DAT-4253 LM 9 - Lab 2: Regression with Transformations

AUTHOR
Aaron Younger

Business Understanding

The client, Jack Person, the Executive Director of a health public policy organization, has extended his health research into the area of assessing personal well-being. This Analysis will be focused on finding the relationship that age and income have on happiness.

Data Understanding

R Version

```r
suppressWarnings(RNGversion("3.5.3"))
options(scipen=999)
```

Libraries

```r
library(readxl)
library(DataExplorer)
library(tidyverse)
library(dplyr)
library(ggplot2)
library(e1071)
library(dlookr)
library(caret)
library(moments)
library(auditor)
library(Metrics)
```

Import Dataset

```r
happy_data <- read_excel("jaggia_ba_2e_ch08_data.xlsx",
    sheet = "Happiness")
View(happy_data)
```

Dataset Exploration

```r
happy_data %>% head()
```

```
# A tibble: 6 × 3
  Happiness   Age Income
      <dbl> <dbl>  <dbl>
1        69    49  52000
2        83    47 123000
3        86    72 112000
4        73    52 166000
5        89    68  90000
6        81    37 152000
```

```r
happy_data %>% tail()
```

```
# A tibble: 6 × 3
  Happiness   Age Income
      <dbl> <dbl>  <dbl>
1        82    77  76000
2        58    52  72000
3        78    75  28000
4        91    79 109000
5        57    47  29000
6        79    31 105000
```

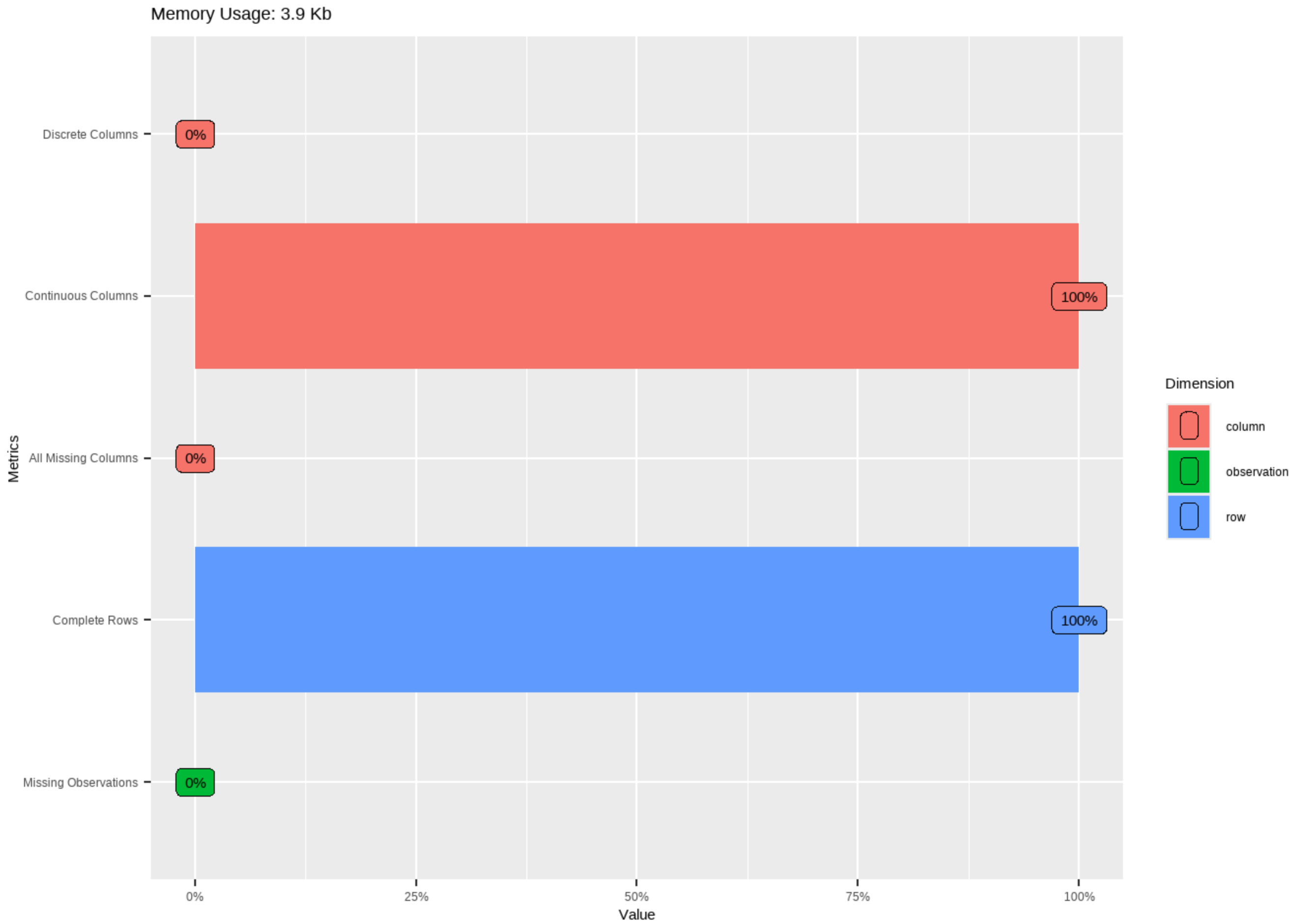
```r
happy_data %>% nrow()
```

```
[1] 100
```

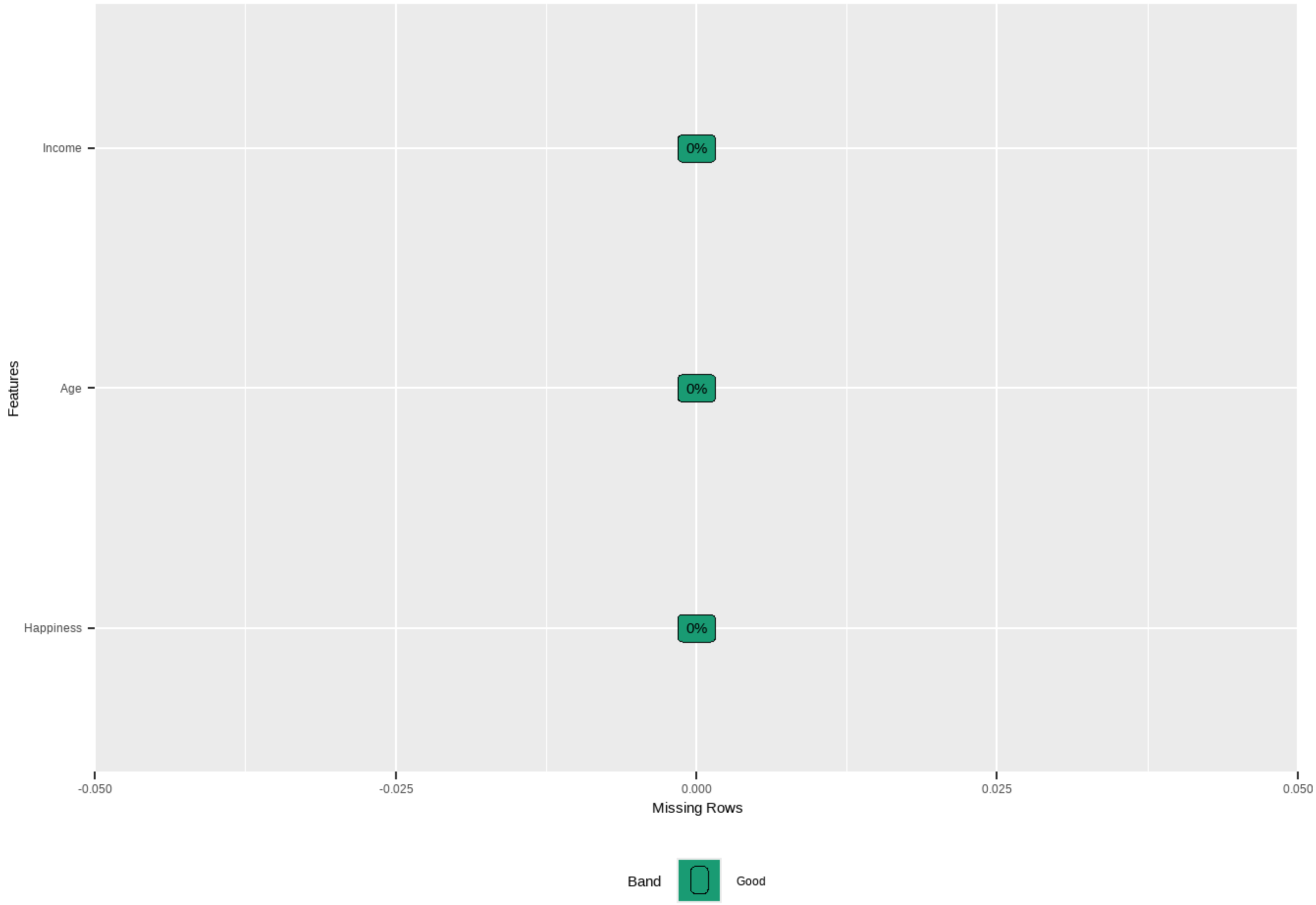
```r
happy_data %>% ncol()
```

```
[1] 3
```

```r
happy_data %>% plot_intro()
```




```r
happy_data %>% plot_missing()
```




```
happy_data %>% str()
```

```
tibble [100 × 3] (S3: tbl_df/tbl/data.frame)
 $ Happiness: num [1:100] 69 83 86 73 89 81 75 56 75 57 ...
 $ Age      : num [1:100] 49 47 72 52 68 37 48 48 56 51 ...
 $ Income   : num [1:100] 52000 123000 112000 166000 90000 152000 58000 50000 93000 27000 ...
```

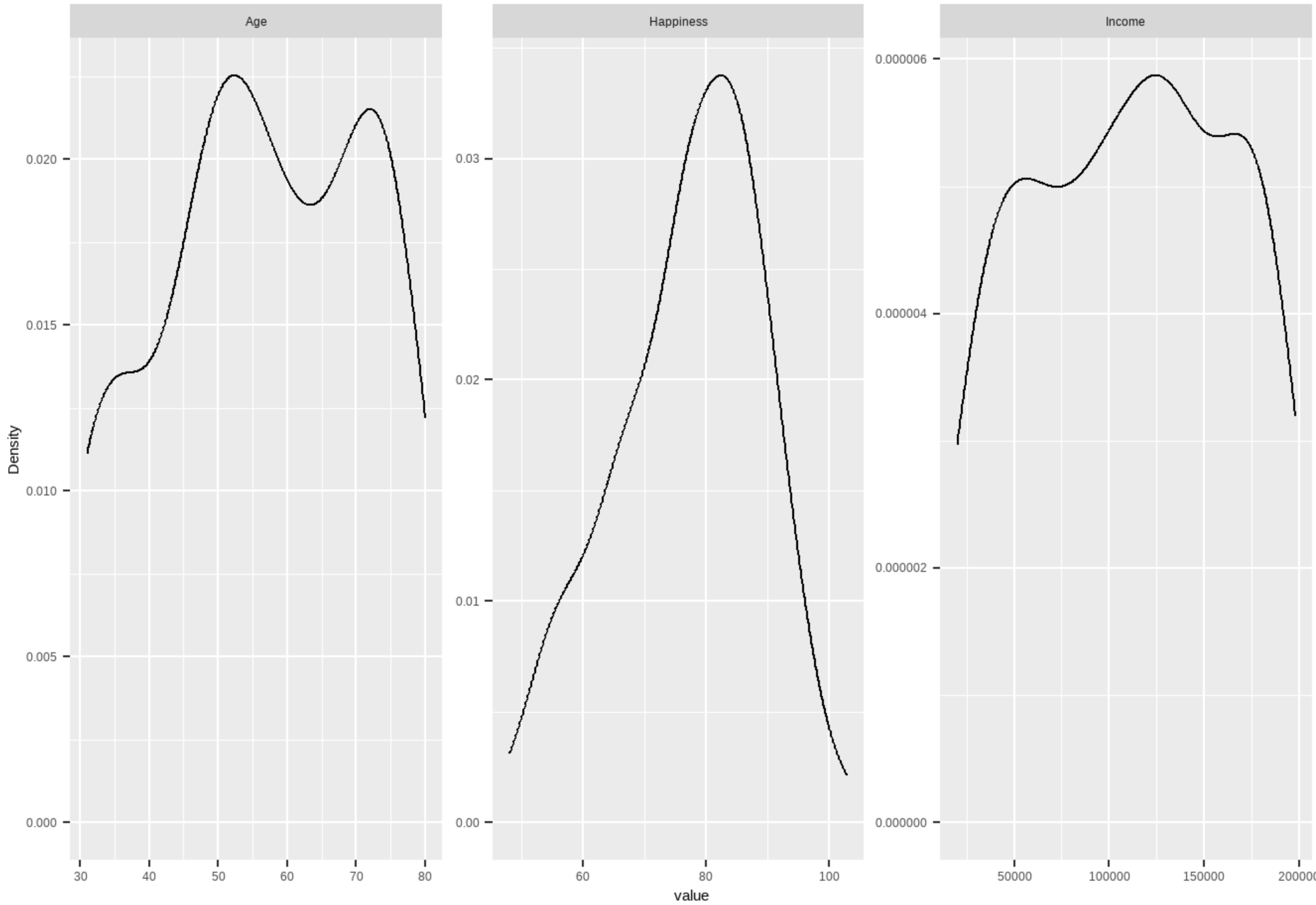
Comments on Data Exploration:
This dataset has 100 rows and three columns. All columns are continuous/numeric columns, the columns are Happiness, Age, and Income. This dataset has no missing values.

Variable Key:
- Age: Represents the Age of the individual.
- Happiness: Overall happiness score, higher values represent greater happiness. The Dependent Variable.
- Income: Annual personal income.

EDA

Distribution of Numeric Values

```
happy_data %>% plot_density()
```



Comments on distribution of numeric values:
Based of the density plots, Happiness and Income seem to be normally distributed. Age seems to also be relatively normally distributed but bimodal with a two clusters of ages.

Skewness of Numeric Variables

```
apply(happy_data[,1:3], 2, skewness)
```

```
  Happiness         Age      Income 
-0.42322611 -0.16478456 -0.06711691 
```

Comments on Skewness:
These variables are left skewed but no extreme enough to base non-normality off skewness.

```
apply(happy_data[,1:3], 2, kurtosis)
```

```
Happiness       Age    Income 
 2.574310  1.889837  1.788168 
```

Comments on Kurtosis:
All three variables have kurtosis values close to 3 or slightly below, which indicates the variables have no major deviation from normality.

Look For Outliers

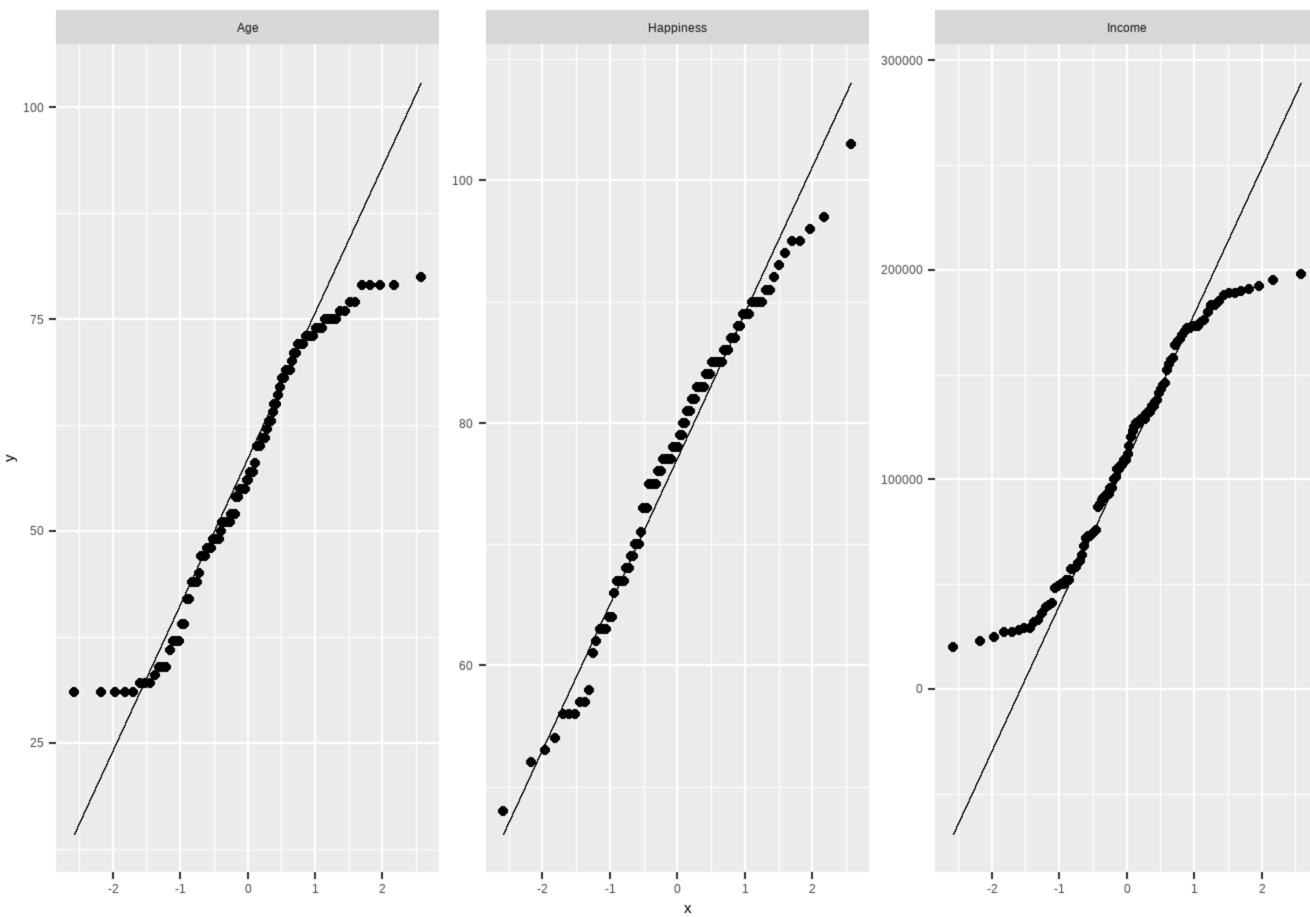
```
diagnose_outlier(happy_data)
```

```
# A tibble: 3 × 6
  variables outliers_cnt outliers_ratio outliers_mean with_mean without_mean
  <chr>            <int>          <dbl>         <dbl>     <dbl>        <dbl>
1 Happiness            0              0           NaN      77.2         77.2
2 Age                  0              0           NaN      56.5         56.5
3 Income               0              0           NaN  110820       110820
```

Comments on Outliers:
This dataset has no outliers.

Q-Q Plots

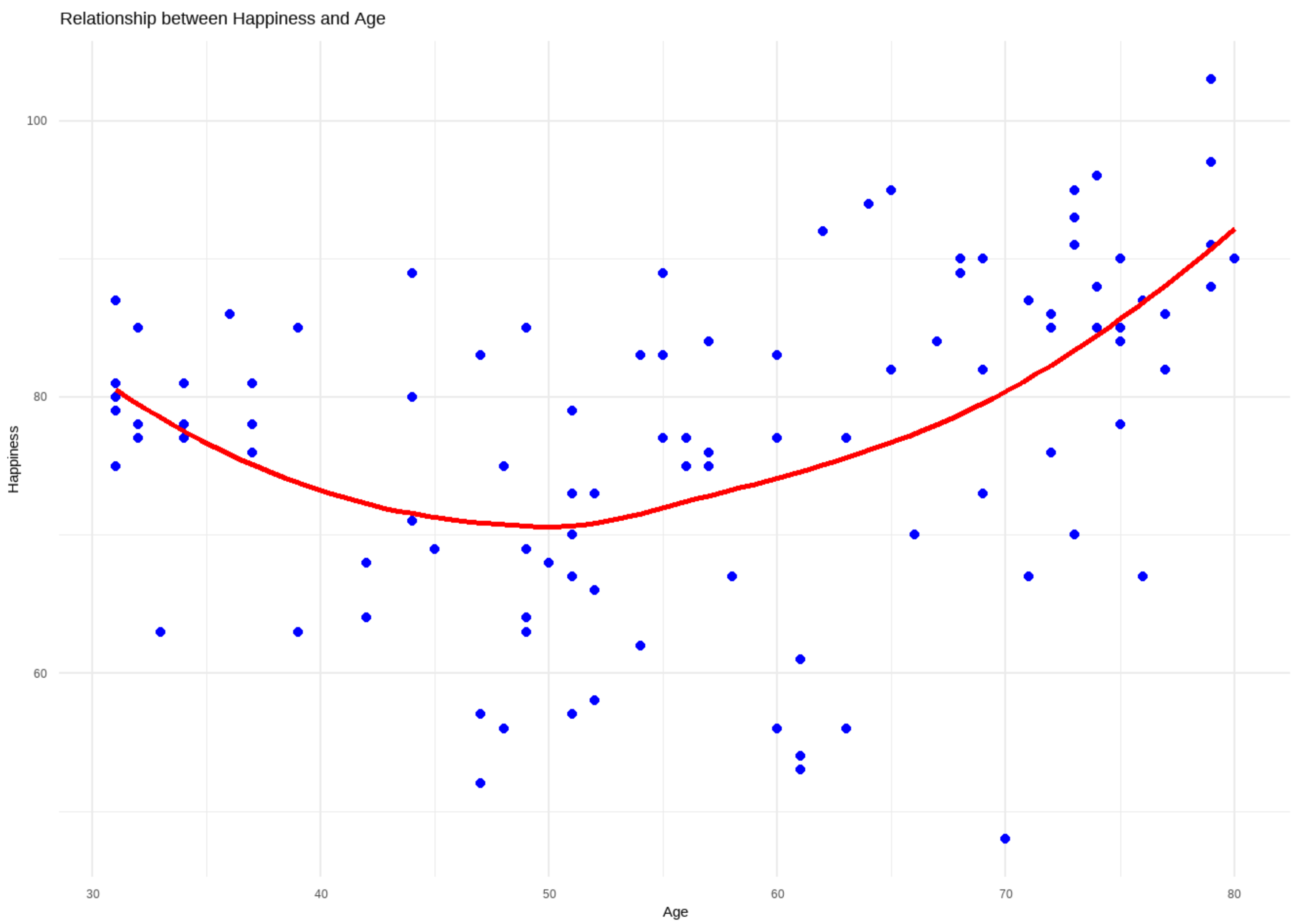
```
DataExplorer::plot_qq(happy_data)
```



Graphing Numeric Variables

Scatterplot of Happiness by Age

```r
ggplot(happy_data, aes(x = Age, y = Happiness)) +
  geom_point(color = "blue") +
  geom_smooth(method = "loess", color = "red", se = FALSE) +
  labs(
    title = "Relationship between Happiness and Age",
    x = "Age",
    y = "Happiness"
  ) +
  theme_minimal()
```

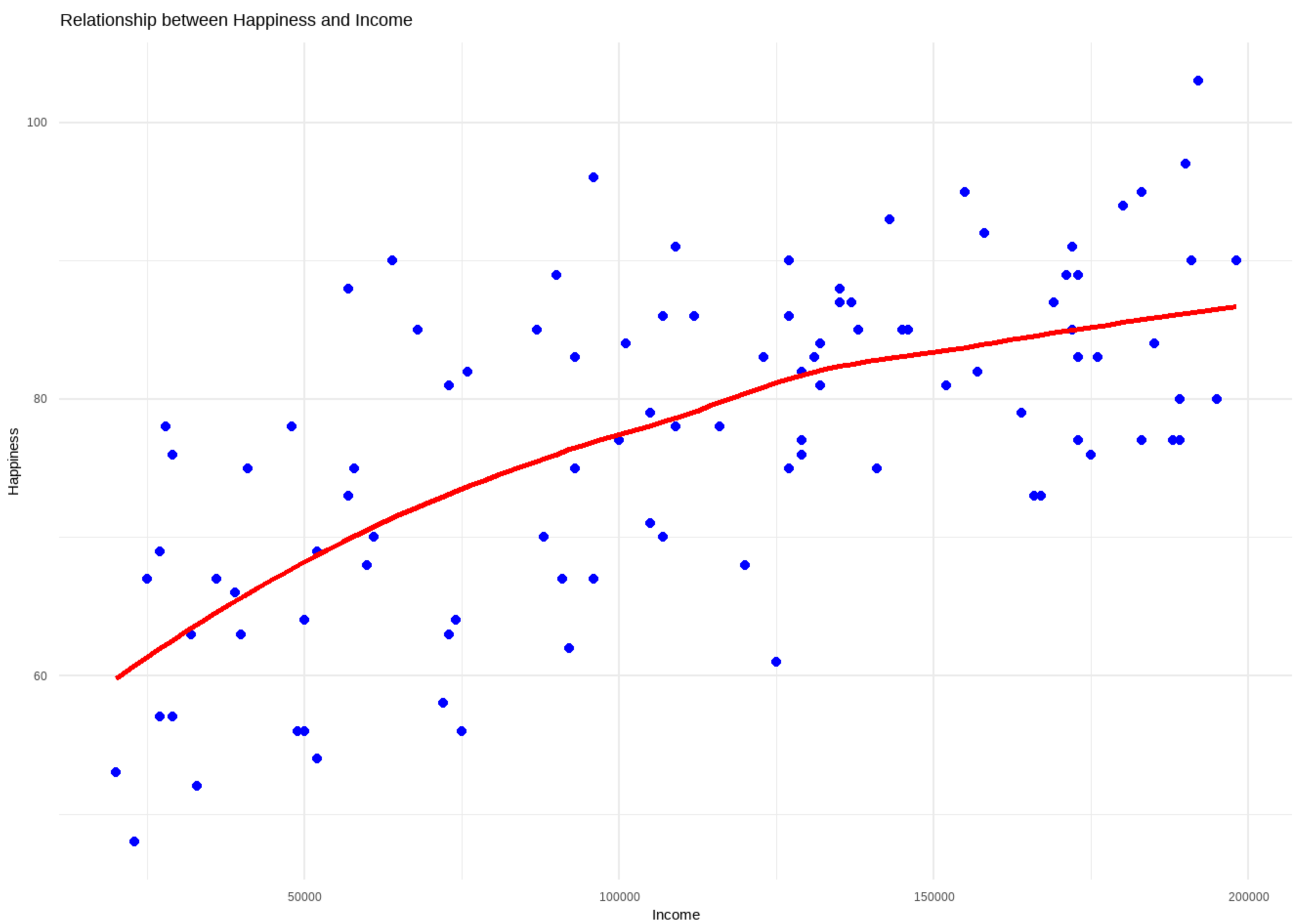


Comments on Relationship between Happiness and Age:
Age and Happiness have a quadratic relationship, happiness is higher in the 30’s then happiness goes down until about age 55 then it happiness goes back up.

Scatterplot of Happiness by Income

```r
ggplot(happy_data, aes(x = Income, y = Happiness)) +
  geom_point(color = "blue") +
  geom_smooth(method = "loess", color = "red", se = FALSE) +
  labs(
    title = "Relationship between Happiness and Income",
    x = "Income",
    y = "Happiness"
```

```
  ) +
  theme_minimal()
```

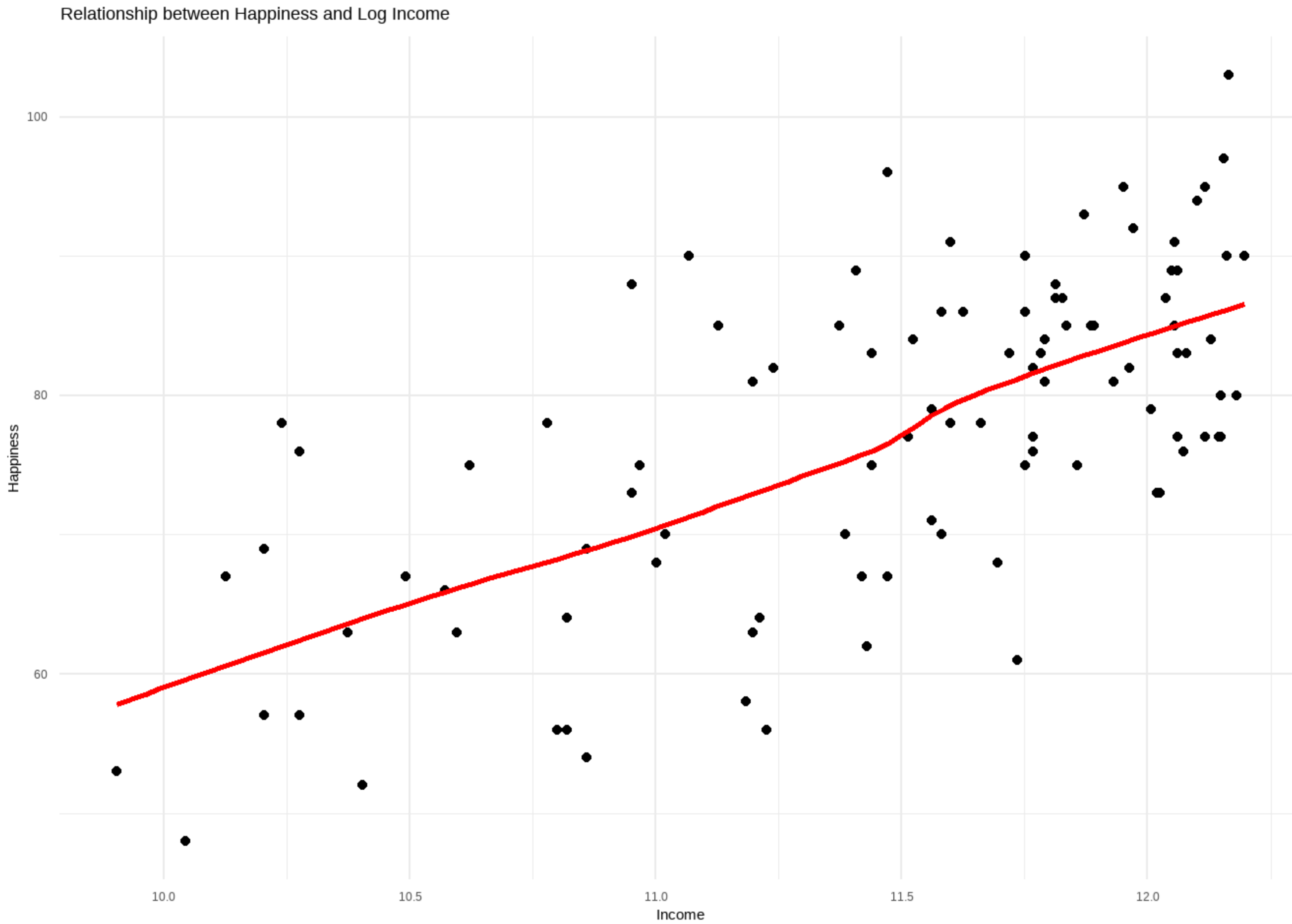


Transform Income to Log

```
happy_data <- happy_data %>%
  mutate(log_income = log(Income))
View(happy_data)
```

Scatterplot of Happiness by log Income

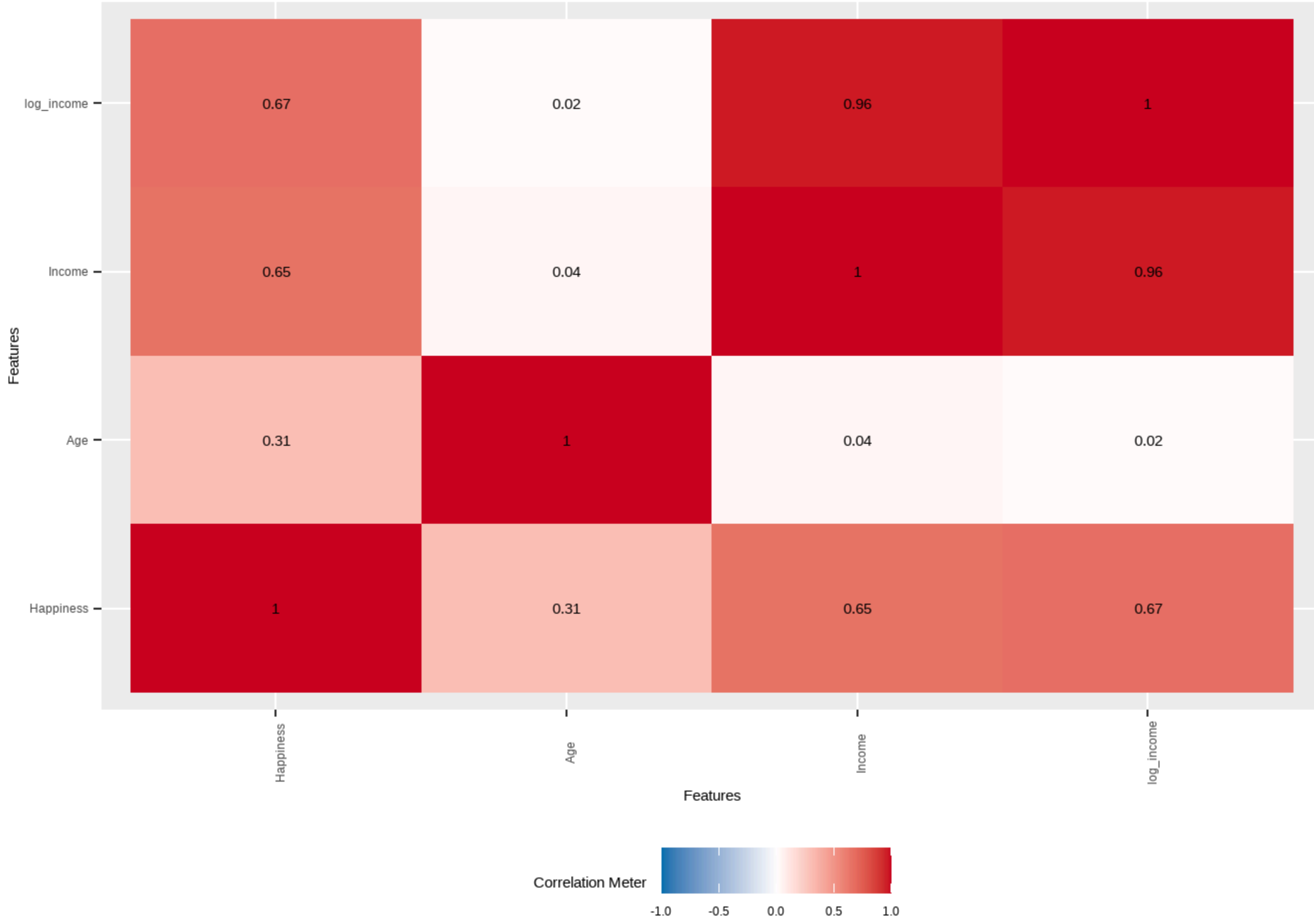
```
ggplot(happy_data, aes(x = log_income, y = Happiness)) +
  geom_point() +
  geom_smooth(method = "loess", color = "red", se = FALSE) +
  labs(
    title = "Relationship between Happiness and Log Income",
    x = "Income",
    y = "Happiness"
  ) +
  theme_minimal()
```



Comments on relationship between Happiness and Income/log Income:
Happiness and Income have a positive relationship where as income goes up so does happiness, however this relationship is not perfectly linear. The relationship between happiness and log income is also positive, as log income goes up so does happiness, however this relationship is a lot more linear.

Correlation Matrix

```
DataExplorer::plot_correlation(happy_data)
```



Comments on Correlation Matrix:
All predictor variables are positively correlated with happiness, income is strongly correlated with happiness and age is moderately correlated with happiness. Happiness is also slightly more correlated to log_income then non-log income.

Data Preperation

```
happy_data <- happy_data %>%
  mutate(quad_age = Age^2)


set.seed(1)
my_index <- createDataPartition(happy_data$Happiness, p = 0.8, list = FALSE)
trainset <- happy_data[my_index, ]
testset <- happy_data[-my_index, ]

## Mean is very close
mean(trainset$Happiness)
```

```
[1] 77.07407
```

```
mean(testset$Happiness)
```

```
[1] 78
```

Comments on Data Partition:
Before modeling the dataset is partitioned into a 80/20 split so the data can be trained then tested. A set.seed of 1 was also given for reproducibility of the model results. The mean of the dependent variable across train and test set was very close.

Modeling - Assume a 10% Level of Signifcance

Baseline Model

```
lm_ctrl <- trainControl(method = "cv", number = 10)

set.seed(1)
model1 <- train(Happiness ~ Age + Income, data = trainset, method = "lm", trControl = lm_ctrl)
summary(model1)
```

```
Call:
lm(formula = .outcome ~ ., data = dat)

Residuals:
     Min       1Q   Median       3Q      Max 
-19.5930  -6.3955   0.4875   6.8399  17.0774 

Coefficients:
               Estimate  Std. Error t value             Pr(>|t|)    
(Intercept) 48.91142792  4.40609904  11.101 < 0.0000000000000002 ***
Age          0.21984356  0.06921302   3.176              0.00214 ** 
Income       0.00014315  0.00001892   7.567      0.0000000000646 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 8.979 on 78 degrees of freedom
Multiple R-squared:  0.4778,	Adjusted R-squared:  0.4644 
F-statistic: 35.69 on 2 and 78 DF,  p-value: 0.000000000009889
```

Coefficient Significance:
- Age: Significant predictor of Happiness. For every one-year increase in age, on average Happiness increases on average by 0.2198 units, ceteris paribus.
- Income: Highly significant predictor of Happiness. For every one-unit increase in Income, on average Happiness increases on average by 0.000143 units, ceteris paribus.

Model with Quadratic Age and Log income

```
set.seed(1)
model2 <- train(Happiness ~ Age + quad_age + log_income, data = trainset, method = "lm",
          trControl = lm_ctrl)
summary(model2)
```

```
Call:
lm(formula = .outcome ~ ., data = dat)

Residuals:
     Min       1Q   Median       3Q      Max 
-16.0962  -4.0718  -0.1398   3.8957  13.6122 

Coefficients:
             Estimate Std. Error t value             Pr(>|t|)    
(Intercept) -8.960518  16.853668  -0.532                0.596    
Age         -2.743185   0.406527  -6.748       0.000000002486 ***
quad_age     0.026867   0.003644   7.372       0.000000000163 ***
log_income  13.072487   1.192429  10.963 < 0.0000000000000002 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 6.692 on 77 degrees of freedom
Multiple R-squared:  0.7137,	Adjusted R-squared:  0.7025 
F-statistic: 63.97 on 3 and 77 DF,  p-value: < 0.00000000000000022
```

Coefficient Significance:
- Age: Age is a significant predictor of happiness and negative, indicating that as age increases, happiness initially decreases when holding other variables constant.
- quad_age: Quadratic of age is a Significant predictor of happiness and positive, showing a quadratic relationship between Age and Happiness. This means originally happiness decreases as age increases but after a certain age happiness starts increasing again.
- log_income: Log Income is a significant predictor of happiness and positive, coefficient explanation in parial effects for Income.

Partial effects for Income:

What this is referring to is since income positively affects happiness but its in log form, each increase in income yields a smaller gain to happiness, a diminishing return. So for the model 2 coefficient of log income a 1% increase in income leads to a 0.13+ point increase in happiness, and a 10% increase in income leads to a 1.31% point increase in happiness, ceteris paribus.

Optima for Age

```
b1 <- coef(model2$finalModel)["Age"]
b2 <- coef(model2$finalModel)["quad_age"]
```

```
optimum_age <- -b1 / (2 * b2)
optimum_age
```

```
     Age
51.05157
```

Comments on the Optima Age:

The Optima Age is 50.5 which means after 50.5 age starts trending upwards again in relation to age.

Evaluation

Predict Models

```
pred_m1 <- predict(model1, newdata = testset)
pred_m2 <- predict(model2, newdata = testset)

testset_results <- cbind(
  testset,
  pred_m1,
  pred_m2
)
View(testset_results)
```

Numeric Evaluation Metrics

```
RMSE_m1 <- rmse(testset$Happiness, pred_m1)
RMSE_m2 <- rmse(testset$Happiness, pred_m2)

## MAE Metrics
MAE_m1 <- mae(testset$Happiness, pred_m1)
MAE_m2 <- mae(testset$Happiness, pred_m2)

## MAD Metrics
MAD_m1 <- mad(testset$Happiness, pred_m1)
MAD_m2 <- mad(testset$Happiness, pred_m2)

## MAPE Metrics
MAPE_m1 <- mape(testset$Happiness, pred_m1)
MAPE_m2 <- mape(testset$Happiness, pred_m2)

## Make table with values
metrics_table <- data.frame(
  Model = c("Model 1", "Model 2"),
   RMSE = c(RMSE_m1, RMSE_m2),
   MAE = c(MAE_m1, MAE_m2),
```

```
    MAD = c(MAD_m1, MAD_m2),
    MAPE = c(MAPE_m1, MAPE_m2)
)
metrics_table
```

```
    Model     RMSE     MAE      MAD       MAPE
1 Model 1 5.240984 4.42640 5.399061 0.05749564
2 Model 2 5.838992 4.87033 6.796802 0.06281986
```

REC Curve and Graph

```
m1_audit <- audit(model1, data = testset, y = testset$Happiness)
```

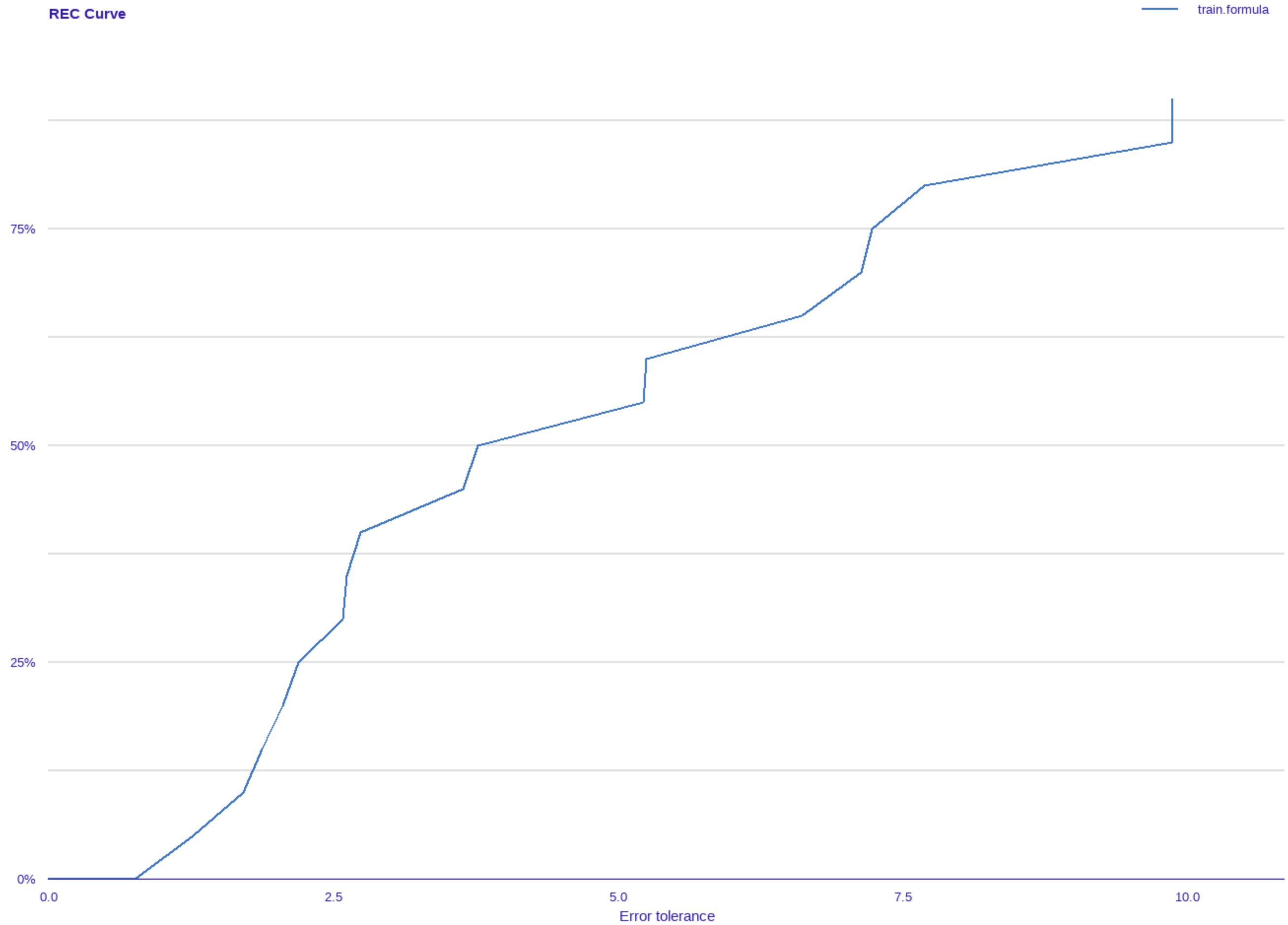
```
Preparation of a new explainer is initiated
  -> model label       :  train.formula  (  default  )
  -> data              :  19  rows  5  cols
  -> data              :  tibble converted into a data.frame
  -> target variable   :  19  values
  -> predict function  :  yhat.train  will be used (  default  )
  -> predicted values  :  No value for predict function target column. (  default  )
  -> model_info        :  package caret , ver. 6.0.94 , task regression (  default  )
  -> predicted values  :  numerical, min =  66.7341 , mean =  77.69749 , max =  92.74212
  -> residual function :  difference between y and yhat (  default  )
  -> residuals         :  numerical, min =  -9.867691 , mean =  0.3025088 , max =  9.86583
  A new explainer has been created!
```

```
m2_audit <- audit(model2, data = testset, y = testset$Happiness)
```

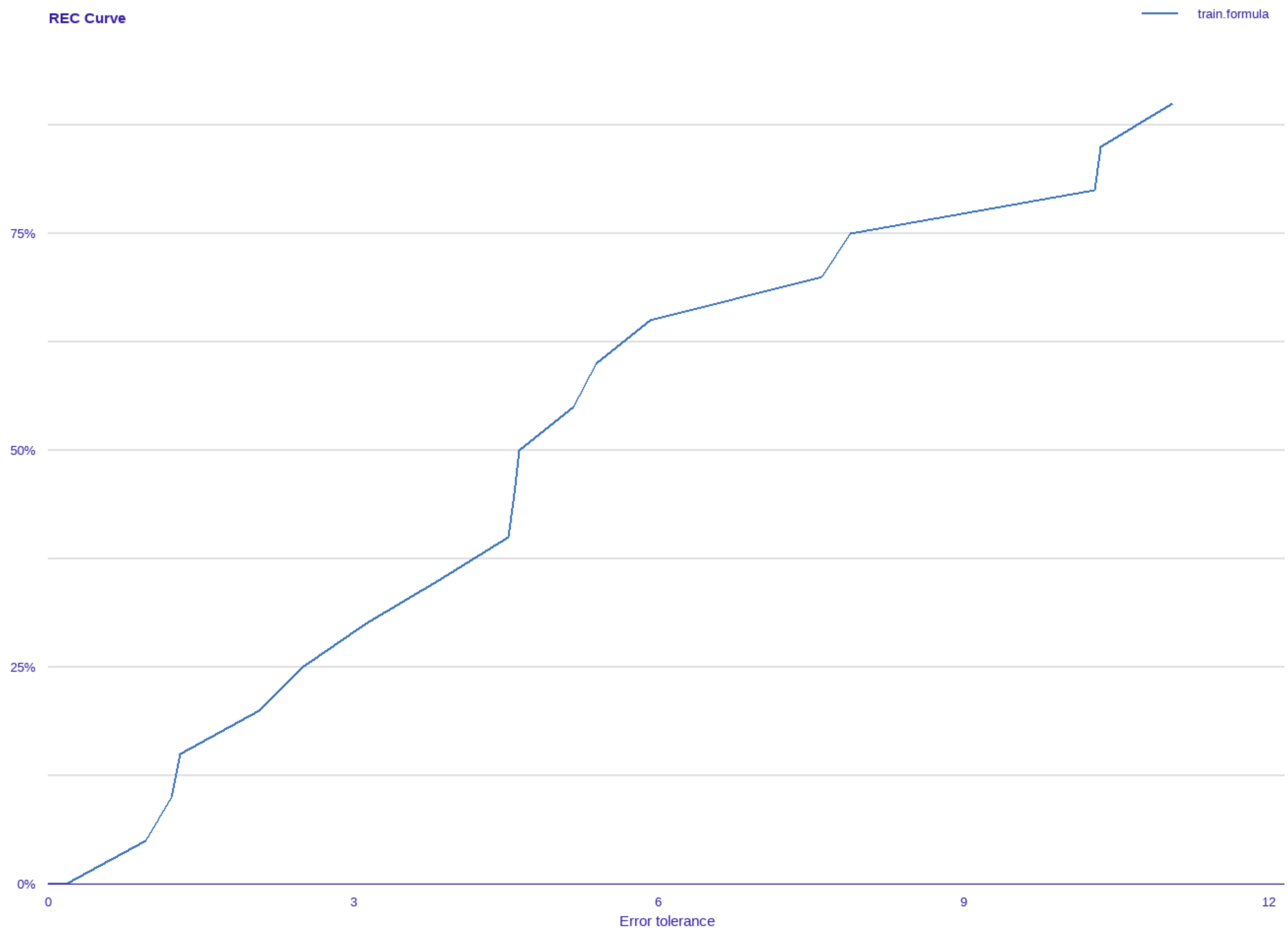
```
Preparation of a new explainer is initiated
  -> model label       :  train.formula  (  default  )
  -> data              :  19  rows  5  cols
  -> data              :  tibble converted into a data.frame
  -> target variable   :  19  values
  -> predict function  :  yhat.train  will be used (  default  )
  -> predicted values  :  No value for predict function target column. (  default  )
  -> model_info        :  package caret , ver. 6.0.94 , task regression (  default  )
  -> predicted values  :  numerical, min =  63.08474 , mean =  79.01364 , max =  95.3882
  -> residual function :  difference between y and yhat (  default  )
  -> residuals         :  numerical, min =  -11.05251 , mean =  -1.013642 , max =  10.35064
  A new explainer has been created!
```

```
mr_m1 <- model_residual(m1_audit)
mr_m2 <- model_residual(m2_audit)

plot_rec(mr_m1)
```




```
plot_rec(mr_m2)
```




```
score_rec(m1_audit)
```

```
rec: 3.939398
```

```
score_rec(m2_audit)
```

```
rec: 4.346172
```

Comments on REC values and Numeric Evaluation Metrics:
Based on the Numeric Evaluation Metrics, Model 1 is the best as it leads to more predictive accuracy. Based on the REC Curve and Score, model 1 has lower prediction errors meaning model 1 predicted closer to the actual numbers. However, Model 2 will be chosen for deployment because it explains significantly more variability in the dataset then model 1 based on the adjusted R^2.

Deployment

Model on entire Dataset

```r
set.seed(1)
model_full <- train(Happiness ~ Age + quad_age + log_income, data = happy_data, method = "lm",
         trControl = lm_ctrl)
summary(model_full)
```

```
Call:
lm(formula = .outcome ~ ., data = dat)

Residuals:
    Min      1Q  Median      3Q     Max 
-16.341  -4.125  -0.156   3.899  13.098 

Coefficients:
             Estimate Std. Error t value             Pr(>|t|)    
(Intercept) -13.30212   15.51251  -0.858                0.393    
Age          -2.42958    0.36196  -6.712      0.0000000013390 ***
quad_age      0.02407    0.00325   7.406      0.0000000000502 ***
log_income   12.72097    1.07381  11.847 < 0.0000000000000002 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 6.5 on 96 degrees of freedom
Multiple R-squared:  0.7001,	Adjusted R-squared:  0.6907 
F-statistic: 74.71 on 3 and 96 DF,  p-value: < 0.00000000000000022
```

Predict Scenario Happiness when Income equals $80,000 and Age equals c(30, 45, 60) years.

```r
pred_age <- data.frame(
  Age = c(30, 45, 60),
  quad_age = c(30, 45, 60)^2,
  log_income = log(80000)
)

pred_age$predicted_happiness <- predict(model_full, newdata = pred_age)
pred_age
```

```
  Age quad_age log_income predicted_happiness
1  30      900   11.28978            79.08781
2  45     2025   11.28978            69.71957
3  60     3600   11.28978            71.18154
```

Predict Scenario Happiness when Age = 60 and Income equals c($25,000, $75,000, $125,000)

```r
pred_income <- data.frame(
  Age = 60,
  quad_age = 60^2,
  log_income = log(c(25000, 75000, 125000))
)

pred_income$predicted_happiness <- predict(model_full, newdata = pred_income)
pred_income
```

```
  Age quad_age log_income predicted_happiness
1  60     3600   10.12663            56.38513
2  60     3600   11.22524            70.36054
3  60     3600   11.73607            76.85874
```

Insight to Mr. Person

In this analysis we found that both Age and Income are related to happiness. Age has a quadratic relation to happiness meaning from age 30-50.5 happiness trends downwards, but from 50.5 onwards happiness trends upwards. This can be seen when the full model is used to predict happiness based of changing ages, age 30 was the highest happiness, age 45 is the lowest level of happiness, then age 60 happiness trended upwards being higher than at age 45. Income has a positive relationship with happiness, as income increases so does happiness. This can be seen when happiness is predicted for different income amounts. $25000 had the lowest happiness, $75000 had the second highest happiness amount, and $125,000 had the highest happiness amount. If this model is deployed again the quadratic of age should be taken and log of income should also be taken as it leads to higher correlation with happiness.