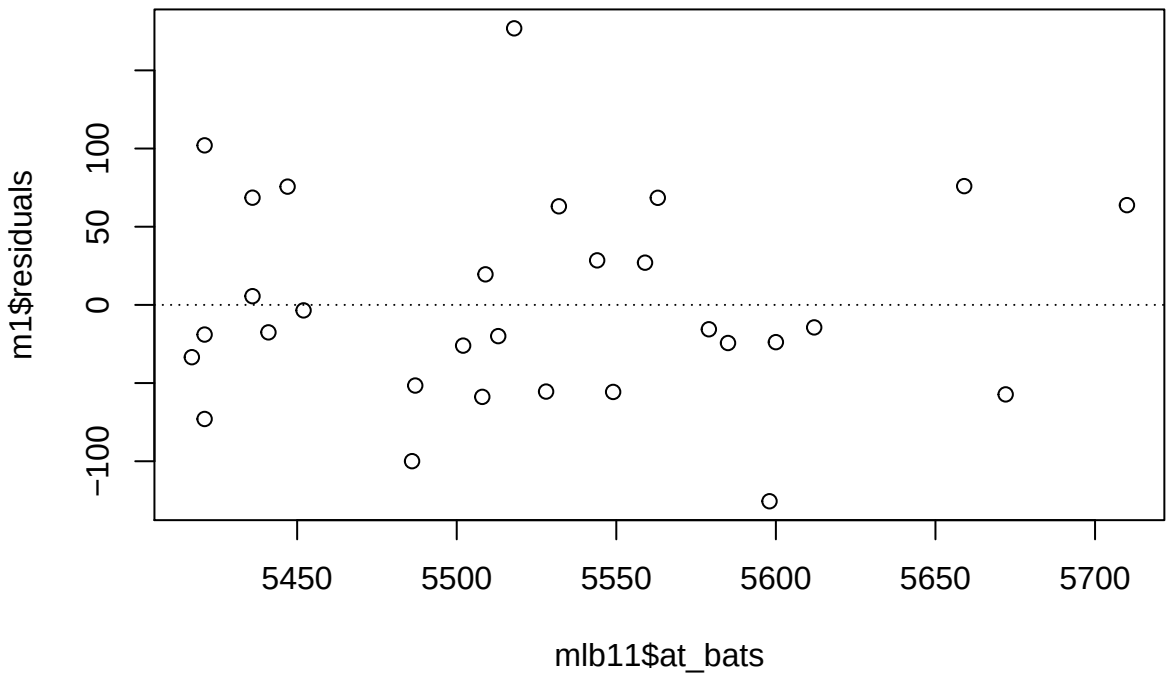
Answer: From the least square plot above, we can see that team manager will predict 728 runs for a team with 5,578 bats. From the same plot we can see that actual value is 715 runs for a team with 5,578 bats. This is an example of over estimation of team runs by a team manager. The magnitude of over estimation is 728 - 715 = 13 runs. The residual for this prediction will be 13

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
```

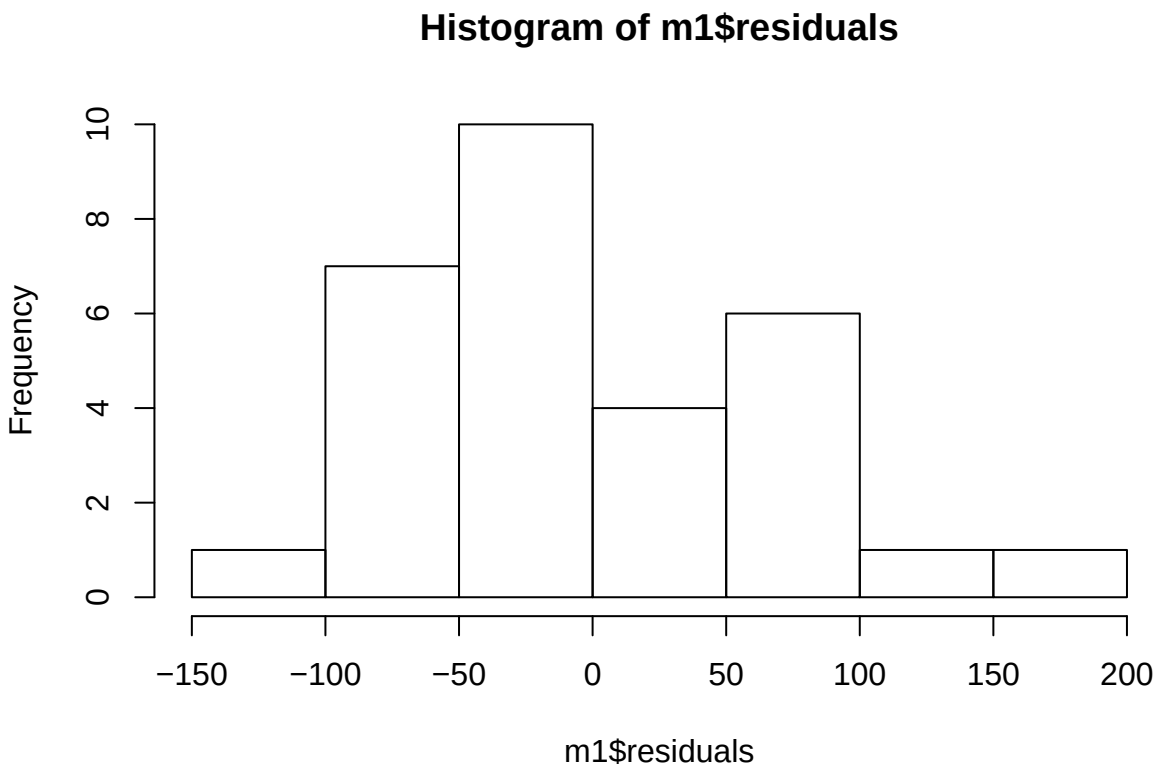


6. Is there any apparent pattern in the residuals plot? What does this indicate about the linearity of the relationship between runs and at-bats?

Answer: There is no specific pattern exhibited in the above residual plots. This indicates that there exists a linear relationship between runs and at_bats variables

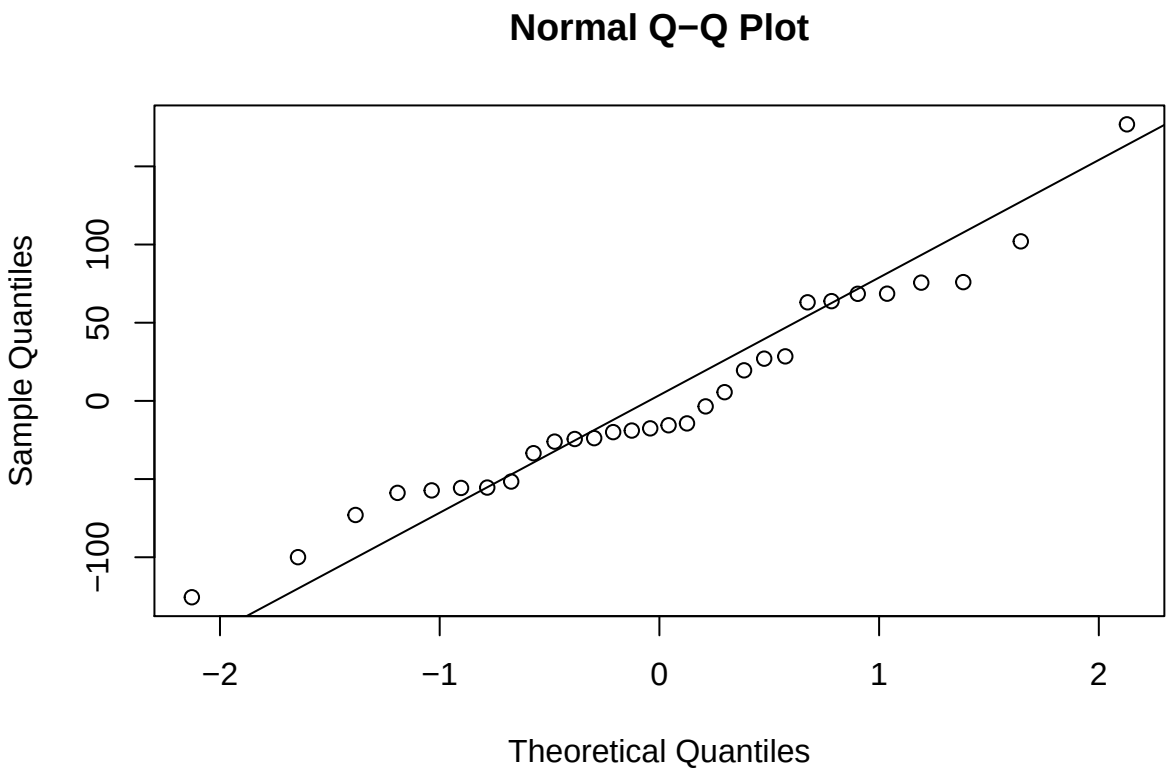
Nearly normal residuals: To check this condition, we can look at a histogram

```
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?

Answer: We can see from the histogram and normal probability plots above that the residuals are nearly normally distributed

Constant variability:

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

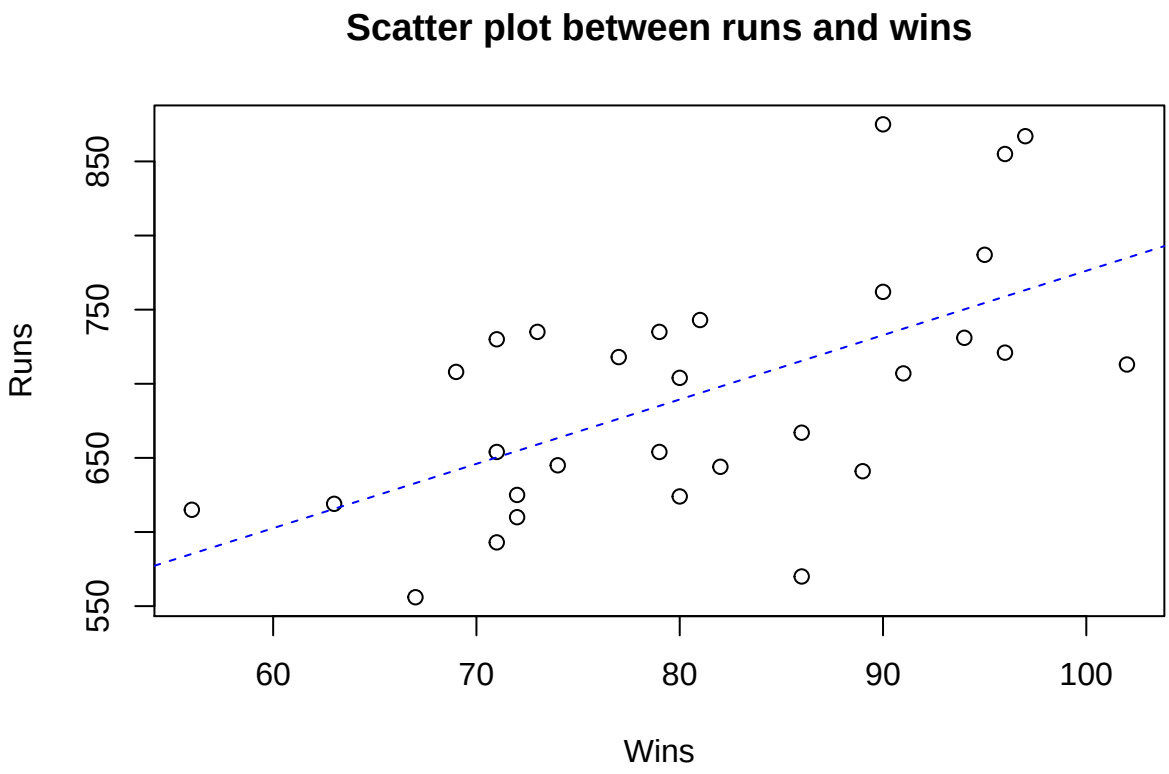
Answer: Based on the plot in (1) it can be concluded that condition for constant variability is met

---

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?

```
fit = lm(mlb11$runs~mlb11$wins)
plot(mlb11$wins, mlb11$runs, xlab="Wins", ylab="Runs", main="Scatter plot between runs and wins")
abline(coefficients(fit), lty=2, col="blue")
```



- 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?

```
fit = lm(runs~wins, data=mlb11)
summary(fit)
```

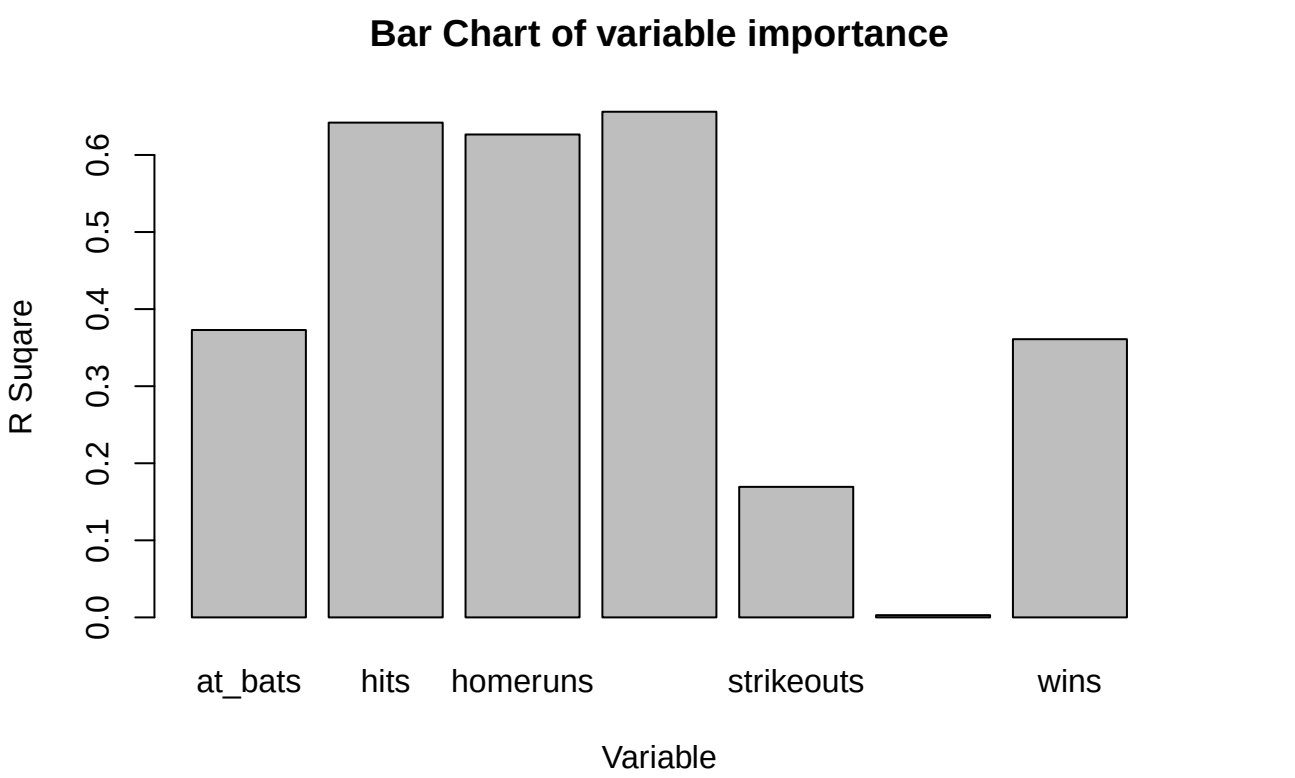
```
##
## Call:
## lm(formula = runs ~ wins, data = mlb11)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -145.450  -47.506   -7.482   47.346  142.186
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  342.121     89.223   3.834 0.000654 ***
## wins           4.341      1.092   3.977 0.000447 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 67.1 on 28 degrees of freedom
## Multiple R-squared:  0.361,  Adjusted R-squared:  0.3381
## F-statistic: 15.82 on 1 and 28 DF,  p-value: 0.0004469
```

Answer : The R Square value indicates lack of model fit to the data. It can be used as the

strength or usefulness of predictor variable while predicting outcome variable. Higer the R Square statistics better is the model fit. When we used at_bats variable to predict runs we got R square = 0.3729. When we use wins variable to predict runs the R square value is 0.361. Higher R square statistics of at_bats variable suggest that at_bats is a better predictor of runs compared to wins variable. Since there is no significant difference in the R square statistics between at_bats and wins we can't say that wins is far worst predictor than at_bats

- 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).

```
vars = c("at_bats","hits","homeruns","bat_avg","strikeouts","stolen_bases","wins")
rsquare = c()
for(i in vars)
{
 fit = lm(runs~get(i), data=mlb11)
 rsquare[i] = summary(fit)$r.squared
}

barplot(rsquare, xlab = "Variable", ylab="R Suqare", main="Bar Chart of variable importance")
```

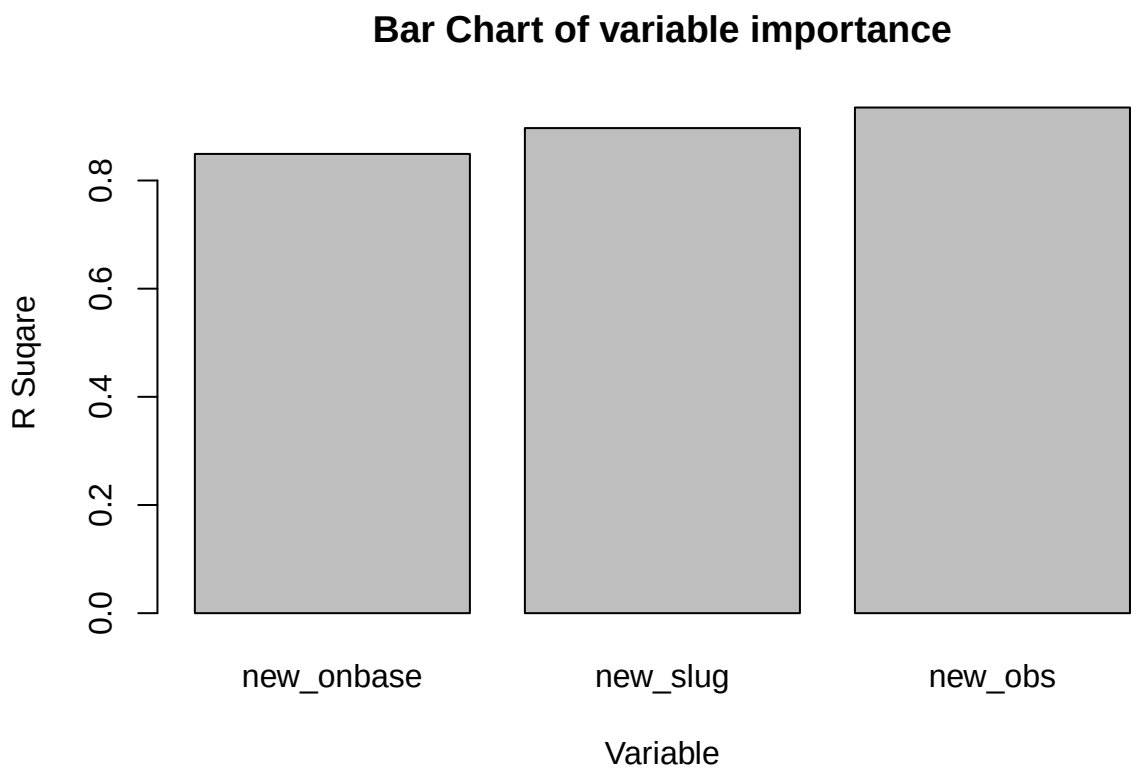


Answer: Above barplot summarizes the R Suqare value obtained for different variables when fitted a linear regression model separately to predict outcome variable runs. From the above graph it can be seen that highest R square is obtained for bat_avg variable. We can conclude that bat_avg variable is the best predictor of outcome variable runs

- 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 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?

```
vars = c("new_onbase", "new_slug", "new_obs"  )
rsquare = c()
for(i in vars)
{
 fit = lm(runs~get(i), data=mlb11)
 rsquare[i] = summary(fit)$r.squared
}

barplot(rsquare, xlab = "Variable", ylab="R Suqare", main="Bar Chart of variable importance")
```



Answer: Above barplot summarizes the R Suqare value obtained for different variables when fitted a linear regression model separately to predict outcome variable runs. From the above graph it can be seen that highest R square is obtained for new_obs variable. We can conclude that new_obs variable is the best predictor of outcome variable runs

After analyzing the R square values for all ten variables we can see the R square values for new variables are higher than R sqaure values for traditional variables. We can see that in general new variables does a better job in predicting outcome variable runs compared to traditional variables. After analyzing all ten variables we observed that new_obs variable have the highest R sqaure value 0.9349. We can conclude that new_obs variable is the best predictor of outcome variable runs amongst all ten variables

The variable new_obs represents "on base" percentage and "slugging percentage". It's a

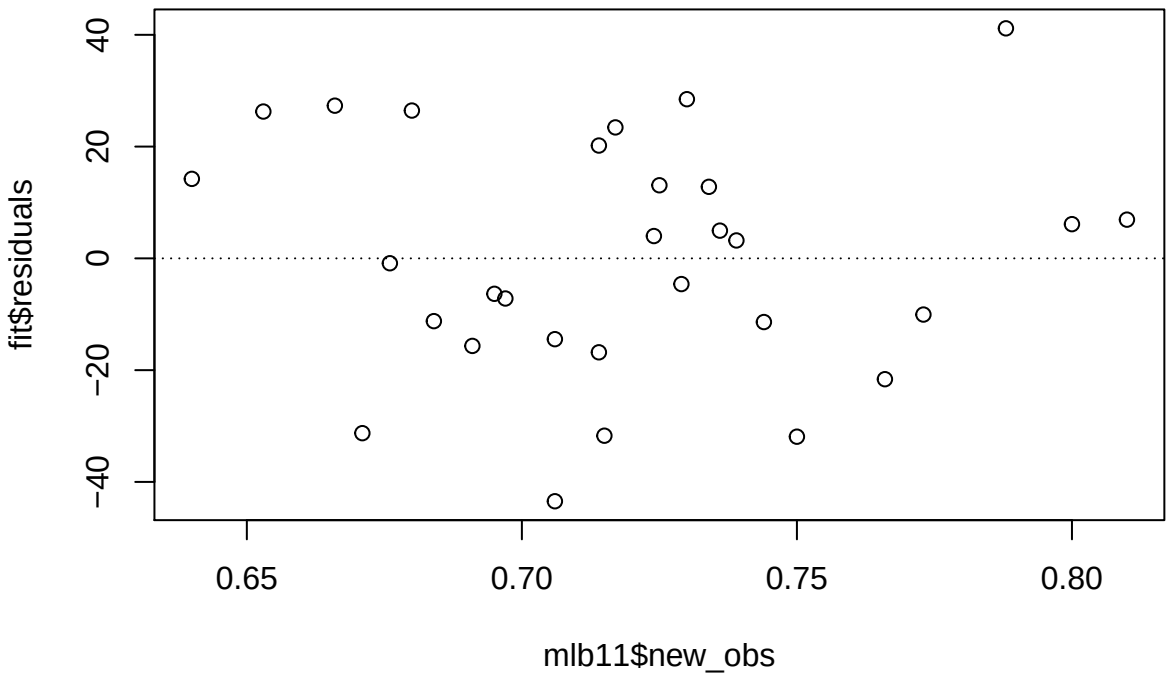
interactive term introduced to factor in both on base and slugging effect. No wonder it is doing the better job in predicting outcome variable runs compared to other variables. Interactive terms often factor in effect of multiple variables and results into increased model accuracy in most cases. Same effect is seen in this case with new_obs variable

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

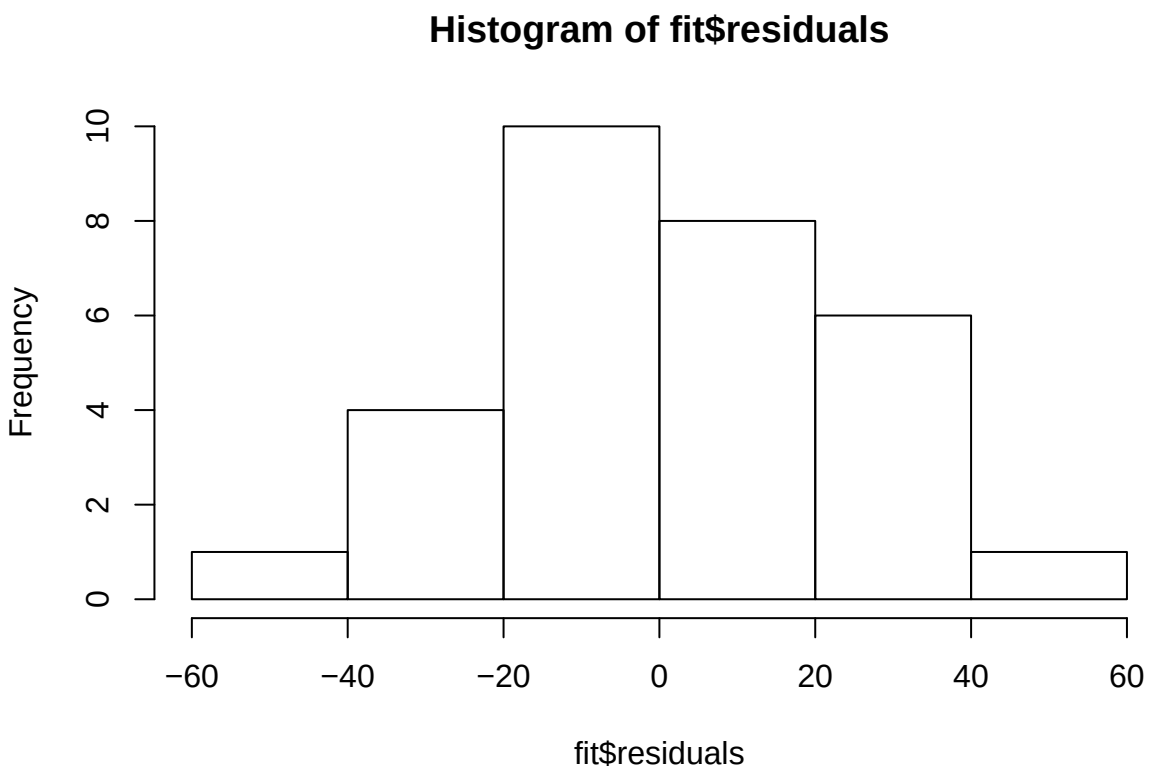
```
fit= lm(runs~new_obs, data=mlb11)
summary(fit)
```

```
##
## 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
```

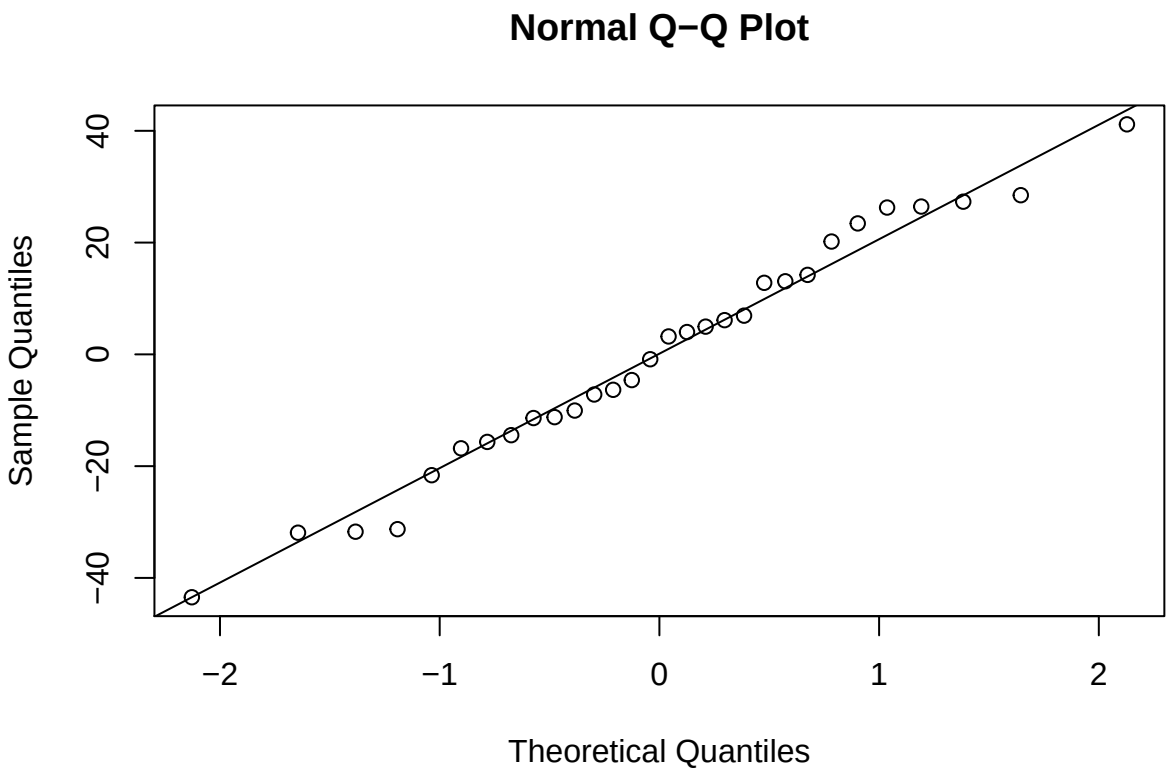
```
plot(fit$residuals ~ mlb11$new_obs)
abline(h = 0, lty = 3)  # adds a horizontal dashed line at y = 0
```




```
hist(fit$residuals)
```




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



Linear model assumes 1. Linear relationship between predictor and outcome variable 2. Constant variance of residuals 3. Near normal distribution of residuals

We can derive following conclusions from model summary stats above

- P value for new_obs variable is statistically significant indicating it is a important variable and play key role while predicting runs
- new_obs variable have a positive regression coefficient with value 1919.36. This indicates new_obs variable impacts runs positively
- From the regression coefficient of new_obs variable we can also conslude that for each increase in new_obs variable runs are increased by 1919.36

We can derive following conclusions from diagnostic plots above

- Residuals have constant variance
- Residuals are nearly normally distributed
- There is one unusual value observed for residuals on residual plot indicating it as a possible outlier