Joseph Flynn

STA 4155

1. After loading in the pizza data, we can regress Score on Calories, Type and Fat with the code:

imod1 <-lm(Score ~ Calories + Type + Fat, data = pizza).

From here, we can take the summary with:

summary(imod1)

This produces the summary below, which has produced the regression model

Score = -148.8173 + (0.7430)\*Calories + (15.6344)\*Type + (-3.8914)\*Fat.

Table

Description automatically generated

In this case, Type is a binary variable that means when a pizza is cheese, the regression line is moved up 15.6344. This means that cheese and peperoni have different intercepts, but similar slopes. However, only Calories is a significant variable at an alpha level of 0.05. This means that Type and Fat are not statistically significant.

1. From here, a residual plot and residual histogram of the model should be conducted to check for unusual observations. To do this, we use the code:

par(mfrow =c(1, 2))

plot(imod1$fitted.values, imod1$residuals, xlab ='Fitted values', ylab ='Residuals')

abline(0, 0)

hist(imod1$residuals, main = "Histogram of Residuals", xlab ='Residuals')

This produces the two graphs shown below:

Chart, histogram

Description automatically generated

The residual plot above clearly shows an n-shaped curve, which means the linearity assumption is violated and two-points in the lower right hand corner away from the other data. While the histogram of residuals shows a slight bell-curve with the middle missing. For the normality assumption, I would be caution of when proceeding. To find what points are causing this, we should first take the standardized residuals and with the code:

std.res <-rstandard(imod1)

lev <-hatvalues(imod1)

par(mfrow =c(1,2))

hist(std.res, xlab ='Studentized Residuals', main = "")

hist(lev, xlab ='Leverages', main = "")

This produces the graphs below:

Chart, histogram

Description automatically generated

The studentized residuals don’t seem to have a large spread with the results looking like a slightly left skewed distribution, but all points seem to be with 3 standard deviations. So, there doesn’t appear to be any outliers. Even then, there are only two points past 2 standard deviations. While the leverage points are fairly close together and when checking for high leverage points, there are none. This is further verified with reports generated with the code:

pizza[which(abs(std.res)>3),c(2,4,5,6)]

lev.cut <- 3\*length(imod1$coefficients)/dim(pizza)[1]

pizza[which(lev>lev.cut),c(2,4,5,6)]

These lines of code produce the reports:

Text

Description automatically generated

Next is to check the Cook’s distance and DFFITS. To check this, we use the commands:

cv <- 2\*sqrt(length(imod1$coefficients)/dim(pizza)[1])

cookd <-cooks.distance(imod1)

par(mfrow =c(1,2))

hist(cookd, xlab = "Cook's distance", main ='')

plot(dffits(imod1) ,ylab = "Standardized DFFits", xlab = "Index", main = "Standardized DFFits")

abline(h = cv, lty = 2)

abline(h = -cv, lty = 2)

The produced graphs are:

Chart, scatter chart

Description automatically generated

The Cook’s distance and DFFITS both have some clearly odd points. For the Cook’s distance, there are three data points that are visibly far from the rest of the data. However, these points don’t have the largest Cook’s distance, so they should be investigated further. While the DFFITS has three points below the cutoff of -2\*sqrt(k/n). The three points in DFFITS are very distance and set off red flags. These points should be identified and to do this we use the code:

pizza[which(cookd>.3),c(2,4,5,6)]

pizza[which(abs(dffits(imod1))>cv),c(2,4,5,6)]

The following reports are given:

Text

Description automatically generated

In both, the marked influential points are 12, 16, and 29. These correspond to Reggio, Michelina, and Healthy Choice Pepperoni.

1. So, let’s start by removing the influential points found earlier. For this, a new dataset excluding the variable is best to create and to do this we use the code:

pizza2 = pizza[-c(12,16,29),]

Then we reset up the regression from earlier with the new dataset. The code and summary are shown below.

pizza2 = pizza[-c(12,16,29),]

imod2 <-lm(Score ~ Calories + Type + Fat, data = pizza2)

summary(imod2)

Text, table

Description automatically generated

The new model is significantly different from before with a new equation of

Score = -351.9436 + (1.5951)\*Calories + (18.1209)\*Type + (-9.8278)\*Fat. In addition, all the coefficients are statistically significant at an alpha level of 0.05. The new model is all around better with a higher R-squared, higher adjusted R-squared, and passes the F test by a larger margin. However, the R-squared and adjusted R-squared are on the lower side. The next step would be to check the assumptions. To checks this, we need to produce some graphs with the first being a residual plot which is produced with the code below.

plot(imod2$fitted.values, imod2$residuals, xlab ='Fitted values', ylab ='Residuals')

abline(0, 0)

The graph for this code:

Scatter chart

Description automatically generated

From the graph above, there is no pattern and that means the linearity assumption is meant. There also doesn’t seem to be any outliers. However, the equal variance assumption may not be met because there seems to be a decreasing trend, so proceed with caution. Now, it is time to check normality assumption with a histogram of the residuals and a q-q plot:

par(mfrow =c(1,2))

hist(imod2$residuals, main ='', xlab ='Residuals')

qqnorm(imod2$residuals)

qqline(imod2$residuals)

Chart, histogram

Description automatically generated

From the graphs above, the normality assumption is satisfied. For the residual histogram is close to a normal distribution and the points of the Q-Q plot are very close to the q-q line, with only 1 point a little bit off. Allowing for the normality assumption to be accepted.

1. The next step would be to check the collinearity between all the variables. To do this, the car package needs to be installed. After installing the package, we can run the code:

vif(imod2)

which has the output:

A picture containing text

Description automatically generated

From here, there is clear collinearity between Fat and Calories because of their high values. This can be verified by taking the correlation between these two variables:

cor(pizza2$Calories,pizza2$Fat)

A picture containing logo

Description automatically generated

There is currently a high amount of correlation between calories and pizza. The reason for this is that food with high fats have a larger number of calories, due to the fats having a lot of calories. So, the more fats cause a larger number of calories.

1. If we consider the Calories and Type, an interaction term may be needed. This is due to pepperoni pizza having more toppings than plain pizza, which would cause pepperoni pizza to have a larger number of calories. Thus, creating a need for an interaction term. To test this, we can add an interaction term to the model and check it with the code:

imod3 <-lm(Score ~ Calories + Type + Fat + Calories\*Type, data = pizza)

summary(imod3)

Table

Description automatically generated

From the new model that uses the interactive term and the whole dataset, we can see that all the variables are statistically significant at an alpha level of 0.05. This means that the interaction term is statistically significant. In addition, the R-squared, adjusted R-squared, and F-stat are higher than model 1, which also uses the whole dataset. However, the new model has a lower R-squared, adjusted R-squared, and F-stat than model 2, which had outliers and influential points removed. So, if you wanted to use the whole dataset, it is best to add the interaction term because it creates a more accurate model than without it.