DS621 - Homework 3

George Cruz Deschamps[1], Karim Hammoud[1], Maliat Islam[1], Matthew Lucich[1], Gabriella Martinez[1], Ken Popkin[1]

Abstract

In this homework assignment, we will explore, analyze and model a data set containing information on crime for various neighborhoods of a major city. Each record has a response variable indicating whether or not the crime rate is above the median crime rate (1) or not (0).

Our objective is to build three different binary logistic regression models on the training data set to predict whether or not neighborhoods are at risk for high crime. We will provide classifications and probabilities for the evaluation data set using your binary logistic regression model. We can only use the variables given (or variables derived from the variables provided). Below is a short description of the variables of interest in the data set:

- `zn` proportion 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` nitrogen 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 centers
- `rad` index of accessibility to radial highways
- `tax` full-value property-tax rate per $10,000
- `ptratio` pupil-teacher ratio by town
- `lstat` % lower status of the population
- `medv` median value of owner-occupied homes in $1000's
- `target` indicating whether or not the crime rate is above the median crime rate (1) or not (0) **response variable**

Email addresses: `georg4re@gmail.com` (George Cruz Deschamps), `cunykarim@gmail.com` (Karim Hammoud), `maliat.islam21@gmail.com` (Maliat Islam), `matt.lucich@gmail.com` (Matthew Lucich), `gpmmrtzz@gmail.com` (Gabriella Martinez), `krpopkin@gmail.com` (Ken Popkin)

The Data

Data Exploration. Exploratory data analysis is the process to get to know your data, so that a hypothesis can be generated and later tested. Visualization techniques are usually applied to aid the exploration of the data.

To get introduced to the dataset, we use DataExplorer's `introduce()` function:

rows	466
columns	13
discrete_columns	0
continuous_columns	13
all_missing_columns	0
total_missing_values	0
complete_rows	466
total_observations	6,058
memory_usage	44,440

Using dplyr's `glimpse()`[1] function, we can take a "glimpse" into both the `crime_train` and `crime_test` data respectively, and easily see the dimensions, variable names and types.

From this `glimpse()` into the `crime_train` dataset, we confirm the two variables `chas` and `target` are factors[2] as noted in the description of variables above. These variables will be transformed in our data preparation stage.

```
## Rows: 466
## Columns: 13
## $ zn      <dbl> 0, 0, 0, 30, 0, 0, 0, 0, 0, 80, 22, 0, 0, 22, 0, 0, 100, 20, 0~
## $ indus   <dbl> 19.58, 19.58, 18.10, 4.93, 2.46, 8.56, 18.10, 18.10, 5.19, 3.6~
## $ chas    <int> 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ nox     <dbl> 0.605, 0.871, 0.740, 0.428, 0.488, 0.520, 0.693, 0.693, 0.515,~
## $ rm      <dbl> 7.929, 5.403, 6.485, 6.393, 7.155, 6.781, 5.453, 4.519, 6.316,~
## $ age     <dbl> 96.2, 100.0, 100.0, 7.8, 92.2, 71.3, 100.0, 100.0, 38.1, 19.1,~
## $ dis     <dbl> 2.0459, 1.3216, 1.9784, 7.0355, 2.7006, 2.8561, 1.4896, 1.6582~
## $ rad     <int> 5, 5, 24, 6, 3, 5, 24, 24, 5, 1, 7, 5, 24, 7, 3, 3, 5, 5, 24, ~
## $ tax     <int> 403, 403, 666, 300, 193, 384, 666, 666, 224, 315, 330, 398, 66~
## $ ptratio <dbl> 14.7, 14.7, 20.2, 16.6, 17.8, 20.9, 20.2, 20.2, 20.2, 16.4, 19~
## $ lstat   <dbl> 3.70, 26.82, 18.85, 5.19, 4.82, 7.67, 30.59, 36.98, 5.68, 9.25~
## $ medv    <dbl> 50.0, 13.4, 15.4, 23.7, 37.9, 26.5, 5.0, 7.0, 22.2, 20.9, 24.8~
## $ target  <int> 1, 1, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1, 1, 1, 0,~
```

```
## Rows: 40
## Columns: 12
## $ zn      <int> 0, 0, 0, 0, 0, 25, 25, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 22, 90~
## $ indus   <dbl> 7.07, 8.14, 8.14, 8.14, 5.96, 5.13, 5.13, 4.49, 4.49, 2.89, 25~
## $ chas    <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0,~
## $ nox     <dbl> 0.469, 0.538, 0.538, 0.538, 0.499, 0.453, 0.453, 0.449, 0.449,~
## $ rm      <dbl> 7.185, 6.096, 6.495, 5.950, 5.850, 5.741, 5.966, 6.630, 6.121,~
## $ age     <dbl> 61.1, 84.5, 94.4, 82.0, 41.5, 66.2, 93.4, 56.1, 56.8, 69.6, 97~
## $ dis     <dbl> 4.9671, 4.4619, 4.4547, 3.9900, 3.9342, 7.2254, 6.8185, 4.4377~
## $ rad     <int> 2, 4, 4, 4, 5, 8, 8, 3, 3, 2, 2, 2, 4, 5, 5, 4, 8, 8, 7, 1, 1,~
## $ tax     <int> 242, 307, 307, 307, 279, 284, 284, 247, 247, 276, 188, 188, 43~
```

[1] https://www.rdocumentation.org/packages/dplyr/versions/0.3/topics/glimpse

[2] https://www.stat.berkeley.edu/~s133/factors.html#

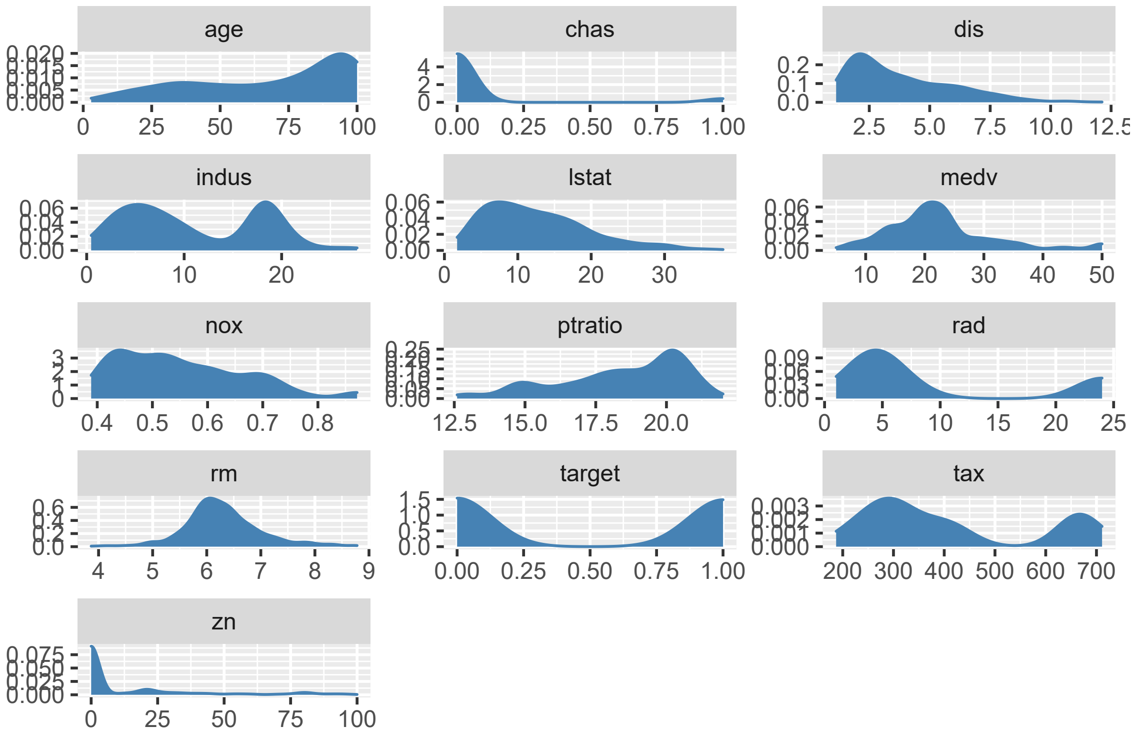
Table 1: Univariate Descriptive Statistics - Training Data Set

	Mean	Std.Dev	Min	Q1	Median	Q3	Max
age	68.3675966	28.3213784	2.9000	43.7000	77.15000	94.1000	100.0000
chas	0.0708155	0.2567920	0.0000	0.0000	0.00000	0.0000	1.0000
dis	3.7956929	2.1069496	1.1296	2.1007	3.19095	5.2146	12.1265
indus	11.1050215	6.8458549	0.4600	5.1300	9.69000	18.1000	27.7400
lstat	12.6314592	7.1018907	1.7300	7.0100	11.35000	16.9400	37.9700
medv	22.5892704	9.2396814	5.0000	17.0000	21.20000	25.0000	50.0000
nox	0.5543105	0.1166667	0.3890	0.4480	0.53800	0.6240	0.8710
ptratio	18.3984979	2.1968447	12.6000	16.9000	18.90000	20.2000	22.0000
rad	9.5300429	8.6859272	1.0000	4.0000	5.00000	24.0000	24.0000
rm	6.2906738	0.7048513	3.8630	5.8870	6.21000	6.6300	8.7800
target	0.4914163	0.5004636	0.0000	0.0000	0.00000	1.0000	1.0000
tax	409.5021459	167.9000887	187.0000	281.0000	334.50000	666.0000	711.0000
zn	11.5772532	23.3646511	0.0000	0.0000	0.00000	17.5000	100.0000

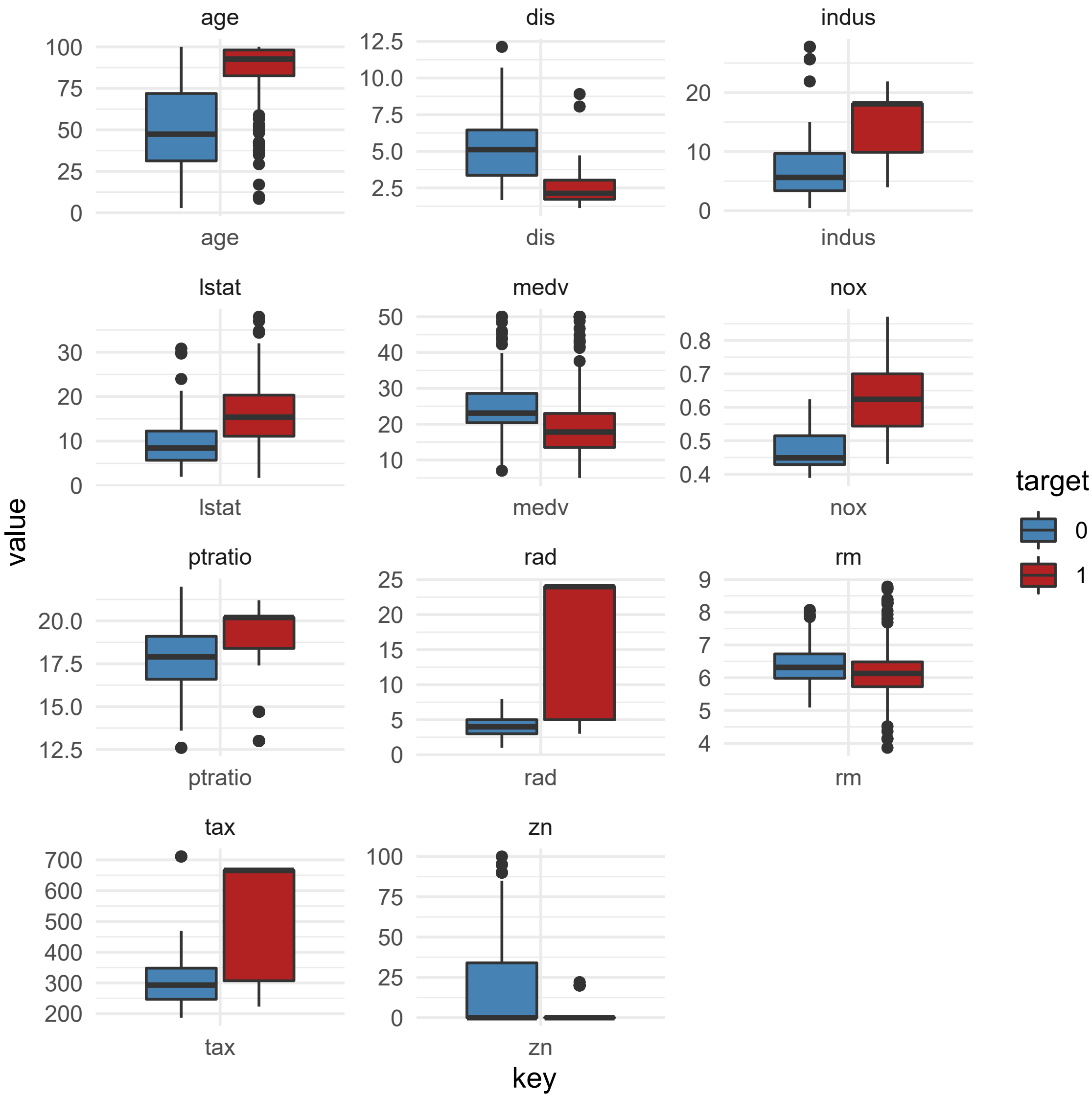
```
## $ ptratio <dbl> 17.8, 21.0, 21.0, 21.0, 19.2, 19.7, 19.7, 18.5, 18.5, 18.0, 19~
## $ lstat   <dbl> 4.03, 10.26, 12.80, 27.71, 8.77, 13.15, 14.44, 6.53, 8.44, 11.~
## $ medv    <dbl> 34.7, 18.2, 18.4, 13.2, 21.0, 18.7, 16.0, 26.6, 22.2, 21.4, 17~
```

Next, we proceed with our exploratory data analysis by providing univariate descriptive statistics on our training dataset, `crime_train`.

Furthermore, in the histogram plot below, we see that `medv`, and `rm` are normally distributed. We also note bi-modal distribution of the variables `indus`, `rad` and `tax`. The rest of the variables show moderate to high skewness on either side respectively.



In the box-plot figure below, we see many variables exhibit outliers We also see very high interquartile range for `rad` and `tax` variables where crime rate is above the median. Lastly, the variance between the 2 values of target differs for `zn`, `nox`, `age`, `dis`, `rad` & `tax`, which indicates that we will want to consider adding quadratic terms for them.

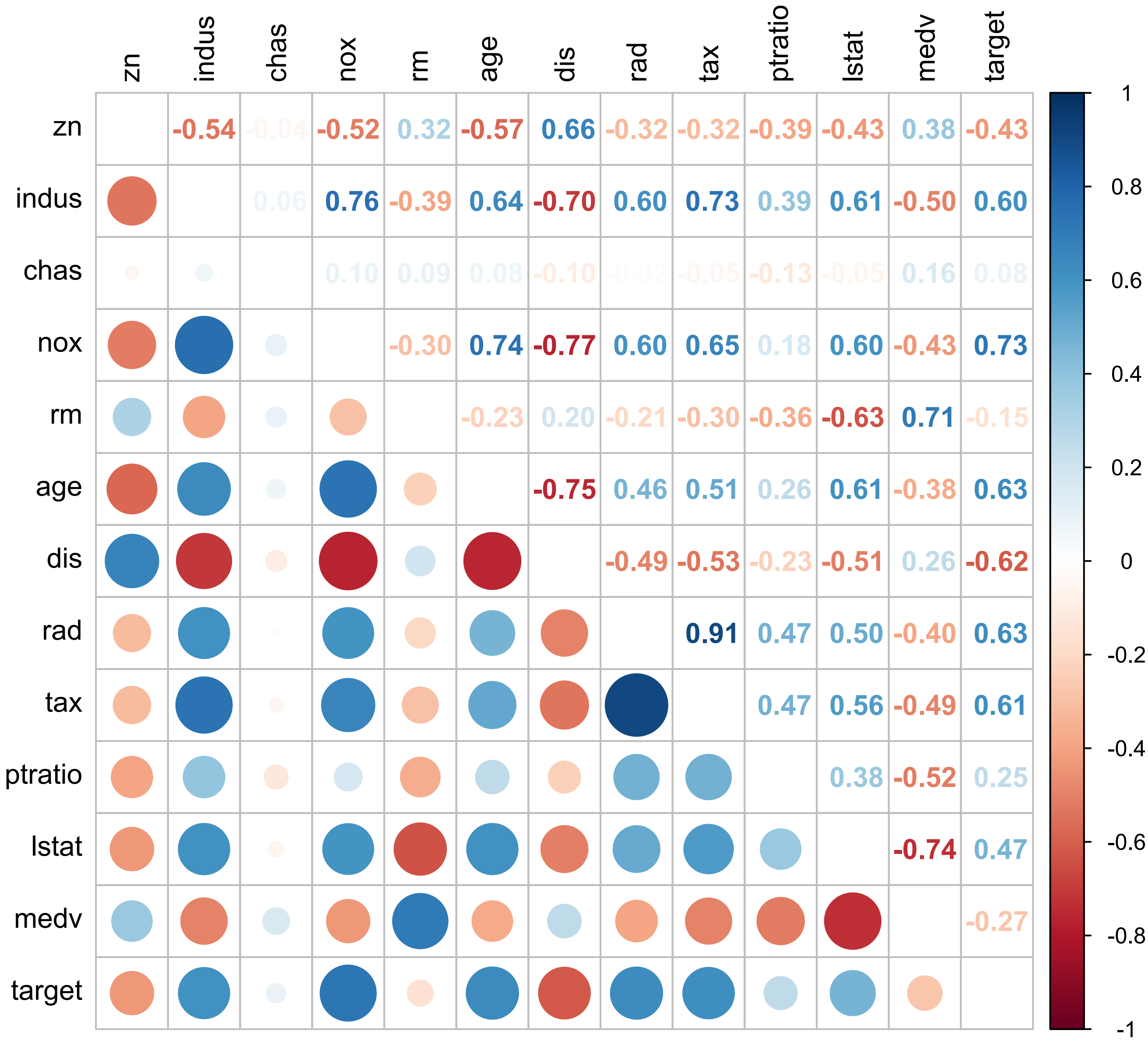


In order to investigate if there is existing correlation between the data and the target variable, we obtain the values of correlation as well as a visualization.

The correlation table and plot below, we see moderate positive correlation between variables `nox`, `age`, `rad`, `tax`, `indus` and `target` variables; and moderate negative correlation between variable dis. And the rest of the variables have weak or no correlations.

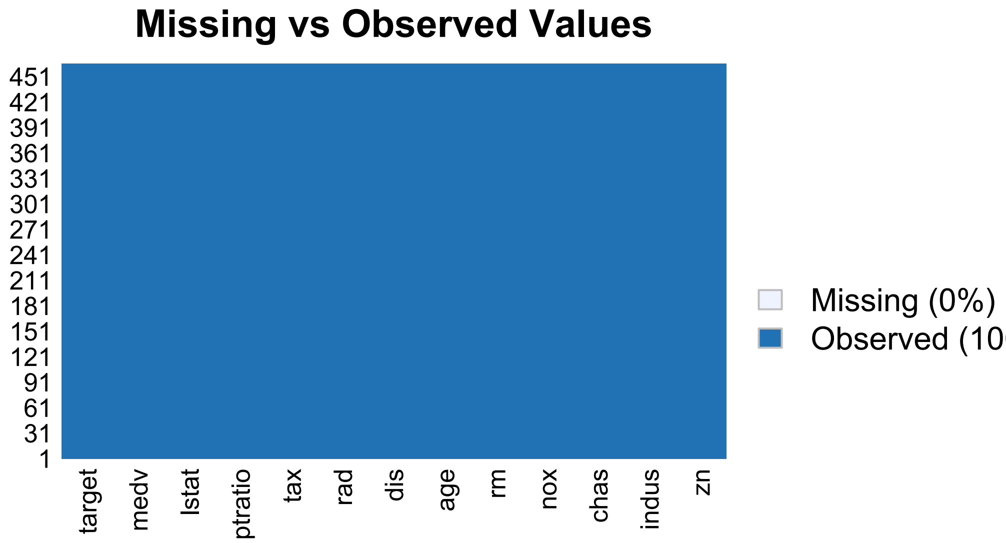
	Correlation
target	1.0000000
nox	0.7261062
age	0.6301062
rad	0.6281049
tax	0.6111133
indus	0.6048507
lstat	0.4691270
ptratio	0.2508489
chas	0.0800419
rm	-0.1525533
medv	-0.2705507
zn	-0.4316818
dis	-0.6186731

Below is a correlation matrix of the feature variables in our dataset. The correlation matrix confirms that multicollinearity is a concern.


zn indus chas nox rm age dis rad tax ptratio lstat medv target
zn -0.54 -0.04 -0.52 0.32 -0.57 0.66 -0.32 -0.32 -0.39 -0.43 0.38 -0.43
indus 0.06 0.76 -0.39 0.64 -0.70 0.60 0.73 0.39 0.61 -0.50 0.60
chas 0.10 0.09 0.08 -0.10 -0.02 -0.05 -0.13 -0.05 0.16 0.08
nox -0.30 0.74 -0.77 0.60 0.65 0.18 0.60 -0.43 0.73
rm -0.23 0.20 -0.21 -0.30 -0.36 -0.63 0.71 -0.15
age -0.75 0.46 0.51 0.26 0.61 -0.38 0.63
dis -0.49 -0.53 -0.23 -0.51 0.26 -0.62
rad 0.91 0.47 0.50 -0.40 0.63
tax 0.47 0.56 -0.49 0.61
ptratio 0.38 -0.52 0.25
lstat -0.74 0.47
medv -0.27
target
1 0.8 0.6 0.4 0.2 0 -0.2 -0.4 -0.6 -0.8 -1

Lastly, we proceed to check if there are any missing data points in the `crime_train`\

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



Data Preparation

Feature Engineering

In an effort to fine tune our model, we will introduce the use of feature engineering on select variables.

- `ptratio_indicator` assigned a value of 1 if the pupil to teacher ratio is > 16, 0 if `ptratio` is greater than 16[3]

- `lstat_indicator` assigned a value of 1 if > 15 % of the population is considered low status, 0 otherwise
- `dis_indicator` assigned a value of 1 if the distance from employment centers is > 4, 0 if `dis` is less than 4 (mean value: 3.8)

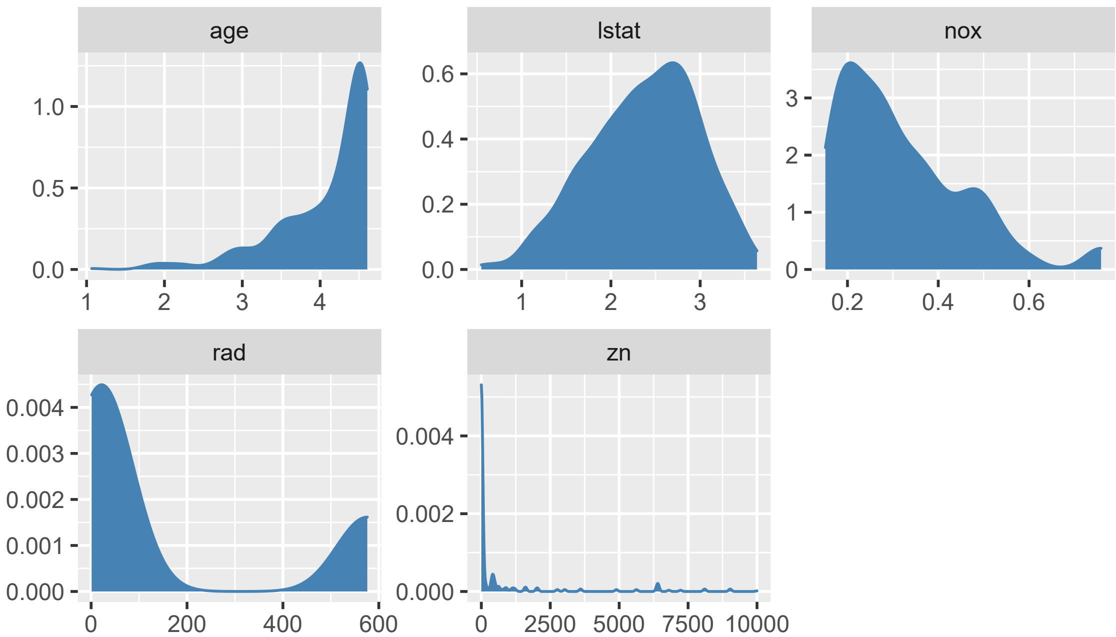
```
## 'data.frame':    466 obs. of  17 variables:
##  $ zn                 : num  0 0 0 30 0 0 0 0 0 80 ...
##  $ indus              : num  19.58 19.58 18.1 4.93 2.46 ...
##  $ chas               : Factor w/ 2 levels "0","1": 1 2 1 1 1 1 1 1 1 1 ...
##  $ nox                : num  0.605 0.871 0.74 0.428 0.488 0.52 0.693 0.693 0.515 0.392 ...
##  $ rm                 : num  7.93 5.4 6.49 6.39 7.16 ...
##  $ age                : num  96.2 100 100 7.8 92.2 71.3 100 100 38.1 19.1 ...
##  $ dis                : num  2.05 1.32 1.98 7.04 2.7 ...
##  $ rad                : int  5 5 24 6 3 5 24 24 5 1 ...
##  $ tax                : int  403 403 666 300 193 384 666 666 224 315 ...
##  $ ptratio            : num  14.7 14.7 20.2 16.6 17.8 20.9 20.2 20.2 20.2 16.4 ...
##  $ lstat              : num  3.7 26.82 18.85 5.19 4.82 ...
##  $ medv               : num  50 13.4 15.4 23.7 37.9 26.5 5 7 22.2 20.9 ...
##  $ target             : Factor w/ 2 levels "0","1": 2 2 2 1 1 1 2 2 1 1 ...
##  $ ptratio_indicator  : Factor w/ 2 levels "0","1": 2 2 1 1 1 1 1 1 1 1 ...
##  $ lstat_indicator    : Factor w/ 2 levels "0","1": 2 1 1 2 2 2 1 1 2 2 ...
##  $ dis_indicator      : Factor w/ 2 levels "0","1": 2 2 2 1 2 2 2 2 1 1 ...
##  $ age_greater_than_77: Factor w/ 2 levels "0","1": 2 2 2 1 2 1 2 2 1 1 ...
```

[3] https://www.publicschoolreview.com/average-student-teacher-ratio-stats/national-data

Data Transformation

Some of the variables are skewed, have outliers or follow a bi-modal distribution. Therefore, we will perform transformation on some of these variables. First, we will remove the variable `tax` due to multi-collinearity and it's high VIF score. Next, we will take log() transformation of `age` and `lstat` variables to lower skewness. Lastly, we will add quadratic term to `zn`, `rad`, and `nox` variables to account for its variances with respect to target variable.

	VIF Score
zn	2.324259
indus	4.120699
chas	1.090265
nox	4.505049
rm	2.354788
age	3.134015
dis	4.240618
rad	6.781354
tax	9.217228
ptratio	2.013109
lstat	3.649059
medv	3.667370




```
## 'data.frame':    466 obs. of  16 variables:
##  $ zn               : num  0 0 0 900 0 0 0 0 0 6400 ...
##  $ indus            : num  19.58 19.58 18.1 4.93 2.46 ...
##  $ chas             : Factor w/ 2 levels "0","1": 1 2 1 1 1 1 1 1 1 1 ...
##  $ nox              : 'AsIs' num  0.366025 0.758641   0.5476 0.183184 0.238144 ...
##  $ rm               : num  7.93 5.4 6.49 6.39 7.16 ...
##  $ age              : num  4.57 4.61 4.61 2.05 4.52 ...
##  $ dis              : num  2.05 1.32 1.98 7.04 2.7 ...
```

```
##  $ rad                 : num  25 25 576 36 9 25 576 576 25 1 ...
##  $ ptratio             : num  14.7 14.7 20.2 16.6 17.8 20.9 20.2 20.2 20.2 16.4 ...
##  $ lstat               : num  1.31 3.29 2.94 1.65 1.57 ...
##  $ medv                : num  50 13.4 15.4 23.7 37.9 26.5 5 7 22.2 20.9 ...
##  $ target              : Factor w/ 2 levels "0","1": 2 2 2 1 1 1 2 2 1 1 ...
##  $ ptratio_indicator   : Factor w/ 2 levels "0","1": 2 2 1 1 1 1 1 1 1 1 ...
##  $ lstat_indicator     : Factor w/ 2 levels "0","1": 2 1 1 2 2 2 1 1 2 2 ...
##  $ dis_indicator       : Factor w/ 2 levels "0","1": 2 2 2 1 2 2 2 2 1 1 ...
##  $ age_greater_than_77: Factor w/ 2 levels "0","1": 2 2 2 1 2 1 2 2 1 1 ...
```

Building the Models

First, we begin by building the null binary regression model. We will use this model to compare it to the subsequent models we build.

Coefficients for the null or Intercept only model:

```
##
## Call:
## glm(formula = target ~ 1, family = binomial(link = "logit"),
##     data = train)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.163  -1.163  -1.163   1.192   1.192
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.03434    0.09266  -0.371    0.711
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 645.88  on 465  degrees of freedom
## Residual deviance: 645.88  on 465  degrees of freedom
## AIC: 647.88
##
## Number of Fisher Scoring iterations: 3
```

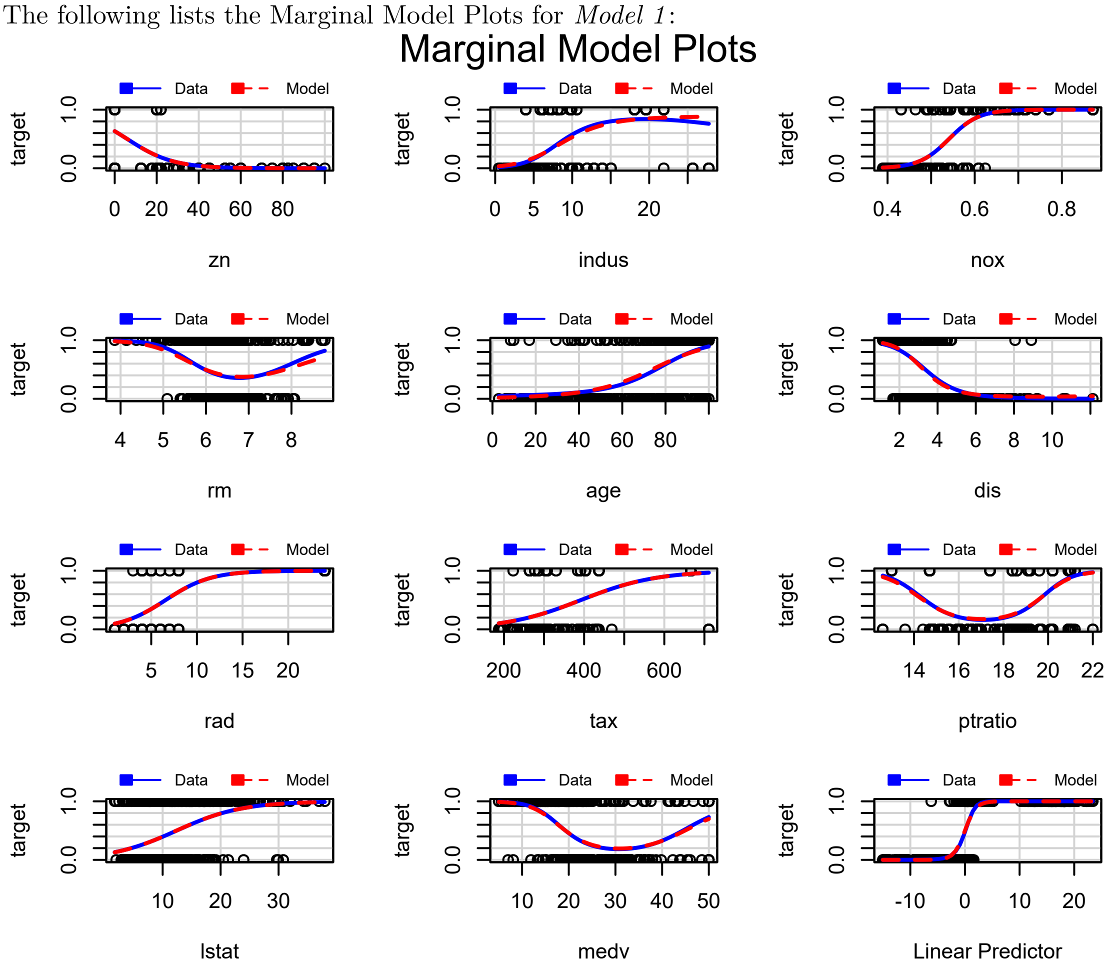
First Model

Next we proceed to make our first model using the both the original, untransformed variables and the engineered features.

```
##
## Call:
## glm(formula = target ~ ., family = binomial(link = "logit"),
##     data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8917  -0.1328  -0.0010   0.0024   3.5163
##
## Coefficients:
##                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)          -42.873132   7.814964  -5.486 4.11e-08 ***
## zn                    -0.078825   0.038257  -2.060 0.039360 *
## indus                 -0.062726   0.049280  -1.273 0.203072
## chas1                  1.059049   0.763475   1.387 0.165398
## nox                   44.583700   8.718567   5.114 3.16e-07 ***
## rm                    -0.567530   0.745600  -0.761 0.446555
## age                    0.019072   0.018456   1.033 0.301411
## dis                    0.666568   0.327609   2.035 0.041887 *
## rad                    0.745172   0.177312   4.203 2.64e-05 ***
## tax                   -0.007360   0.003288  -2.238 0.025213 *
```

```
## ptratio                 0.626502   0.188187   3.329 0.000871 ***
## lstat                   0.078997   0.075812   1.042 0.297405
## medv                    0.181089   0.071225   2.543 0.011006 *
## ptratio_indicator1      1.803413   1.156048   1.560 0.118764
## lstat_indicator1        0.492508   0.686231   0.718 0.472942
## dis_indicator1          0.296323   0.734391   0.403 0.686584
## age_greater_than_771    0.655696   0.683259   0.960 0.337226
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 645.88  on 465  degrees of freedom
## Residual deviance: 187.56  on 449  degrees of freedom
## AIC: 221.56
##
## Number of Fisher Scoring iterations: 9
```

The following lists the Marginal Model Plots for *Model 1*:



Although the plot appears to hint at a good fit, certain variables seem to have many outliers which hint at weaker relationships. Several of these are cubic or quadratic and two of these variables(the *proportion of landzones* and *distance to employment centers*) have a negative relationship.

Next, we take a look at the confidence intervals for the regression coefficients:

```
##                             2.5 %       97.5 %
## (Intercept)          -58.19018108 -27.556083221
## zn                    -0.15380693  -0.003842606
## indus                 -0.15931427   0.033861437
## chas1                 -0.43733469   2.555432086
## nox                   27.49562301  61.671777219
## rm                    -2.02887961   0.893820505
## age                   -0.01710002   0.055244413
## dis                    0.02446530   1.308670937
## rad                    0.39764649   1.092698161
## tax                   -0.01380474  -0.000914667
## ptratio                0.25766289   0.995341208
## lstat                 -0.06959190   0.227586891
## medv                   0.04149092   0.320686330
## ptratio_indicator1    -0.46239913   4.069225818
## lstat_indicator1      -0.85247956   1.837495043
## dis_indicator1        -1.14305613   1.735703034
## age_greater_than_771  -0.68346653   1.994858485
```

```
##                                OR        2.5 %       97.5 %
## (Intercept)          2.401238e-19 5.349651e-26 1.077817e-12
## zn                   9.242019e-01 8.574376e-01 9.961648e-01
## indus                9.392004e-01 8.527283e-01 1.034441e+00
## chas1                2.883626e+00 6.457553e-01 1.287686e+01
## nox                  2.303854e+19 8.733682e+11 6.077326e+26
## rm                   5.669243e-01 1.314828e-01 2.444451e+00
## age                  1.019255e+00 9.830454e-01 1.056799e+00
## dis                  1.947542e+00 1.024767e+00 3.701251e+00
## rad                  2.106804e+00 1.488318e+00 2.982310e+00
## tax                  9.926673e-01 9.862901e-01 9.990858e-01
## ptratio              1.871054e+00 1.293903e+00 2.705647e+00
## lstat                1.082202e+00 9.327744e-01 1.255567e+00
## medv                 1.198521e+00 1.042364e+00 1.378073e+00
## ptratio_indicator1   6.070332e+00 6.297709e-01 5.851165e+01
## lstat_indicator1     1.636415e+00 4.263564e-01 6.280785e+00
## dis_indicator1       1.344905e+00 3.188431e-01 5.672915e+00
## age_greater_than_771 1.926483e+00 5.048638e-01 7.351163e+00
```

The odds ratio would indicate the multiplicative change in odds of crime for every one unit increase on a predictor variable.

Odds-ratios for coefficients:

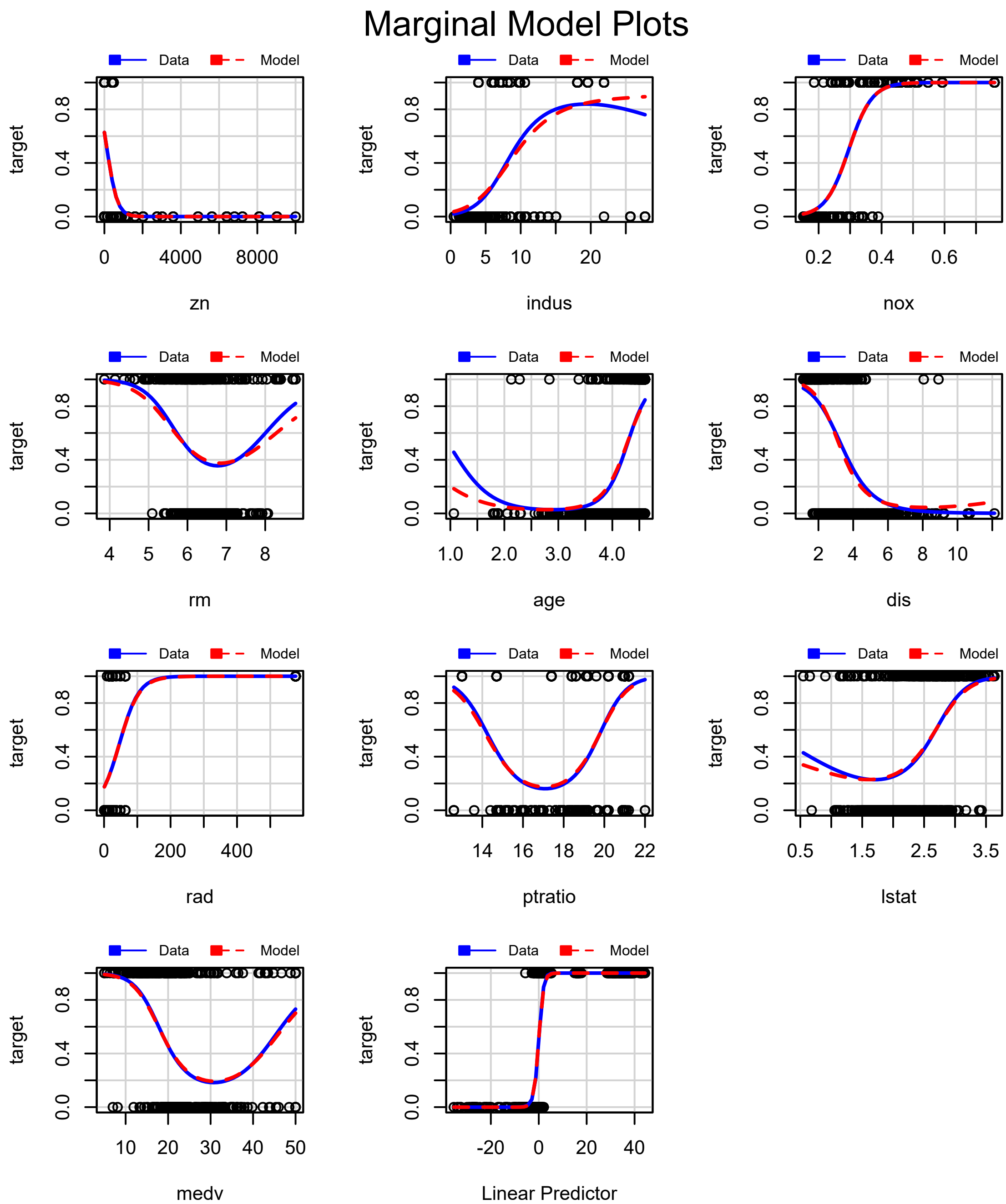
```
##          (Intercept)                   zn                indus
##         2.401238e-19         9.242019e-01         9.392004e-01
##                chas1                  nox                   rm
##         2.883626e+00         2.303854e+19         5.669243e-01
##                  age                  dis                  rad
##         1.019255e+00         1.947542e+00         2.106804e+00
##                  tax              ptratio                lstat
##         9.926673e-01         1.871054e+00         1.082202e+00
##                 medv   ptratio_indicator1     lstat_indicator1
```

```
##         1.198521e+00         6.070332e+00         1.636415e+00
##       dis_indicator1 age_greater_than_771
##         1.344905e+00         1.926483e+00
```

Second Model

For our second model, we will use the `trans_train` dataset which includes the transformed variables in the previous section in addition to the engineered features.

```
##
## Call:
## glm(formula = target ~ ., family = binomial(link = "logit"),
##     data = trans_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0936  -0.1954   0.0000   0.0000   3.3461
##
## Coefficients:
##                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)          -24.446684   6.912815  -3.536 0.000406 ***
## zn                    -0.003352   0.001694  -1.979 0.047854 *
## indus                 -0.131736   0.049064  -2.685 0.007254 **
## chas1                  1.556410   0.761120   2.045 0.040865 *
## nox                   41.581269   7.974850   5.214 1.85e-07 ***
## rm                    -0.842981   0.697236  -1.209 0.226650
## age                   -0.038224   0.667053  -0.057 0.954304
## dis                    0.571657   0.301933   1.893 0.058315 .
## rad                    0.055708   0.013966   3.989 6.64e-05 ***
## ptratio                0.535961   0.176198   3.042 0.002352 **
## lstat                  0.164198   0.799817   0.205 0.837342
## medv                   0.177604   0.064275   2.763 0.005724 **
## ptratio_indicator1     1.408573   1.082973   1.301 0.193377
## lstat_indicator1       0.066430   0.565626   0.117 0.906507
## dis_indicator1         0.004908   0.720640   0.007 0.994566
## age_greater_than_771   1.342623   0.565052   2.376 0.017497 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 645.88  on 465  degrees of freedom
## Residual deviance: 195.16  on 450  degrees of freedom
## AIC: 227.16
##
## Number of Fisher Scoring iterations: 10
```

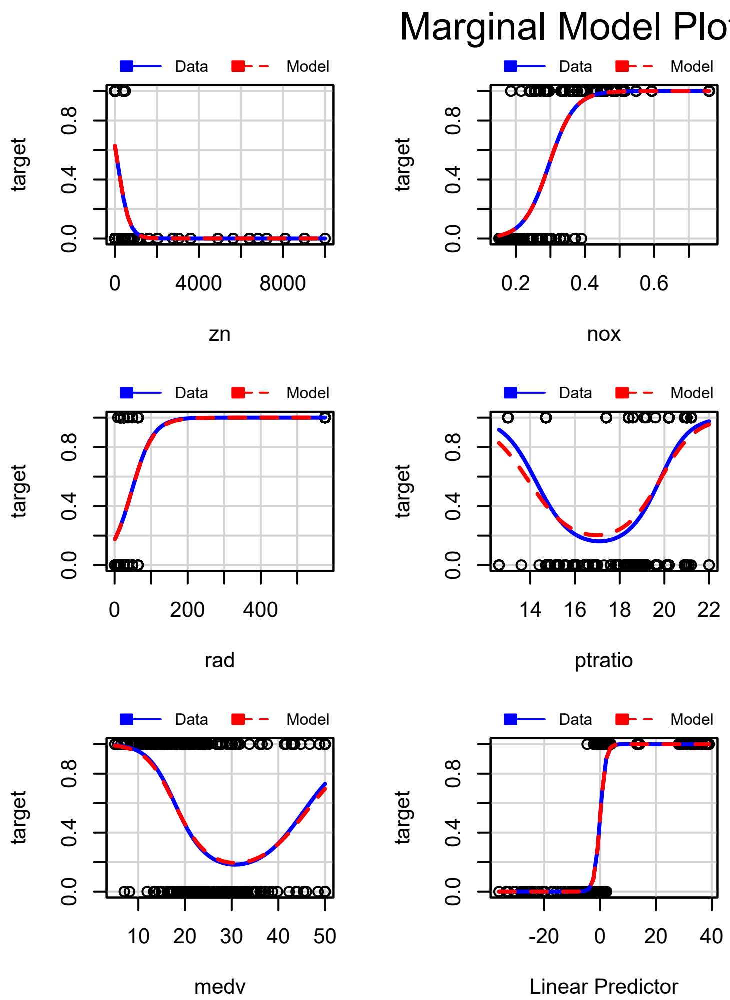


This plot looks very similar to model_1's, with fewer outliers and stronger relationships between the data and the model.

Third Model

For our third model, we will use some of the original variables:

```
##
## Call:
## glm(formula = target ~ . - rm - chas - age - indus, family = binomial(link = "logit"),
##     data = trans_train_mod_3)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1908  -0.2806   0.0000   0.0000   3.0407
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -22.812101   4.437941  -5.140 2.74e-07 ***
## zn           -0.003397   0.001442  -2.355  0.01850 *
## nox          34.818425   5.399202   6.449 1.13e-10 ***
## dis           0.554423   0.190879   2.905  0.00368 **
## rad           0.052140   0.011910   4.378 1.20e-05 ***
## ptratio       0.236630   0.099364   2.381  0.01724 *
## lstat         0.750120   0.560658   1.338  0.18092
## medv          0.133242   0.042226   3.155  0.00160 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 645.88  on 465  degrees of freedom
## Residual deviance: 215.94  on 458  degrees of freedom
## AIC: 231.94
##
## Number of Fisher Scoring iterations: 10
```



Based on the Marginal model plot, model3 might be the best fit for our data.

Select Models

To determine if the models produced have significant improvement in fit over the null model, we make use of the `anova()` function.

```
## Analysis of Deviance Table
##
## Model 1: target ~ 1
## Model 2: target ~ zn + indus + chas + nox + rm + age + dis + rad + tax +
##     ptratio + lstat + medv + ptratio_indicator + lstat_indicator +
##     dis_indicator + age_greater_than_77
## Model 3: target ~ zn + indus + chas + nox + rm + age + dis + rad + ptratio +
##     lstat + medv + ptratio_indicator + lstat_indicator + dis_indicator +
##     age_greater_than_77
## Model 4: target ~ (zn + indus + chas + nox + rm + age + dis + rad + ptratio +
##     lstat + medv) - rm - chas - age - indus
##   Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
## 1       465     645.88
## 2       449     187.56 16   458.32 < 2.2e-16 ***
## 3       450     195.16 -1    -7.60  0.005845 **
## 4       458     215.94 -8   -20.78  0.007747 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The Akaike information criterion (AIC) is a good test for model fit. AIC calculates the information value of each model by balancing the variation explained against the number of parameters used.

In AIC model selection, we compare the information value of each model and choose the one with the lowest AIC value (a lower number means more information explained!)

```
##
## Model selection based on AICc:
##
##         K   AICc Delta_AICc AICcWt Cum.Wt      LL
## model1 17 222.92       0.00   0.93   0.93  -93.78
## model2 16 228.37       5.44   0.06   0.99  -97.58
## model3  8 232.25       9.33   0.01   1.00 -107.97
```

Conclusions

We fitted three models using a combination of variables/strategies: original, engineered and removing certain variables. We looked at the marginal model plots as well as the Akaike Information Criterion. Since for the AIC a lower number means more information explained, we have chosen Model3 as the best fit model.

References

Minitab Support, Accessed 10/2021: “Interpret the key results for Marginal Plot” [https://support.minitab.com/en-us/minitab/19/help-and-how-to/graphs/marginal-plot/interpret-the-results/key-results/]

Research compendium cboettig/noise-phenomena: Supplement to: “From noise to knowledge: how randomness generates novel phenomena and reveals information” by Carl Boettiger

Yogita Bor, Accessed 10/2021: Guide for building an End-to-End Logistic Regression Model. [https://www.analyticsvidh for-building-an-end-to-end-logistic-regression-model/?utm_source=feedburner&utm_medium=email&utm_campaign=Fe

Appendix A: Code

```
#load data
crime_train <- read.csv(paste("https://raw.githubusercontent.com",
                      "/akarimhammoud/Data_621/main/Assignment_3/",
                      "data/crime-training-data_modified.csv"))
crime_test <- read.csv(paste("https://raw.githubusercontent.com",
                       "/akarimhammoud/Data_621/main/Assignment_3/",
                       "data/crime-evaluation-data_modified.csv"))
```

```
# Data Exploration
descr(crime_train,
  headings = FALSE, #remove headings#
  transpose = TRUE #allows for better display due to large amount of variables
  ) %>%
  kbl(caption = "Univariate Descriptive Statistics - Training Data Set") %>%
  kable_styling(bootstrap_options = c("striped", "hover", "condensed"))
```

```
# distribution of the varaibles
crime_train %>%
  gather(variable, value, zn:target) %>%
  ggplot(., aes(value)) +
  geom_density(fill = "steelblue", color="steelblue") +
  facet_wrap(~variable, scales ="free", ncol = 4) +
  labs(x = element_blank(), y = element_blank())
```

```
# Check for NA values
map(crime_train, ~sum(is.na(.))) %>% t()
```

```
missmap(crime_train, main = "Missing vs Observed Values")
```

```
#make a copy of original dataset
train <- crime_train

#convert chas and target to factors
train$chas <- as.factor(train$chas)
train$target <- as.factor(train$target)

#add new variables
train$ptratio_indicator <- as.factor(ifelse(train$ptratio < 16, 1, 0))
train$lstat_indicator <- as.factor(ifelse(train$lstat < 15, 1, 0))
train$dis_indicator <- as.factor(ifelse(train$dis < 4, 1, 0))
train$age_greater_than_77 <- as.factor(ifelse(train$age >= 77, 1, 0)) #median age is 77
```

```
#MI find multicollinear variables
kable((car::vif(glm(target ~. ,
                    data = crime_train))),
      col.names = c("VIF Score")) %>%  #remove tax for high vif score
  kable_styling(full_width = F)
```

```r
#capping outliers
trans_train_cap <- train %>%
  dplyr::select(-tax)
```

```r
crimeid <- c(1:12)
for (val in crimeid) {
  qnt <- quantile(crime_train[,val], probs=c(.25, .75), na.rm = T)
  caps <- quantile(crime_train[,val], probs=c(.05, .95), na.rm = T)
  H <- 1.5 * IQR(crime_train[,val], na.rm = T)
  crime_train[,val][crime_train[,val] < (qnt[1] - H)] <- caps[1]
  crime_train[,val][crime_train[,val] > (qnt[2] + H)] <- caps[2]
}
```

```r
# MI transformation of the variables.
trans_train <- train %>%
  dplyr::select(-tax) %>%
  mutate(age = log(age),
         lstat = log(lstat),
         zn = zn^2,
         rad = rad^2,
         nox = I(nox^2))
```

```r
# MI histogram distribution of the transformed variables
trans_train %>%
  gather(key, value, c(age, lstat, zn, rad, nox)) %>%
  ggplot(., aes(value)) +
  geom_density(fill = "steelblue", color="steelblue") +
  facet_wrap(~ key, scales = 'free', ncol = 3) +
  labs(x = element_blank(), y = element_blank())
```

```r
#Null Model
null_model <- glm(target~1,
                  data=train,
                  family="binomial"(link = "logit"))

#Model 1, data: train
model_1 <- glm(target~.,
               family = "binomial"(link = "logit"),
               data = train)
confint.default(model_1)

exp(cbind(OR=coef(model_1), confint.default(model_1)))

#Model 2 data: trans_train
model_2 <- glm(target~ .,
               family = binomial(link = "logit"),
               data = trans_train)

#Model 3 Data: trans_train_mod_3
trans_train_mod_3 <- trans_train %>%
```

```r
  dplyr::select(1:12)
model_3 <- glm(target ~ . -rm -chas - age -indus,
             family = binomial(link = "logit"),
             trans_train_mod_3)

#Comparing Models via Anova
anova(null_model,
      model_1,
      model_2,
      model_3,
      test="LRT")


#Comparing models with AIC
model.set <- list(model_1,
      model_2,
      model_3)
model.names <- c("model1", "model2", "model3")

aictab(model.set, modnames = model.names)
```