EDA HW 3

Shaya Engelman

2024-03-19

```
url <- "https://raw.githubusercontent.com/Shayaeng/DATA621_Group/main/HW3/Provided%20data/crime-training
train <- read.csv(url)
dim(train)
```

```
## [1] 466  13
```

```
head(train)
```

```
##   zn indus chas   nox    rm   age    dis rad tax ptratio lstat medv target
## 1  0 19.58    0 0.605 7.929  96.2 2.0459   5 403    14.7  3.70 50.0      1
## 2  0 19.58    1 0.871 5.403 100.0 1.3216   5 403    14.7 26.82 13.4      1
## 3  0 18.10    0 0.740 6.485 100.0 1.9784  24 666    20.2 18.85 15.4      1
## 4 30  4.93    0 0.428 6.393   7.8 7.0355   6 300    16.6  5.19 23.7      0
## 5  0  2.46    0 0.488 7.155  92.2 2.7006   3 193    17.8  4.82 37.9      0
## 6  0  8.56    0 0.520 6.781  71.3 2.8561   5 384    20.9  7.67 26.5      0
```

The training dataset has 466 records (rows) of 13 different variables. All the variables are numeric, however, column 'chas' is a dummy variable.

The columns represent the following:

- zn: proportion of residential land zoned for large lots (over 25000 square feet) (predictor variable)
- indus: proportion of non-retail business acres per suburb (predictor variable)
- chas: a dummy variable for whether the suburb borders the Charles River (1) or not (0) (predictor variable)
- nox: nitrogen oxides concentration (parts per 10 million) (predictor variable)
- rm: average number of rooms per dwelling (predictor variable)
- age: proportion of owner-occupied units built prior to 1940 (predictor variable)
- dis: weighted mean of distances to five Boston employment centers (predictor variable)
- rad: index of accessibility to radial highways (predictor variable)
- tax: full-value property-tax rate per $10,000 (predictor variable)
- ptratio: pupil-teacher ratio by town (predictor variable)
- lstat: lower status of the population (percent) (predictor variable)
- medv: median value of owner-occupied homes in $1000s (predictor variable)
- target: whether the crime rate is above the median crime rate (1) or not (0) (response variable)

For some of these variables, like, 'zn, 'chas', 'tax', and 'medv', it is easy to hypothesize whether the relationship between it and the target variable would be positive or negative. For other variables, it is a bit more difficult.

First we should check for missing values:

```
sapply(train, function(x) sum(is.na(x)))
```

```
##      zn   indus    chas     nox      rm     age     dis     rad     tax ptratio
##       0       0       0       0       0       0       0       0       0       0
##   lstat    medv  target
##       0       0       0
```

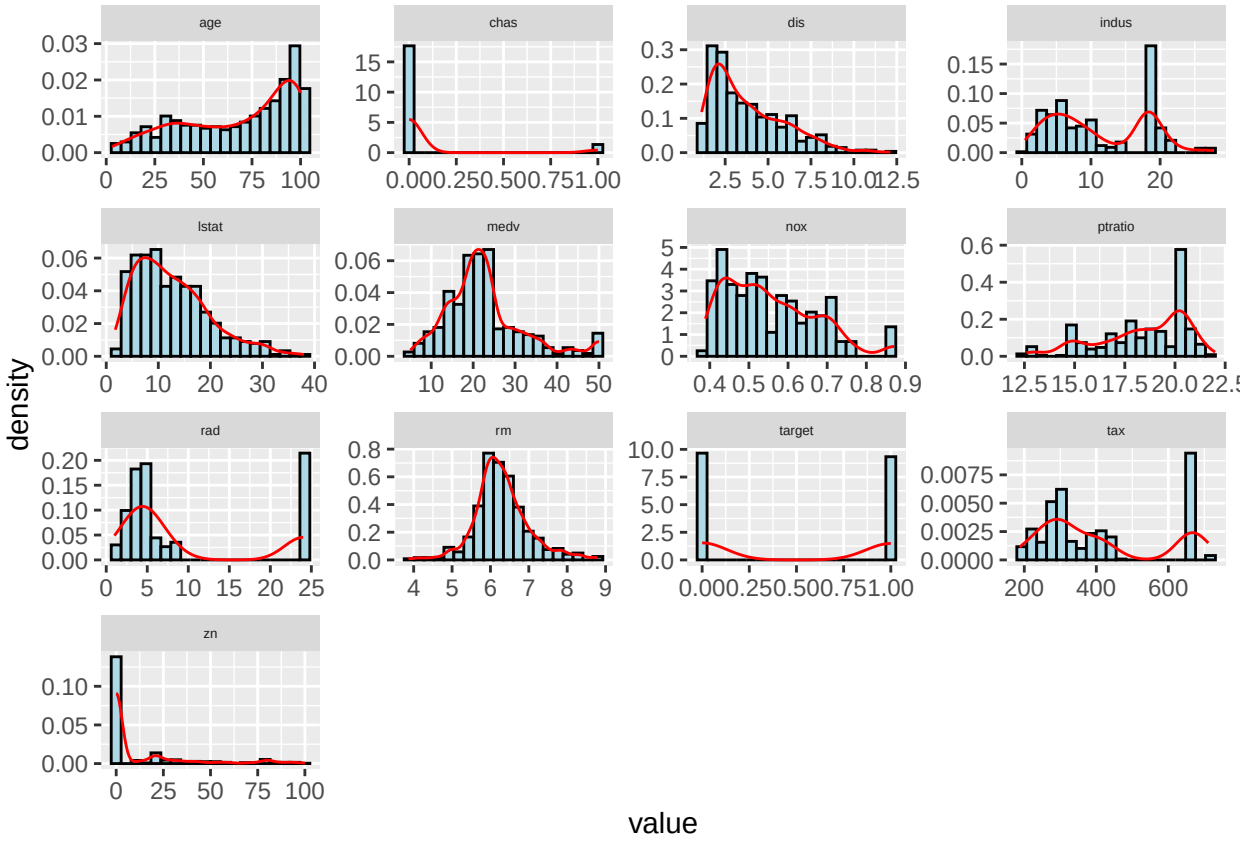
We have zero missing values. That means we should not have to impute anything.

```
descr <- round(descr(train), 2)
kable(descr)
```

	age	chas	dis	indus	lstat	medv	nox	ptratio	rad	rm	target	tax	zn
Mean	68.37	0.07	3.80	11.11	12.63	22.59	0.55	18.40	9.53	6.29	0.49	409.50	11.58
Std.Dev	28.32	0.26	2.11	6.85	7.10	9.24	0.12	2.20	8.69	0.70	0.50	167.90	23.36
Min	2.90	0.00	1.13	0.46	1.73	5.00	0.39	12.60	1.00	3.86	0.00	187.00	0.00
Q1	43.70	0.00	2.10	5.13	7.01	17.00	0.45	16.90	4.00	5.89	0.00	281.00	0.00
Median	77.15	0.00	3.19	9.69	11.35	21.20	0.54	18.90	5.00	6.21	0.00	334.50	0.00
Q3	94.10	0.00	5.21	18.10	16.94	25.00	0.62	20.20	24.00	6.63	1.00	666.00	17.50
Max	100.00	1.00	12.13	27.74	37.97	50.00	0.87	22.00	24.00	8.78	1.00	711.00	100.00
MAD	30.02	0.00	1.91	9.34	7.07	6.00	0.13	1.93	1.48	0.52	0.00	104.52	0.00
IQR	50.22	0.00	3.11	12.96	9.89	7.98	0.18	3.30	20.00	0.74	1.00	385.00	16.25
CV	0.41	3.63	0.56	0.62	0.56	0.41	0.21	0.12	0.91	0.11	1.02	0.41	2.02
Skewness	-0.58	3.34	1.00	0.29	0.91	1.08	0.75	-0.75	1.01	0.48	0.03	0.66	2.18
SE.Skewness	0.11	0.11	0.11	0.11	0.11	0.11	0.11	0.11	0.11	0.11	0.11	0.11	0.11
Kurtosis	-1.01	9.15	0.47	-1.24	0.50	1.37	-0.04	-0.40	-0.86	1.54	-2.00	-1.15	3.81
N.Valid	466.00	466.00	466.00	466.00	466.00	466.00	466.00	466.00	466.00	466.00	466.00	466.00	466.00
Pct.Valid	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00

The above table gives us a concise summary of the variable statistics. It confirms that we have no missing values in any of the variables and shows us some important insights. We see significant skew in some of the variables and those would probably need some type of transformation.We can get a better idea of the distributions and skewness using the following plots:

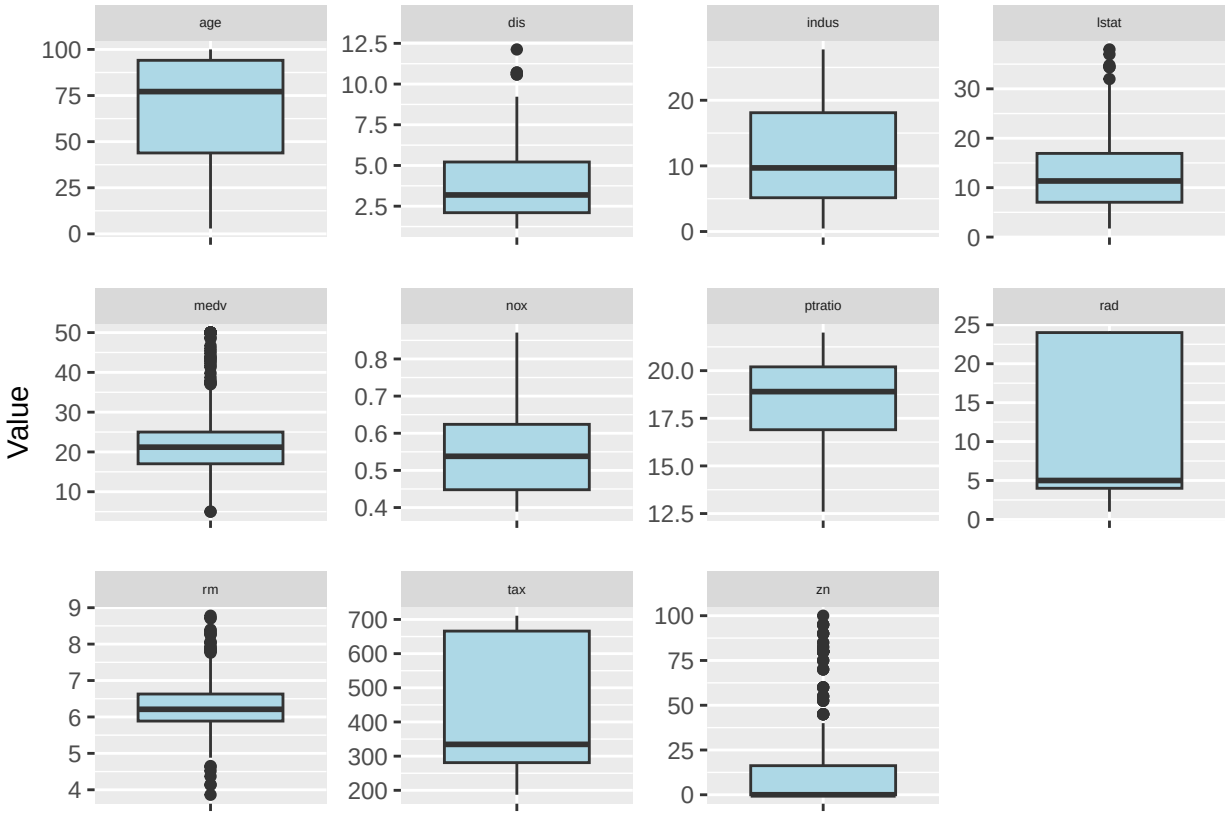
```
train |>
  gather(key = "variable", value = "value") |>
  ggplot(aes(x = value)) +
  geom_histogram(aes(y = after_stat(density)), bins = 20, fill = 'lightblue', color = 'black') +
  stat_density(geom = "line", color = "red") +
  facet_wrap(~ variable, scales = 'free') +
  theme(strip.text = element_text(size = 5))
```



The plots clearly show significant right skew, kurtosis, in 'dis', and 'lstat'. It also shows left skew in 'age' and 'pratio'. These skewed variables might be candidates for transformation. The plots also illustrate thaat 'chas' is binary and can only have a value of 0 or 1. Another interesting observation is that variables 'rad', 'tax' and possibly 'indus' appear to be bimodal. Bimodal data is when we have two or more different classes in a dataset that act as groups.

The above plots also seem to show some of the variables have wide distributions and many points above the density lines. These outliers can be visualized using boxplots:

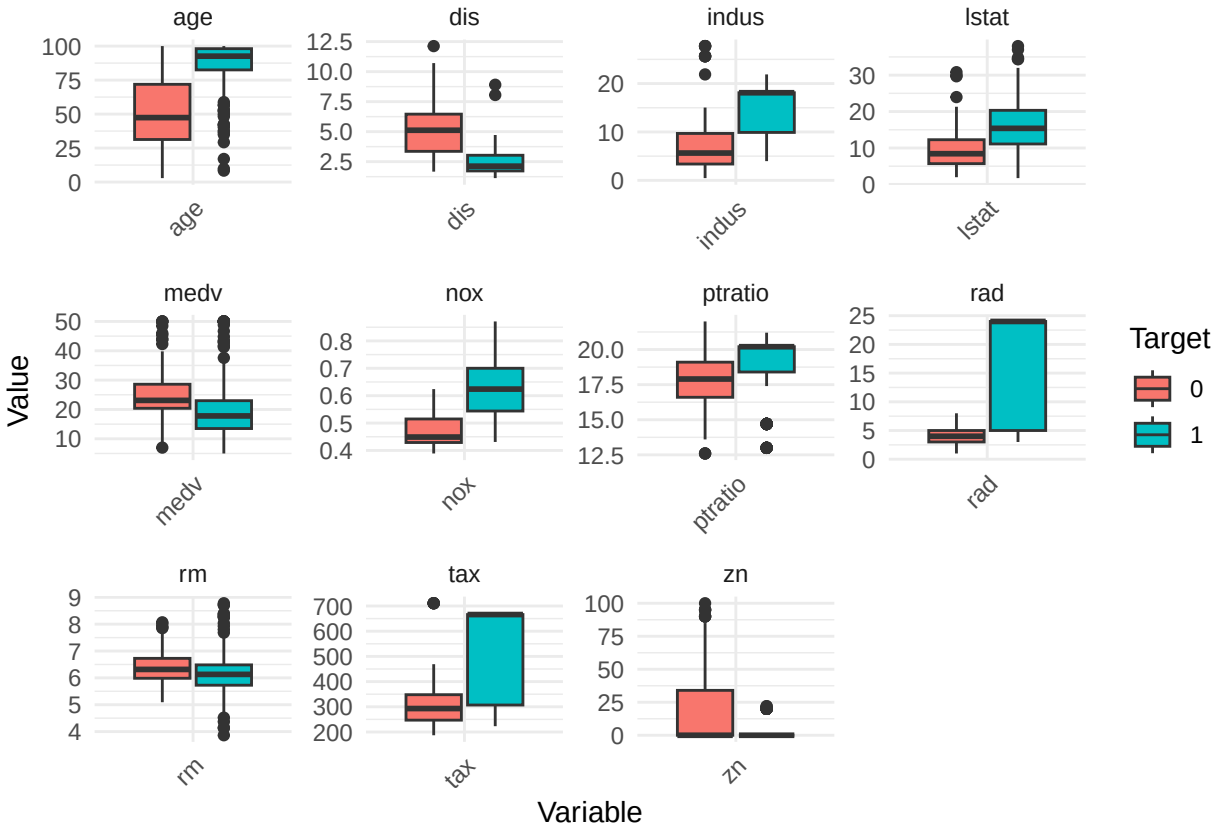
```
train |>
  select(-chas, -target) |> #drop 'chas' and 'target' since they are binary variables
  gather(key = "Variable", value = "Value") |>
  ggplot(aes(x = "", y = Value)) +
  geom_boxplot(fill = "lightblue") +
  facet_wrap(~ Variable, scales = "free") +
  labs(x = NULL, y = "Value") +
  theme(strip.text = element_text(size = 5))
```



These boxplots further confirm the skewness mentioned earlier. They also reveal that variables 'medv', 'm' and 'zn' all have a large amount of outliers. These should be investigated.

We can gain more insight by plotting the boxplots broken down by the target variable:

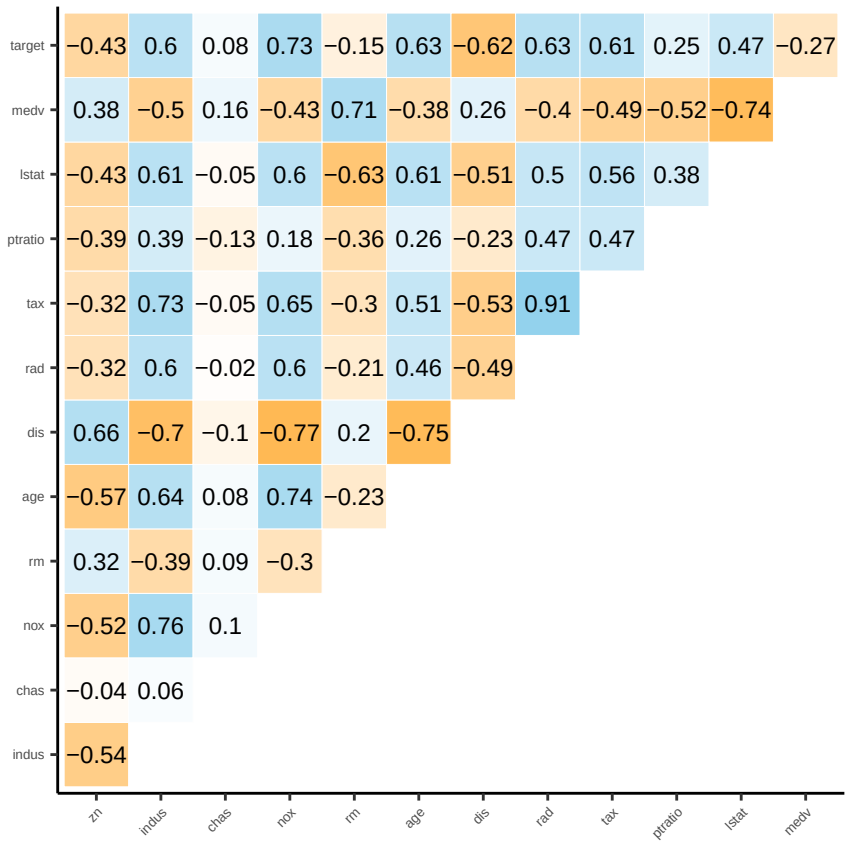
```
train |>
  select(-chas) |> #drop 'chas' since it is a dummy variable
  pivot_longer(cols = -target, names_to = "variable", values_to = "value") |>
  ggplot(aes(x = variable, y = value, fill = factor(target))) +
  geom_boxplot() +
  labs(x = "Variable", y = "Value", fill = "Target") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  facet_wrap(~variable, scales = "free")
```



We now see some of the variables have very large differences in their distributions based on the target variable. These are variables that strongly seem to be correlated with the target variable and should presumably be included in our model.

Our next step is to check the correlation between all our variables. This is for two purposes. One, to check which seem to be correlated with our target variable for inclusion in our models. Two, to check for multicollinearity between two of our predictor variables. We can use the below plot to visualize the correlations.

```
q <- cor(train)
ggcorrplot(q, type = "upper", outline.color = "white",
           ggtheme = theme_classic,
           colors = c("orange", "white", "skyblue"),
           lab = TRUE, show.legend = F, tl.cex = 5, lab_size = 3)
```



Negative Correlations with Crime Rate: Variables such as 'indus', 'nox' (nitrogen oxides concentration), 'age', 'dis' (distance to employment centers), 'rad' (accessibility to radial highways), 'tax', 'ptratio' (pupil-teacher ratio), 'lstat' (lower status of the population), and 'medv' (median value of owner-occupied homes) exhibit negative correlations with the target variable 'target', indicating that as these variables increase, the likelihood of the crime rate being above the median decreases. This suggests that areas with higher industrial presence, pollution levels, older housing stock, longer distances to employment centers, poorer accessibility to highways, higher tax rates, higher pupil-teacher ratios, lower socio-economic status, and lower median home values tend to have lower crime rates.

Positive Correlations with Crime Rate: Conversely, variables such as 'zn' (proportion of residential land zoned for large lots) and 'chas' (proximity to Charles River) exhibit positive correlations with the target variable 'target', implying that as these variables increase, the likelihood of the crime rate being above the median also increases. This suggests that areas with larger residential lots and those bordering the Charles River may experience higher crime rates.

The correlation matrix also illustrates some strong relationship between some of the predictor variables. For example, 'tax' and 'rad' have a very strong correlation of 0.91. While none of the rest of the predictor variables have anything that high there are still a few with pretty significant correlations. The following table extracts all the pairs of predictors with a correlation above 0.70 (chosen arbitrarily), these can all cause issues with collinearity and should be treated as such.

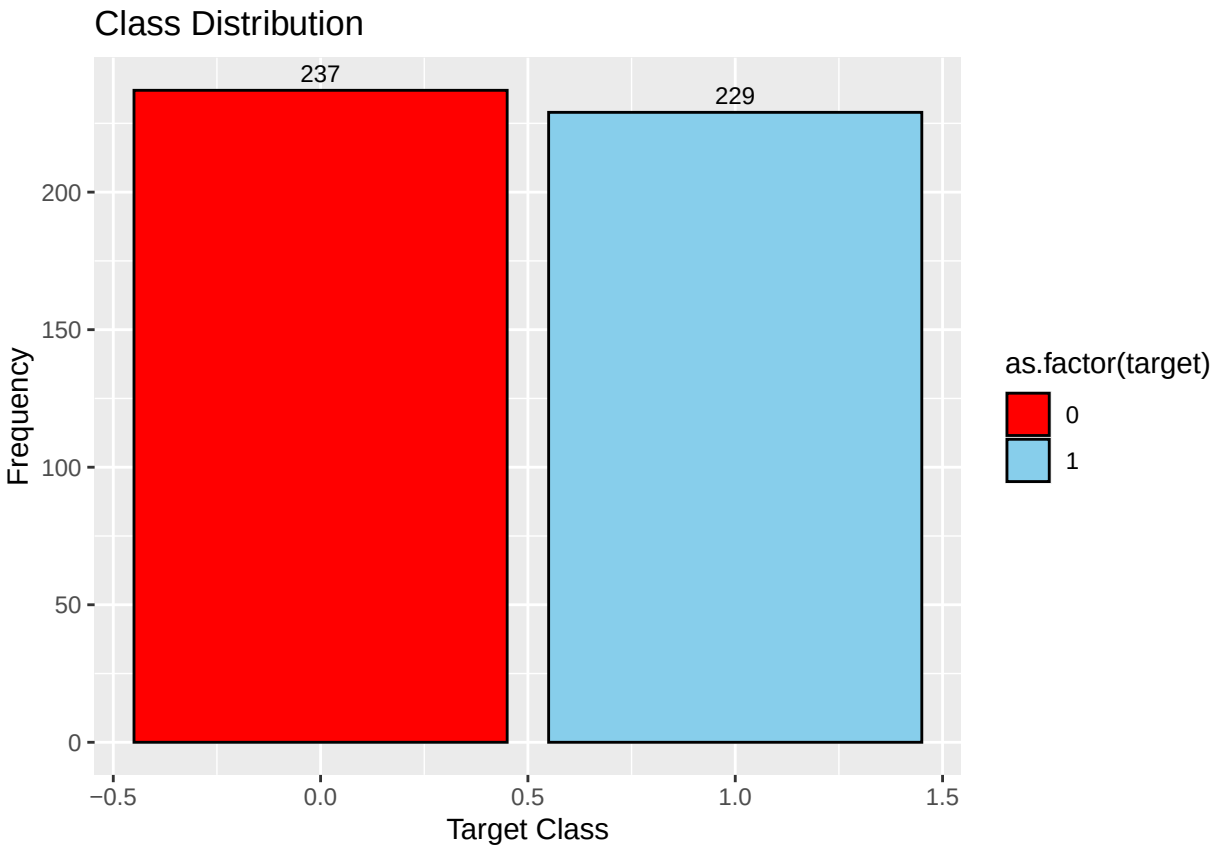
```r
# create a list of high correlation pairs
high_correlation_pairs <- list()

for (i in 1:(ncol(q) - 1)) {
  for (j in (i + 1):ncol(q)) {
    if (abs(q[i, j]) > 0.7) { # Exclude self-correlation and pairs already included
      high_correlation_pairs[[toString(c(i, j))]] <- c(rownames(q)[i], rownames(q)[j], q[i, j])
```

```
    }
  }
}

# convert the list to a data frame
high_correlation_df <- data.frame(do.call(rbind, high_correlation_pairs))
rownames(high_correlation_df) <- NULL
colnames(high_correlation_df) <- c("Variable_1", "Variable_2", "Correlation")
high_correlation_df <- high_correlation_df |>
  arrange(desc(abs(as.numeric(Correlation))))

kable(high_correlation_df)
```

Variable_1	Variable_2	Correlation
rad	tax	0.906463228913755
nox	dis	-0.768884042099931
indus	nox	0.759630083272316
age	dis	-0.750897585962367
lstat	medv	-0.735800779671144
nox	age	0.735127819507785
indus	tax	0.732229223102887
nox	target	0.726106218470473
rm	medv	0.705336793853476
indus	dis	-0.703618859912332

One last important thing to check is whether the classes of the target variable are balanced. Class imbalance can lead to misleading models. For example, if the data has an imbalance of 95%/5% success/fail rate, then predicting 100% percent of the time will be a success will result in a model successfull 95% of the time but of zero actual value to us. Since we are dealing with above or below the mean crime rate, I assume the data is balanced.

```
class_freq <- train |>
  count(target)

ggplot(train, aes(x = target, fill = as.factor(target))) +
  geom_bar(color = "black") +
  geom_text(data = class_freq, aes(label = n, y = n), vjust = -0.5, size = 3, color = "black") +
  scale_fill_manual(values = c("red", "skyblue")) +  # Customize fill colors
  labs(title = "Class Distribution",
       x = "Target Class",
       y = "Frequency")
```



The above plot shows we are working with balanaced data with 237 below mean crime rate and 229 above in our records.