FORMATIVE ASSESSMENT 4
DSC1105
March 2025
Zion John Yousef T. Ramilo

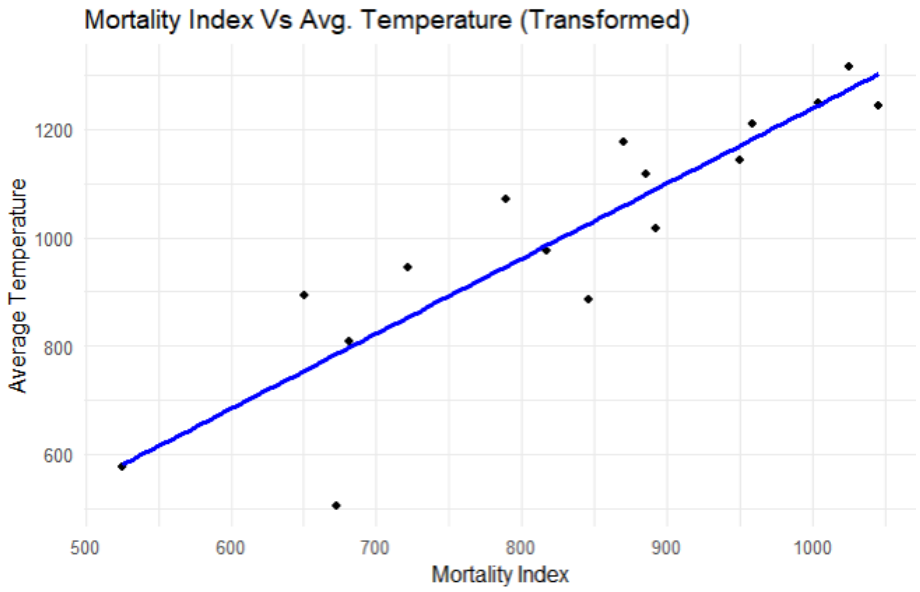
1. Using the Mortality by Latitude data Download Mortality by Latitude data, make a plot of the mortality index against mean average temperature.
    a. Is it hollow up or hollow down?
    b. Try to identify a transformation of one of the variables that will straighten out the relationship.
    c. Make a plot of the residuals to check for any remaining patterns.

```
library(tidyverse)
library(broom)
library(MASS)

mortality_by_latitude <- read_csv("Formative Assessment 4/mortality_by_latitude.csv")

bc <- boxcox(mortality_by_latitude$temperature~mortality_by_latitude$mortality_index)
best_lambda <- bc$x[which.max(bc$y)]
print(best_lambda)

Output:
>> 2
```

Remarks:
Box-Cox transformation shall be used to identify whether or not the given behavior of the data is hollow up or hollow down. Through the lambda which is currently defined as 2 being the best lambda for a Box-Cox transformation, which is positive, therefore the data is hollow downward.

Given by $\lambda = 2$ our Box-Cox transformation on y which in this case is the temperature, will have the following form $\frac{y^2-1}{2}$.

If we create a plot for the transformed data it would take a positive relationship as shown in the graph below.

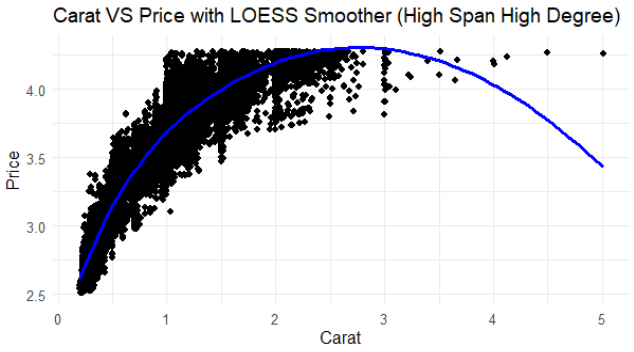


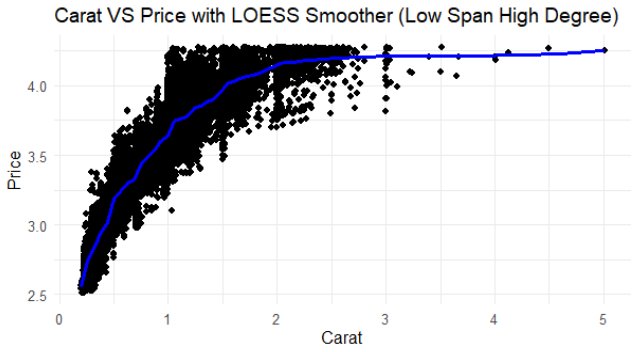
Creating a linear model of the given data shows the residuals, shown below.

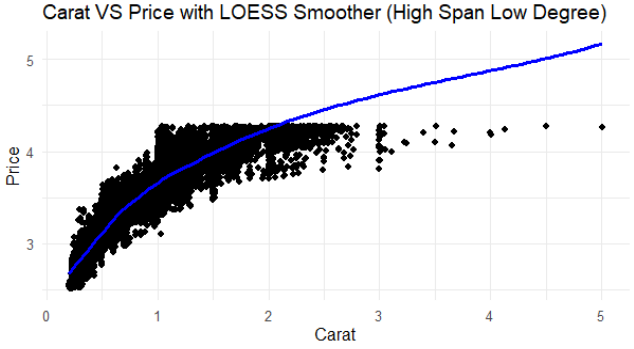
```
lm_model <- lm(transformed_temp ~ mortality_index, data = mortality_by_latitude)
tidy(lm_model)
augment(lm_model)
glance(lm_model)

Output:
> tidy(lm_model)
# A tibble: 2 × 5
  term            estimate std.error statistic    p.value
  <chr>              <dbl>     <dbl>     <dbl>      <dbl>
1 (Intercept)      -148.     159.       -0.931 0.368
2 mortality_index     1.39     0.188     7.37  0.00000352

> glance(lm_model)
# A tibble: 1 × 12
  r.squared adj.r.squared sigma statistic  p.value    df logLik   AIC   BIC deviance df.residual
      <dbl>         <dbl> <dbl>     <dbl>    <dbl> <dbl>  <dbl> <dbl> <dbl>    <dbl>       <int>
1     0.795         0.780  110.      54.3  3.52e-6     1  -96.8  200.  202.  168693.          14
# ℹ 1 more variable: nobs <int>
```

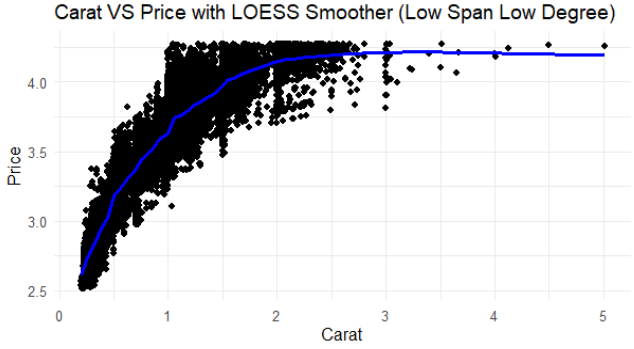
2. Using the same subset of the diamonds dataset, make a plot of log price as a function of carat with a loess smoother. Try several values for the span and degree arguments and comment briefly about your choice.

```
diamonds = diamonds %>% mutate(log_price = log10(price))
dia_log_plot_original <-ggplot(diamonds, aes(x = carat, y = log_price)) +
  geom_point() +
  geom_smooth(method = "loess", span = 0.75, se = FALSE, color = "blue") +
  labs(title = "Carat VS Price with LOESS Smoother", x = "Carat", y = "Price") +
  theme_minimal()

dia_log_plot_lowSpan_highDegree <-ggplot(diamonds, aes(x = carat, y = log_price)) +
  geom_point() +
  geom_smooth(method = "loess", span = 0.05,method.args = list(degree = 2), se = FALSE,
color = "blue") +
  labs(title = "Carat VS Price with LOESS Smoother (Low Span High Degree)", x = "Carat",
y = "Price") +
  theme_minimal()

dia_log_plot_highSpan_lowDegree <-ggplot(diamonds, aes(x = carat, y = log_price)) +
  geom_point() +
  geom_smooth(method = "loess", span = 0.75,method.args = list(degree = 1), se = FALSE,
color = "blue") +
  labs(title = "Carat VS Price with LOESS Smoother (High Span Low Degree)", x = "Carat",
y = "Price") +
  theme_minimal()

dia_log_plot_lowSpan_lowDegree <-ggplot(diamonds, aes(x = carat, y = log_price)) +
  geom_point() +
  geom_smooth(method = "loess", span = 0.05,method.args = list(degree = 1), se = FALSE,
color = "blue") +
  labs(title = "Carat VS Price with LOESS Smoother (Low Span Low Degree)", x = "Carat",
y = "Price") +
  theme_minimal()
```









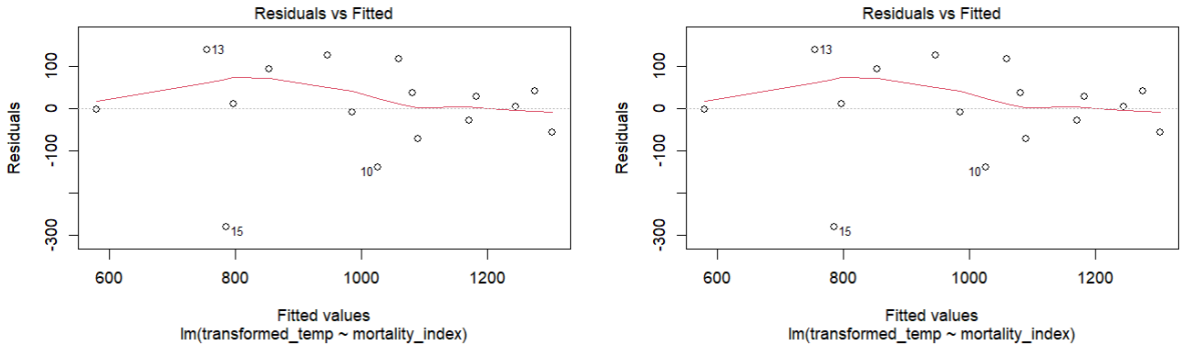
Remarks:
I have chosen 4 combinations of the given span and degree, the plot with a high span and high degree takes upon a quadratic form and is very smooth, it interprets that as the carat goes up the price goes down. The plot with a high span but low degree specifically a degree of 1 takes a smooth form and somewhat resembles a cubic polynomial, its interpretation shows that the price goes up as the carat increases. For the plots that have a low span, they have almost the same behavior wherein the interpretation shows that the price somewhat plateaus at some price as the carat increases.

3. Compare the fit of the loess smoother to the fit of the polynomial + step function regression using a plot of the residuals in the two models. Which one is more faithful to the data?

```
model1<-lm(transformed_temp ~ mortality_index, data = mortality_by_latitude)
model2<-lm(transformed_temp ~ mortality_index+I(mortality_index^2), data =
mortality_by_latitude)
model3<-lm(transformed_temp ~ mortality_index+I(mortality_index^2)+I(mortality_index^3),
data = mortality_by_latitude)

kable(anova(model1), caption = "ANOVA Model 1")
kable(anova(model2), caption = "ANOVA Model 2")
kable(anova(model3), caption = "ANOVA Model 3")
```

```
> kable(anova(model1), caption = "ANOVA Model 1")

Table: ANOVA Model 1
|                | Df|   Sum Sq|   Mean Sq|  F value|  Pr(>F)|
|:---------------|--:|--------:|---------:|--------:|-------:|
|mortality_index |  1| 654226.7| 654226.66| 54.29482| 3.5e-06|***
|Residuals       | 14| 168693.3|  12049.52|       NA|      NA|

> kable(anova(model2), caption = "ANOVA Model 2")

Table: ANOVA Model 2
|                     | Df|      Sum Sq|     Mean Sq|    F value|    Pr(>F)|
|:--------------------|--:|-----------:|-----------:|----------:|---------:|
|mortality_index      |  1| 654226.6571| 654226.6571| 50.5926091| 0.0000079|***
|I(mortality_index^2) |  1|    586.8254|    586.8254|  0.0453803| 0.8346105|
|Residuals            | 13| 168106.5021|  12931.2694|         NA|        NA|

> kable(anova(model3), caption = "ANOVA Model 3")

Table: ANOVA Model 3
|                     | Df|      Sum Sq|     Mean Sq|    F value|    Pr(>F)|
|:--------------------|--:|-----------:|-----------:|----------:|---------:|
|mortality_index      |  1| 654226.6571| 654226.6571| 46.8225473| 0.0000179|***
|I(mortality_index^2) |  1|    586.8254|    586.8254|  0.0419987| 0.8410573|
|I(mortality_index^3) |  1|    436.8569|    436.8569|  0.0312655| 0.8625985|
|Residuals            | 12| 167669.6452|  13972.4704|         NA|        NA|
```

```
step_model <- step(model1, direction = "both")
summary(step_model)
plot(step_model, which = 1)

summary(model1)
plot(model1, which = 1)
plot_grid(plot(step_model, which = 1),
          plot(model1, which = 1),
          ncol = 2)
```



Remarks:

Creating 3 polynomial models with increasing degrees and subjecting them to an ANOVA shows that only the linear model or linear part of the polynomial is able to explain the variance within the data indicating that model 1 is the best model for us to use for the step function as well as the main model for comparison. Given by the plots above it shows that there are no differences with the residuals of both of the models of the step function and the linear model, which indicates that the best model is indeed the linear model.