Boston Dataset

1 Introduction

A widely used dataset to benchmark algorithms is the Boston housing dataset. This dataset comprises information gathered by the United States Census Bureau about housing in the Boston, Massachusetts area. It was collected from the StatLib repository (http://lib.stat.cmu.edu/datasets/boston) and has been frequently used to benchmark algorithms in the literature. With only 506 instances, the dataset is tiny.

Harrison, D., and D.L. Rubinfeld published the data in 'Hedonic prices and the demand for clean air,' J. Environ. Economics & Management, vol.5, 81-102, 1978.

The target of this dataset is to predict the median value of a home.

Figure 1: Boston Landscape (Internet images)

Dataset basic information:

Variable	Description
`MEDV` (target)	median value of owner-occupied homes in USD 1000's
`CRIM`	per capita crime rate by town
`ZN`	prop. of residential land zoned for lots over 25,000 sq.ft
`INDUS`	proportion of non-retail business acres per town
`CHAS`	Charles River dummy variable (= 1 if tract bounds river; 0 otherwise)
`NOX`	nitric oxides concentration (parts per 10 million)
`RM`	average number of rooms per dwelling
`AGE`	proportion of owner-occupied units built prior to 1940
`DIS`	weighted distances to five Boston employment centres
`RAD`	index of accessibility to radial highways
`TAX`	full-value property-tax rate per USD 10,000
`PTRATIO`	pupil-teacher ratio by town
`B`	$1000(B-0.63)^2$ where B is the prop. of blacks by town
`LSTAT`	percentage of lower status of the population

We use OpenML (R-Package) to download the dataset in a machine-readable format and convert it into a `data.frame`:

```
# load the dataset from OpenML Library
d <- OpenML::getOMLDataSet(data.id = 531)

# convert the OpenML object to a tibble (enhanced data.frame)
boston <- d %>% dplyr::as_tibble()
skimmed_boston <- skimr::skim(boston)
print(boston, width = Inf)
```

```
## # A tibble: 506 x 14
##       CRIM    ZN INDUS CHAS    NOX    RM   AGE   DIS RAD     TAX PTRATIO     B
##      <dbl> <dbl> <dbl> <fct> <dbl> <dbl> <dbl> <dbl> <fct> <dbl>   <dbl> <dbl>
##  1 0.00632  18    2.31 0     0.538  6.58  65.2  4.09 1       296    15.3  397.
##  2 0.0273    0    7.07 0     0.469  6.42  78.9  4.97 2       242    17.8  397.
##  3 0.0273    0    7.07 0     0.469  7.18  61.1  4.97 2       242    17.8  393.
##  4 0.0324    0    2.18 0     0.458  7.00  45.8  6.06 3       222    18.7  395.
##  5 0.0690    0    2.18 0     0.458  7.15  54.2  6.06 3       222    18.7  397.
##  6 0.0298    0    2.18 0     0.458  6.43  58.7  6.06 3       222    18.7  394.
##  7 0.0883   12.5  7.87 0     0.524  6.01  66.6  5.56 5       311    15.2  396.
##  8 0.145    12.5  7.87 0     0.524  6.17  96.1  5.95 5       311    15.2  397.
##  9 0.211    12.5  7.87 0     0.524  5.63 100    6.08 5       311    15.2  387.
## 10 0.170    12.5  7.87 0     0.524  6.00  85.9  6.59 5       311    15.2  387.
##    LSTAT  MEDV
##    <dbl> <dbl>
##  1  4.98  24
##  2  9.14  21.6
##  3  4.03  34.7
##  4  2.94  33.4
##  5  5.33  36.2
##  6  5.21  28.7
##  7 12.4   22.9
##  8 19.2   27.1
##  9 29.9   16.5
## 10 17.1   18.9
## # ... with 496 more rows
```

2 Exploratory Data Analysis (EDA)

In this part, we will walk through a few characteristics of Boston dataset using library `skimr` and `DataExplorer`.

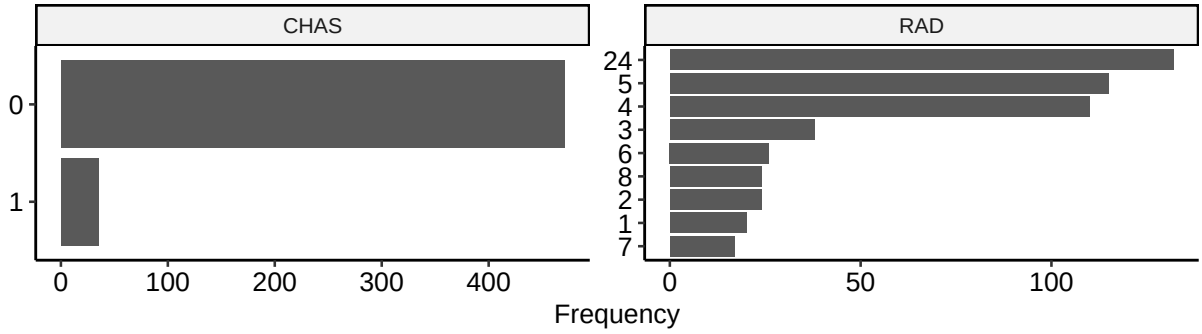
2.1 Factor variables

General statistics about factor variables from Boston dataset:

```
skimr::partition(skimmed_boston)$factor %>%
        knitr::kable(format = 'latex', booktabs = TRUE) %>%
        kableExtra::kable_styling(latex_options = 'HOLD_position')
```

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
CHAS	0	1	FALSE	2	0: 471, 1: 35
RAD	0	1	FALSE	9	24: 132, 5: 115, 4: 110, 3: 38

```
DataExplorer::plot_bar(boston, ggtheme = ggpubr::theme_pubr(base_size = 10))
```



The dataset has 2 factor variables, i.e. `CHAS` and `RAD`. The two variables don't have missing values. Looking at the discrete distributions of the two variables, we can see that they are very skewed. Feature `CHAS`'s value 0 accounts for roughly 93% of the data, this indicates that in the Boston dataset, not many homes' tracts bound the Charles River. Similarly, feature `RAD`'s values 24 & 5 & 4 account for 70% of the data.

2.2 Numerical variables

General statistics about numerical variables from Boston dataset:

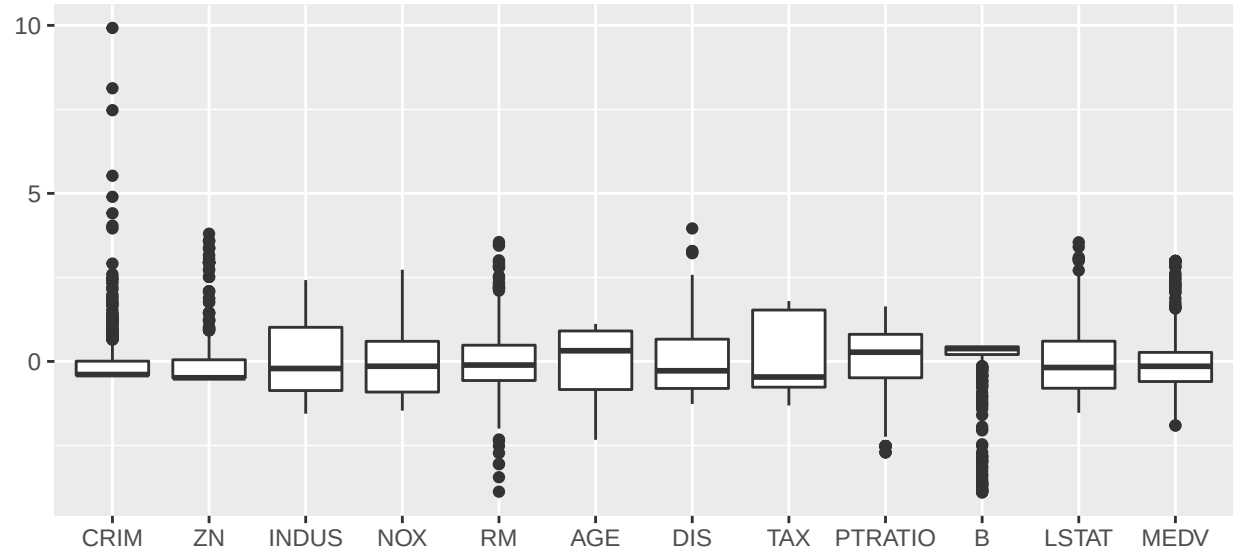
```
skimr::partition(skimmed_boston)$numeric %>%
        knitr::kable(format = 'latex', booktabs = TRUE, digits = 2) %>%
        kableExtra::kable_styling(latex_options = 'HOLD_position')
```

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
CRIM	0	1	3.61	8.60	0.01	0.08	0.26	3.68	88.98	
ZN	0	1	11.36	23.32	0.00	0.00	0.00	12.50	100.00	
INDUS	0	1	11.14	6.86	0.46	5.19	9.69	18.10	27.74	
NOX	0	1	0.55	0.12	0.38	0.45	0.54	0.62	0.87	
RM	0	1	6.28	0.70	3.56	5.89	6.21	6.62	8.78	
AGE	0	1	68.57	28.15	2.90	45.02	77.50	94.07	100.00	
DIS	0	1	3.80	2.11	1.13	2.10	3.21	5.19	12.13	
TAX	0	1	408.24	168.54	187.00	279.00	330.00	666.00	711.00	
PTRATIO	0	1	18.46	2.16	12.60	17.40	19.05	20.20	22.00	
B	0	1	356.67	91.29	0.32	375.38	391.44	396.22	396.90	
LSTAT	0	1	12.65	7.14	1.73	6.95	11.36	16.96	37.97	
MEDV	0	1	22.53	9.20	5.00	17.02	21.20	25.00	50.00	

As can be seen from the statistics, similar to the factor variables, numerical variables in this dataset don't have missing values. The ranges of values of the numerical features extremely differ from one to another.

Next we create the boxplots of (scaled) numerical variables. Due to the differences in measurements and values of the numerical features, we need to scale the features first before creating boxplots to better visualize the results, here we use standard scaling (0 mean and unit variance).

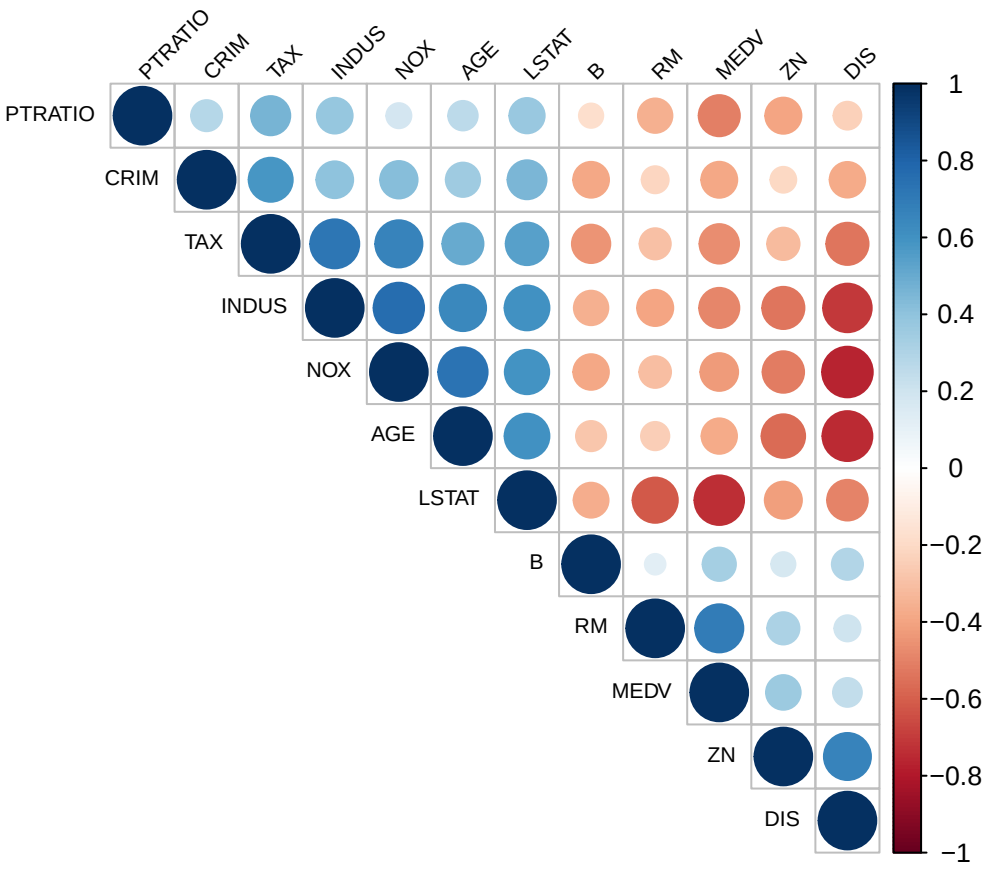
```
boston_numerical <- boston %>% select(where(is.numeric))
boston_numerical_scale <- boston_numerical %>% mutate_all(~(scale(.) %>% as.vector))
boston_numerical_scale_melt <- melt(boston_numerical_scale)
ggplot(data = boston_numerical_scale_melt, aes(x=variable, y=value)) +
        geom_boxplot() +
        theme(axis.title.x=element_blank(),
              axis.title.y=element_blank())
```



. From the boxplots, we notice that apart from the features `INDUS`, `NOX`, `AGE`, and `TAX`, the other features have a lot of outliers, especially `CRIM`, `ZN` and `B`. `CRIM` and `ZN` have right-skewed distribution and `B` has left-skewed distribution. `NOX` and `RM` are the most balanced and seem to be not skewed at all.

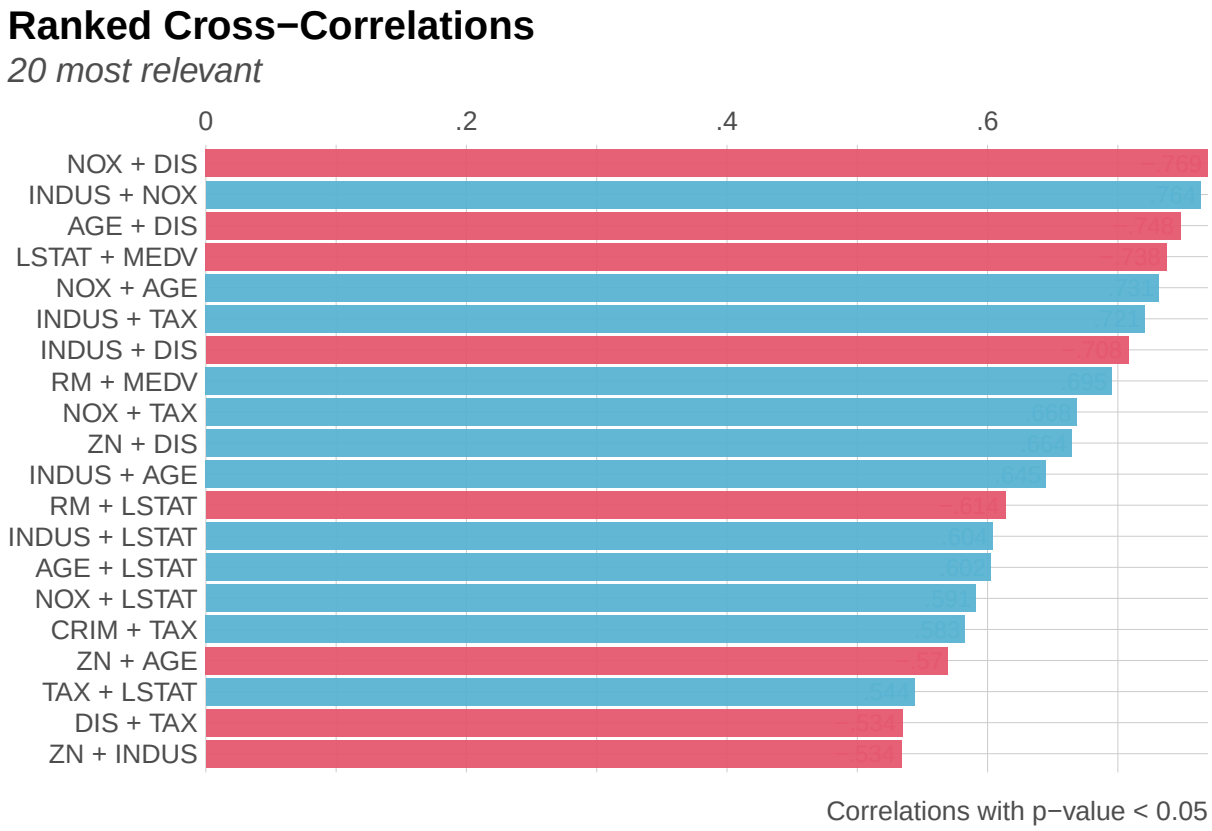
To understand more the linear relationship between the pairs of numerical variables, we create a correlation matrix:

```
boston_numerical %>%
        cor() %>%
        corrplot(
           type = "upper",
           order = "hclust",
           tl.col = "black",
           tl.srt = 45,
           tl.cex = 0.7
        )
```



We can also create a ranking of top 20 pairs of variables by the magnitude of correlation to interpret the result with `corr_cross` function from library `lares`:

```
corr_cross(boston_numerical,
  max_pvalue = 0.05, # display only significant correlations (at 5% level)
  top = 20 # display top 20 couples of variables (by correlation coefficient)
)
```



Looking at the correlation matrix and ranking, there are a few patterns that we can see. First, `DIS` has strongly negative correlation with `INDUS`, `NOX`, and `AGE`, while these three features have positive correlation with each other. This suggests that five Boston employment centers are located far from industrial areas, where the non-retail businesses are and that the employment centers have lower level of pollution and newer buildings. This might depict the evolution of the Boston metropolitan region throughout time. The positive correlation between the three features `INDUS`, `NOX`, and `AGE` also indicate that industrial areas have a lot of older buildings and are associated with pollution. Furthermore, `DIS` is positively associated with `ZN`, which implies that proportion of residential area zoned for large lots are concentrated near the Boston centers. Moreover, taking a look at our target variable `MEDV`, we can see that it has highly negative correlation with `LSTAT` and positive correlation with `RM`. It is not surprising if the area having high housing price is not an ideal place for low incomers. Another unsurprising thing is that the number of rooms per home is one possible metric of housing size, so we also expect that it can be a positive indicator for the housing prices.

Next, we begin with the data preprocessing notes.

3 Data preprocessing notes

In this section, we present a few notes that can be beneficial for preprocessing the data.

3.1 Data quality assessment

From the EDA, we can see that this dataset is clean with no missing data, mismatched data types. The fact that the measurements are different across numerical features (much different ranges of values) and that there are a lot of highly skewed features indicate the need for data transformation.

3.2 Data cleaning

It can be seen from the EDA's boxplots of the numerical features that there are a lot of outliers for most of them. However, handling outliers needs to be taken with care. Do those outliers exist because of some errors in measurements? Or do they just represent natural variations in the true population? In the case of this dataset, a few features like `CRIM`, `ZN` or `DIS` have highly right skewed distribution, which can be the cause of outliers.

3.3 Data transformation

Because the majority of the numerical variables are asymmetric, with a larger density on the left side, it may be helpful to some models to perform some transformations. `log` or `exp` (to the power of `x < 1`) transformation can be used as it can potentially reduce the skewness of the original data and also mitigate the problem with outlier. Below is an example with feature `DIS`.

```
boston_numerical <- boston_numerical %>% mutate(DIS_LOG = log(DIS))
DataExplorer::plot_histogram(boston_numerical %>% select(DIS, DIS_LOG))
```

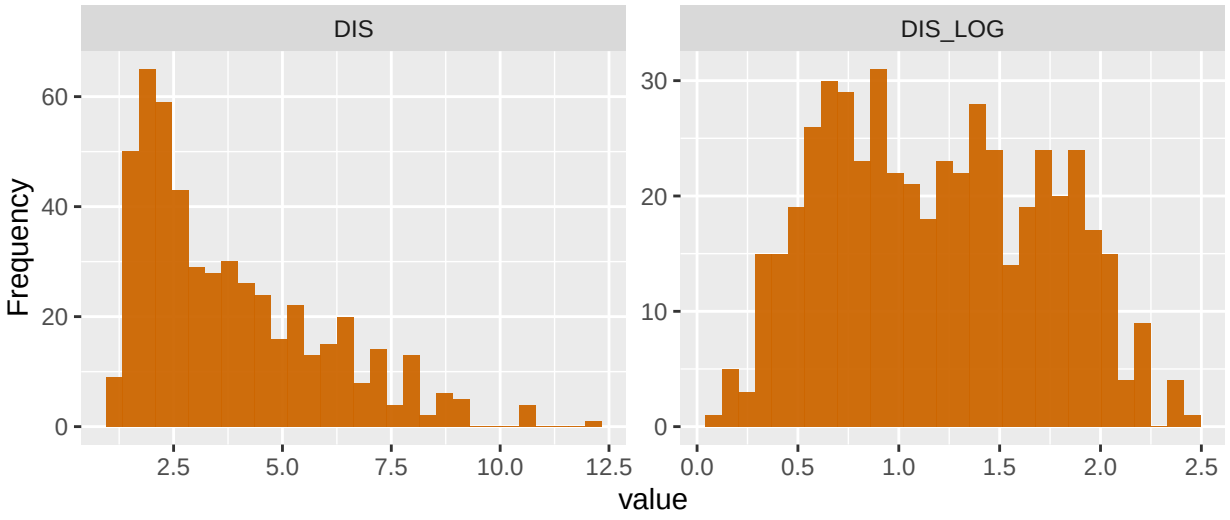


Figure 2: Histogram of feature `DIS` before (left) and after (right) `log` transformation