Crime Prediction

Biguzzi, Connin, Greenlee, Moscoe, Sooklall, Telab, and Wright

10/15/2021

Introduction

Is it possible to predict whether a neighborhood's crime rate will be above or below the city's median crime rate? In this report we will attempt to answer this question using logistic regression techniques. The `crime` dataset provided could show which variables are instrumental in crime prediction. Understanding and identifying variables where there are differences within the two target groups are essential to predicting other neighborhoods not in the dataset.

In this report we will:

- Explore the data
- Transform data to address multicollenearity and meet variable distribution needs
- Compare different models and select the most accurate model
- Test our model on the evaluation dataset

Data Exploration

In exploring the data we wanted to understand the following:

- Distributions of the variables to understand what transformations or variable re-coding makes sense.
- Understand what variables seem to have a difference between the target groups.
- Correlation matrix to figure out the covariance and multicollinearity between the predictor variables to help understand what interactions would make sense.

We also looked at missing values and found that no column is missing any values. There is also not enough evidence to suggest that any value within column is a code for missing data. Therefore, we decided to keep all the data points.

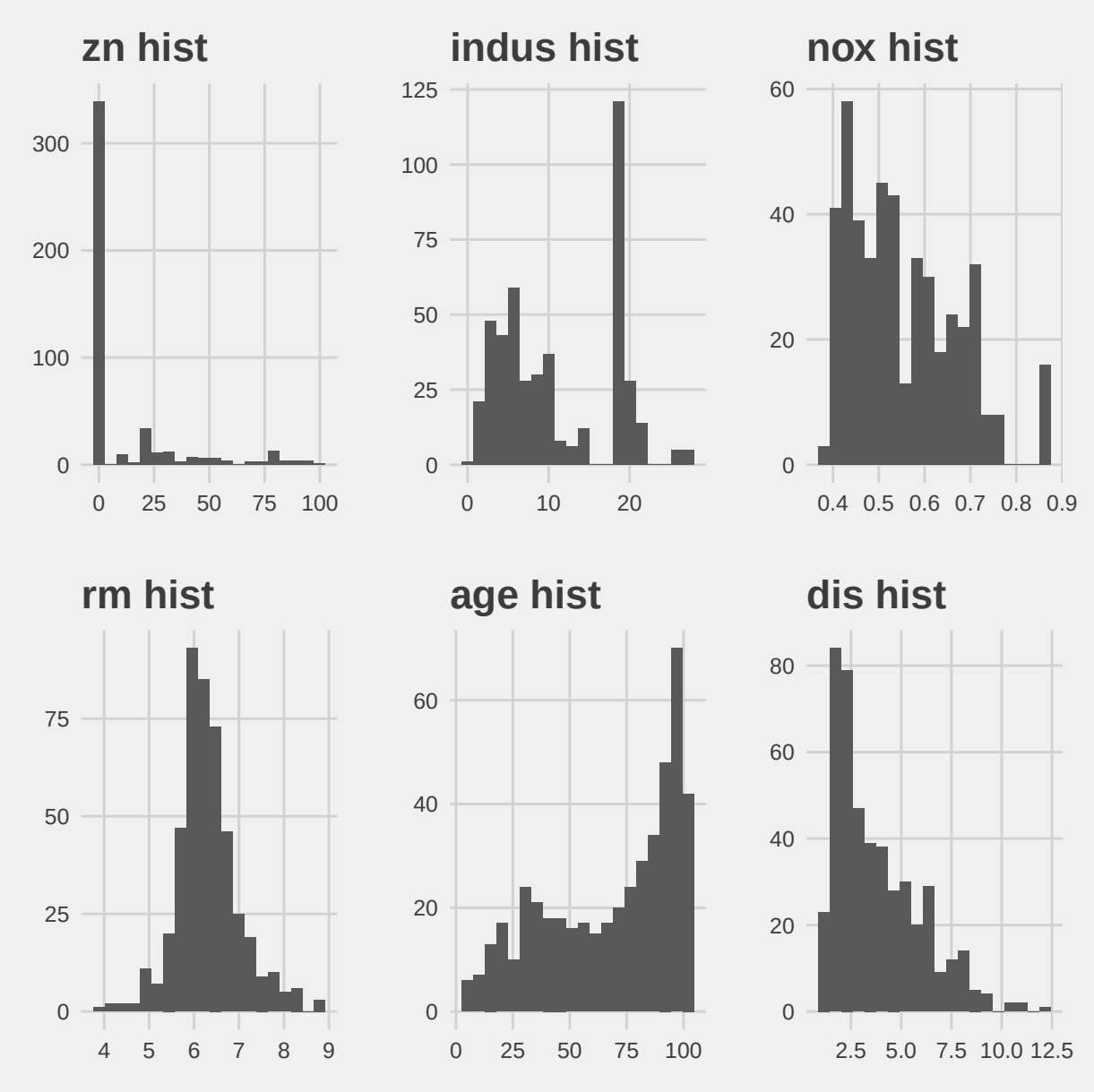
Distribution

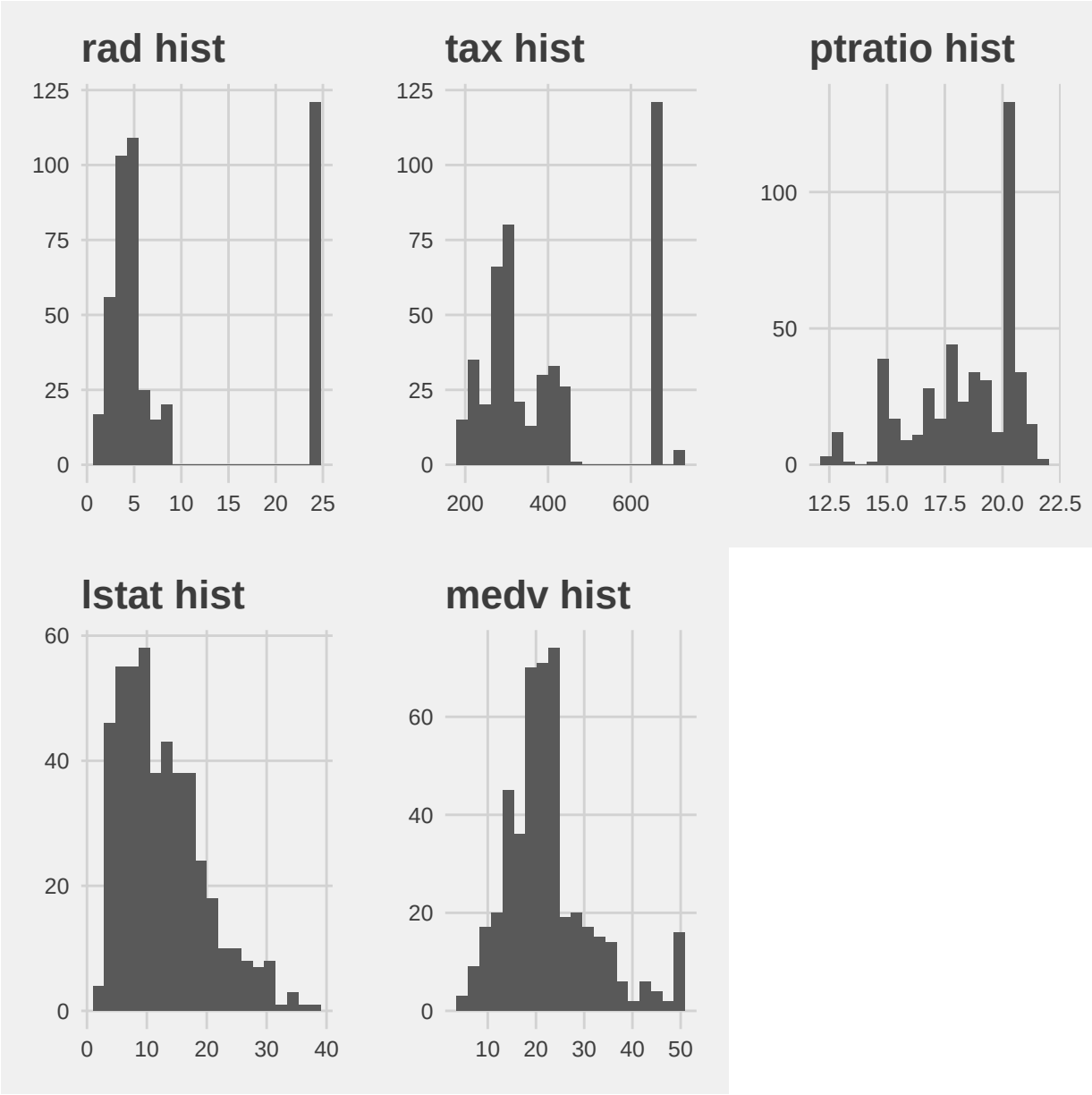
Considering we are attempting to create a logistic regression we want to first check the distribution with the data of how many data points are above the median crime rate and how many are below the median crime rate.

	Distribution
below median	237
above median	229

Looking at the table we see a fairly even distribution of the target variable with 50.9% of the neighborhoods crime rate below the median and 49.1% having a crime rate above the median.

Next, we wanted to look at the distribution for all the numeric variables as well as the distribution of the other factor variable, `chas`.





Because we are attempting to create a logistic regression model normally distributed variables are not necessary as in a linear regression. Getting a general idea of the distributions allows us to start understanding the data. For example, the histograms plots for `tax` and `rad` seem to be very similar, which might suggest multicollinearity. Additionally, it might make sense to re-code some of the other variable like `zn` as binary variable. We will explore these ideas in the below sections. We also wanted to check the distribution of the data for the `chas` column.

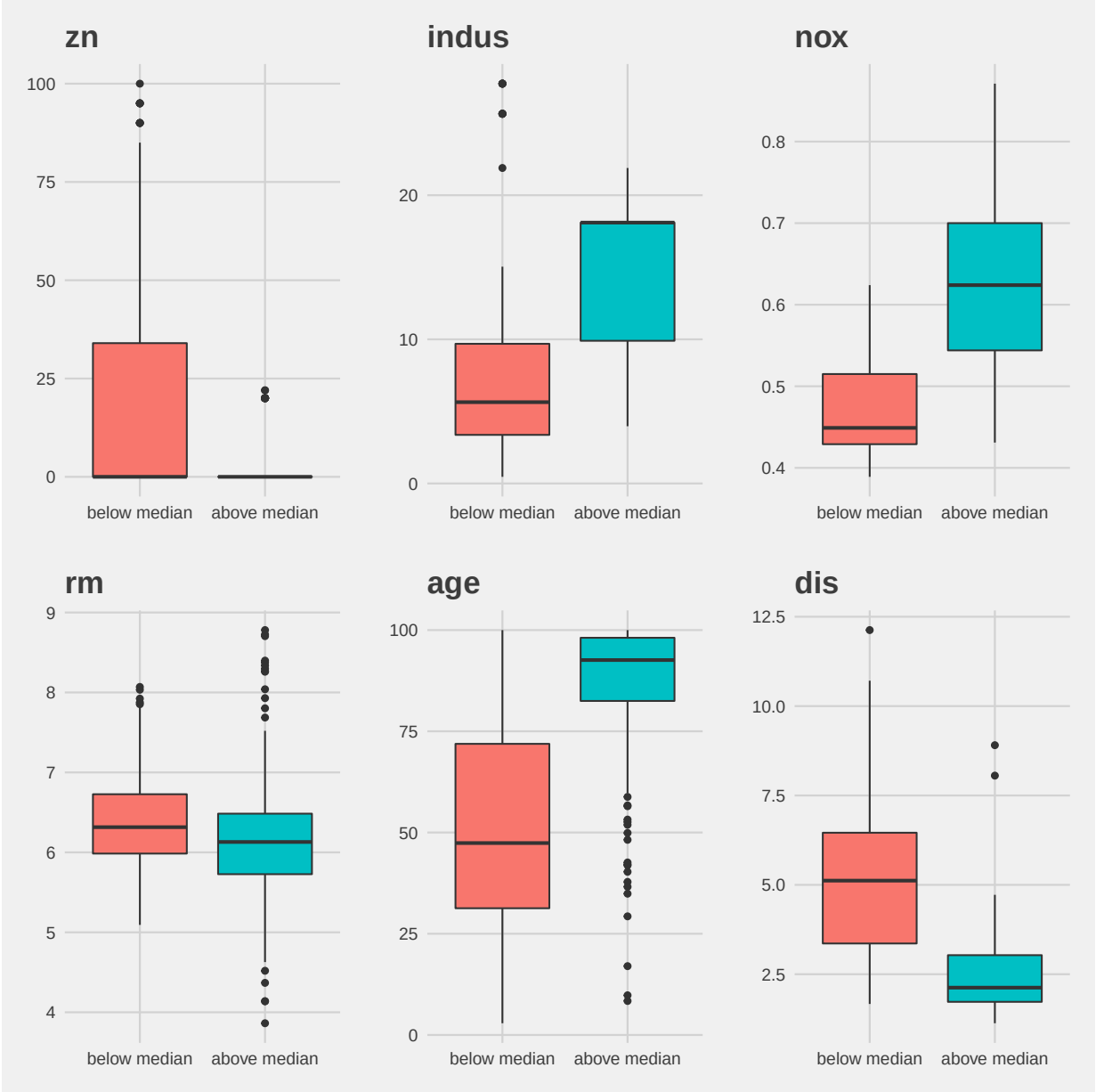
	Distribution
no river border	433
river border	33

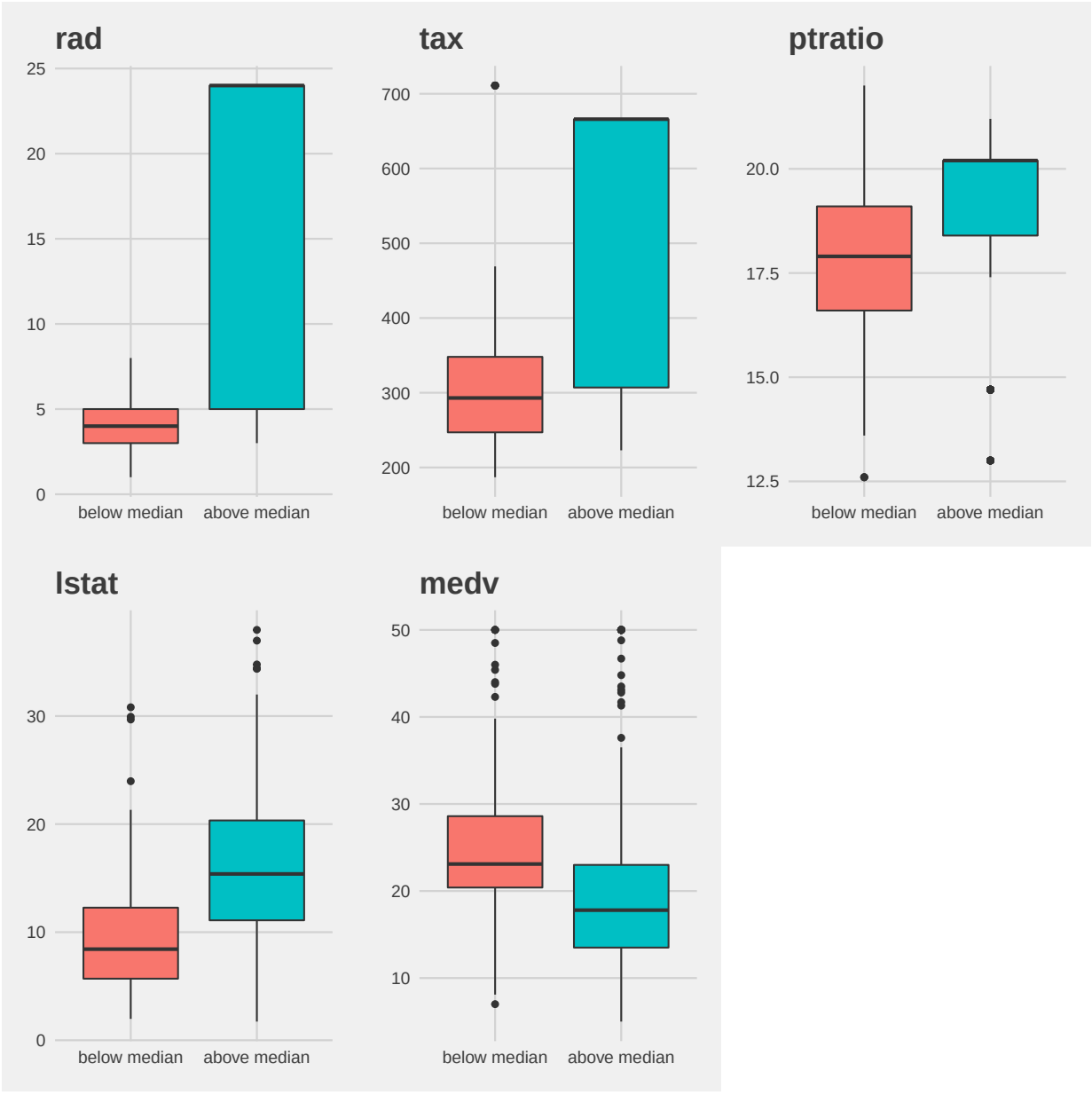
Looking at the table above we see that 92.9% of the neighborhoods don't border a river and only 7.1% of

neighborhoods border a river. This suggest the `chas` variable is not very useful for prediction and can be dropped from all models.

Predictor variables vs Target

Since the target variable is a factor variable we can use boxplots to get a sense of what the differences between groups are. Using the boxplots below, we can see the change in variable distributions that are associated with the target variable.





Examining the boxplots we can see that in almost all of the variables there seems to be a difference between the target groups. The only variable where it seems minimal is `rm`. This suggests the variable will not be a great predictor for the target.

Furthermore, to double check that our assumption to drop the `chas` variable was correct.

	below median	above median
no river border	225	208
river border	12	21

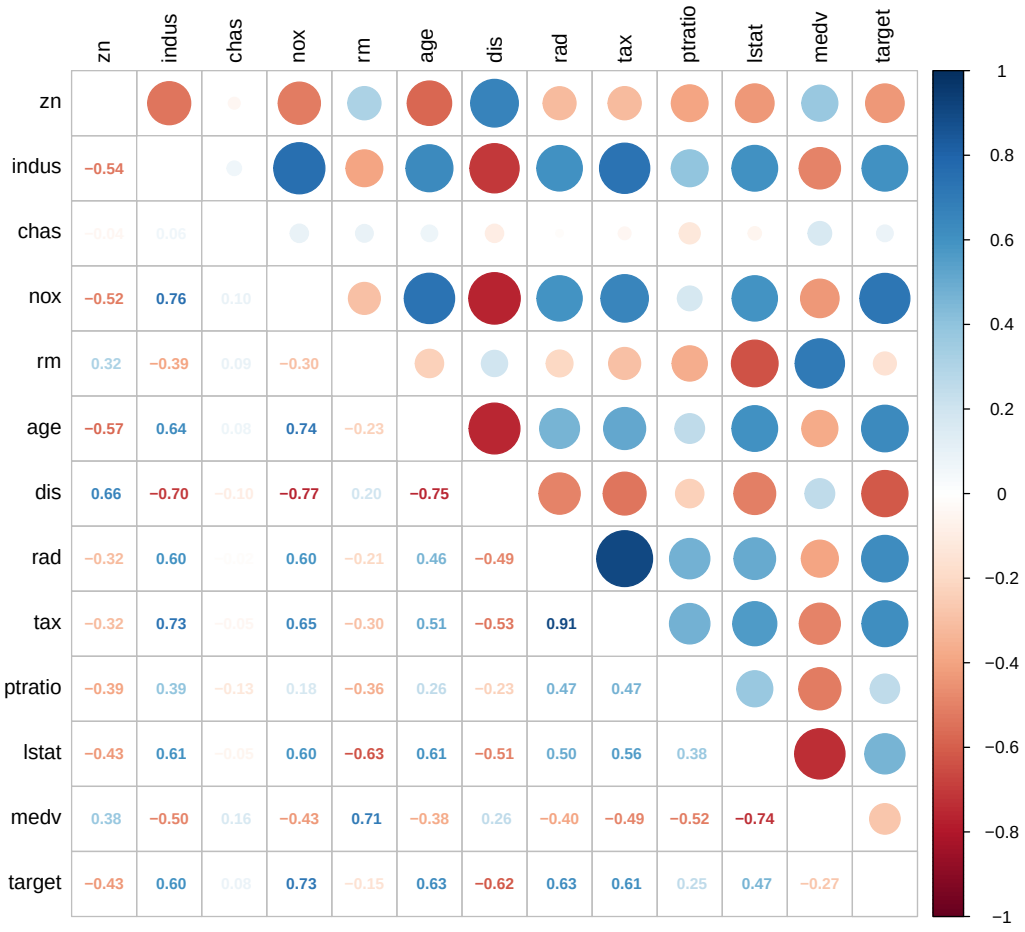
There is not much difference between the percent of neighborhoods without a river border between neighborhoods with below median crime rates, 94.9%, and neighborhoods with above median crime rates, 90.8%.

Furthermore, running a χ^2 test we see that there is no difference between the `chas` groups with a p-value of 0.122. This suggests that remove the `chas` variable was the correct decision.

Covariance & Collinearity

One of the assumptions with logistic regression is that there is no collinearity within the independent variables. We will address these issues in the following section.

The first step is to understand the covariance. Covariance is not an issue in itself, but a correlation matrix helps us understand which variables need to be explored further.

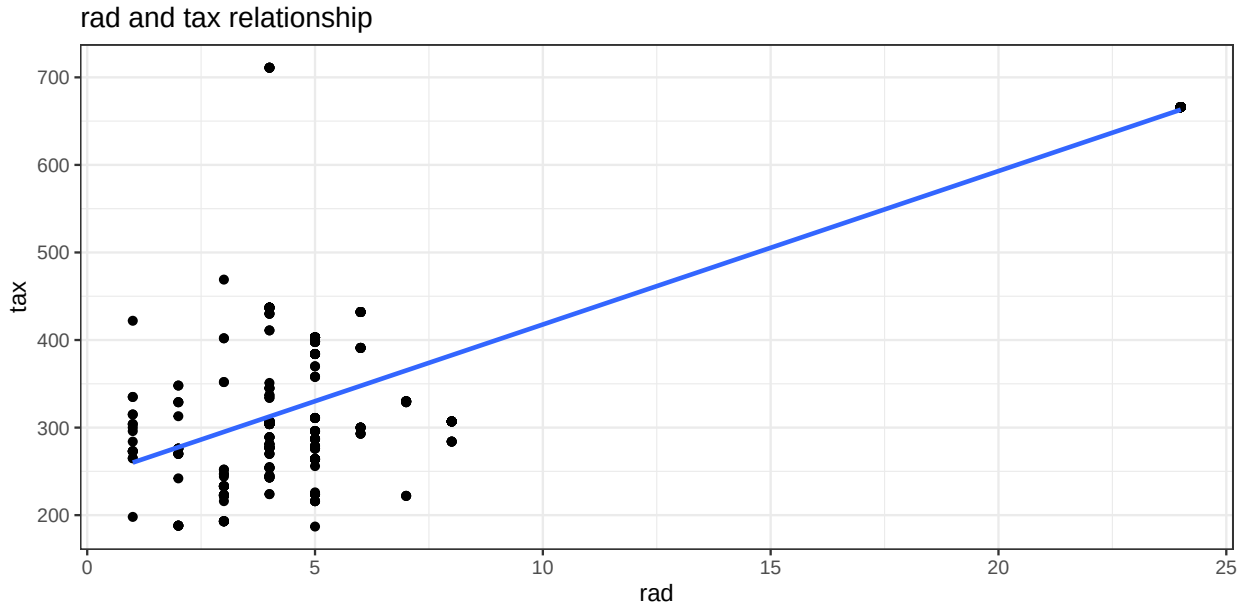


The correlation matrix is a little hard to read so looking at the top 5 correlation coefficients and the associated variables will give us a better sense of the variables to check.

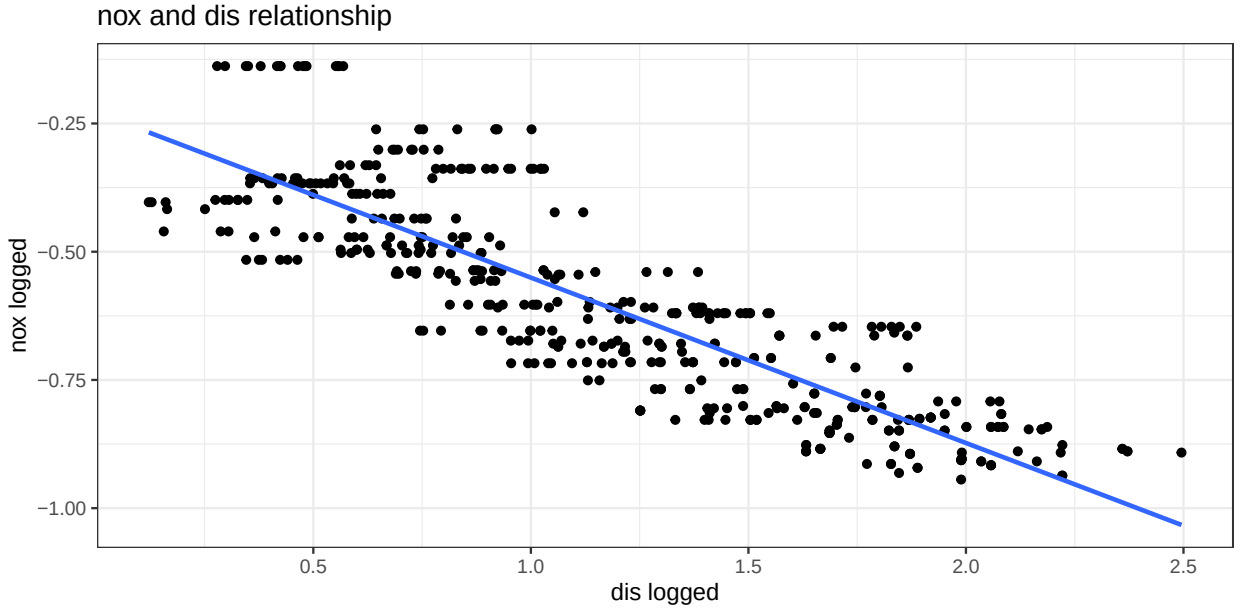
correlation	variable 1	variable 2
0.9064632	tax	rad
-0.7688840	nox	dis
0.7596301	nox	indus
-0.7508976	dis	age
-0.7358008	medv	lstat
0.7351278	nox	age

correlation	variable 1	variable 2
0.7322292	tax	indus
0.7261062	target	nox
0.7053368	rm	medv
-0.7036189	indus	dis

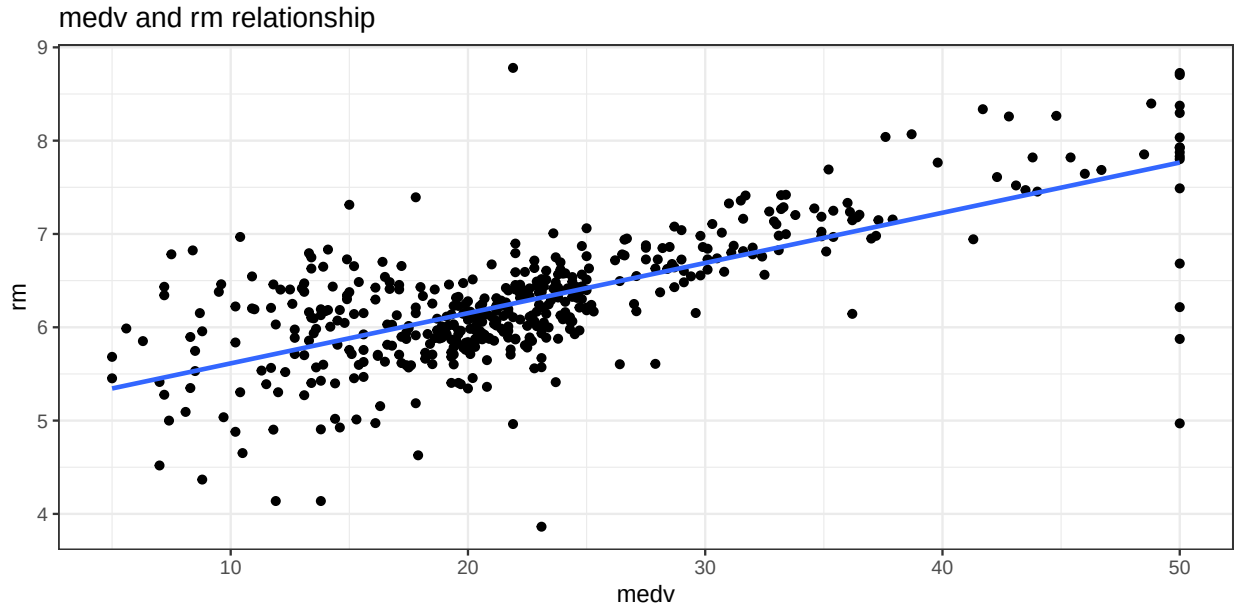
The table above suggest there might be collinearity between `rad` and `tax`, `nox` and `dis`, `nox` and `indus`, `dis` and `age`, and `medv` and `rm`. Let's take a closer look at `rad` and `tax`, `nox` and `dis`, and `medv` and `rm` variables.



For the `rad` and `tax` plot, the regression line doesn't seem to fit the data perfectly, but with a correlation coefficient of 0.906 we will have to deal with them.



For the `dis` and `nox` variables, we see a much clearer pattern. With the clear relationship and a correlation coefficient of -0.769, we will also have to figure out a way to deal with this relationship.



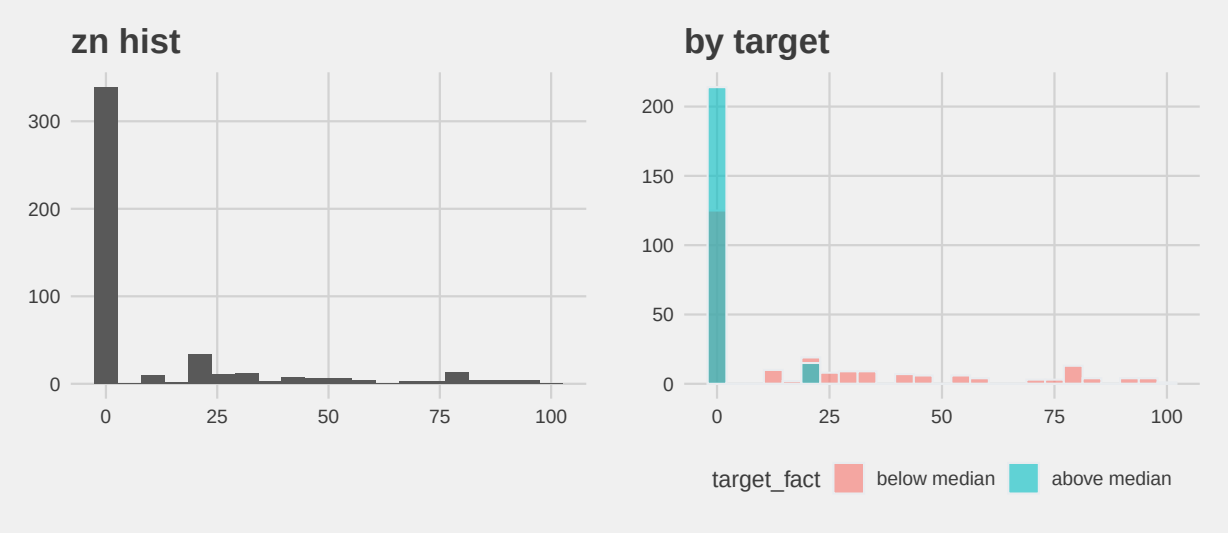
Finally, `medv` and `rm` have a lower correlation coefficient with 0.705, but it makes theoretical sense that they would be correlated. The more rooms in a house the higher the value increases.

Data Prep

In this section we will prepare the data for our modeling.

Re-Coding

`zn` was one of the variables of interest for re-coding. Let's take a closer look at the histogram and compare it to a histogram of `zn` by target variable.

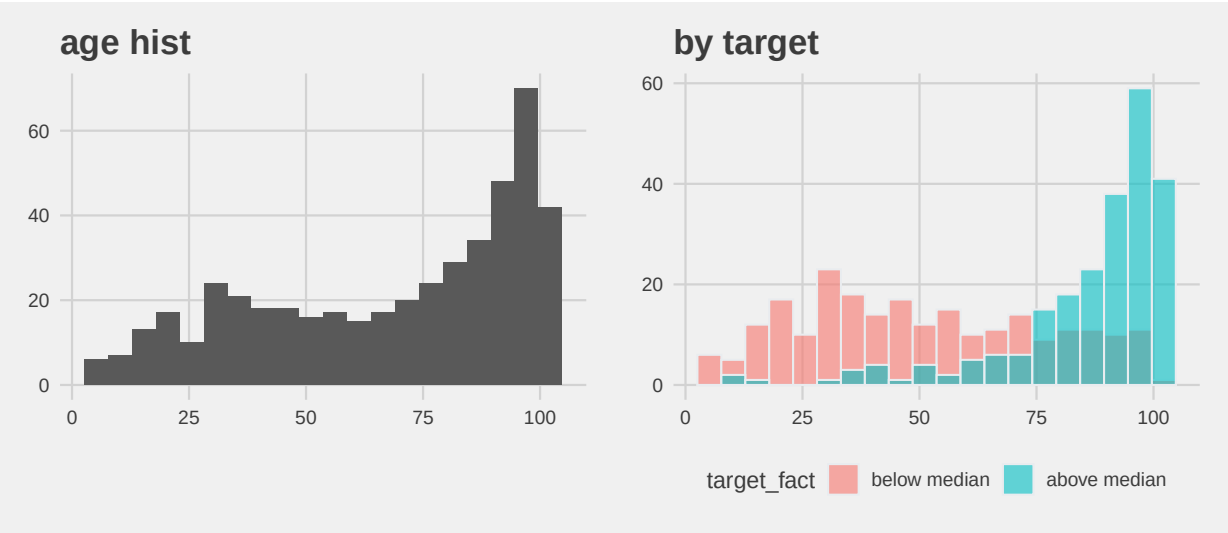


We can see that almost all of the neighborhoods with above median crime rates have no land zoned for large lots, while there is more distribution of the proportion of land zoned for large lots for neighborhoods with below median crime rates.

	below median	above median
No large lot	125	214
Large lot	112	15

In fact, looking at the table above, we can see how large the split is for neighborhoods above the median crime rate, with 93.4% having no land zoned for large lots. it's worth attempting to re-code this variable as it might help our final model make predictions.

Another variable that looked interesting was the `age` variable. Let's compare the two histograms, the first without splitting by target variable and the second grouping by target variable.

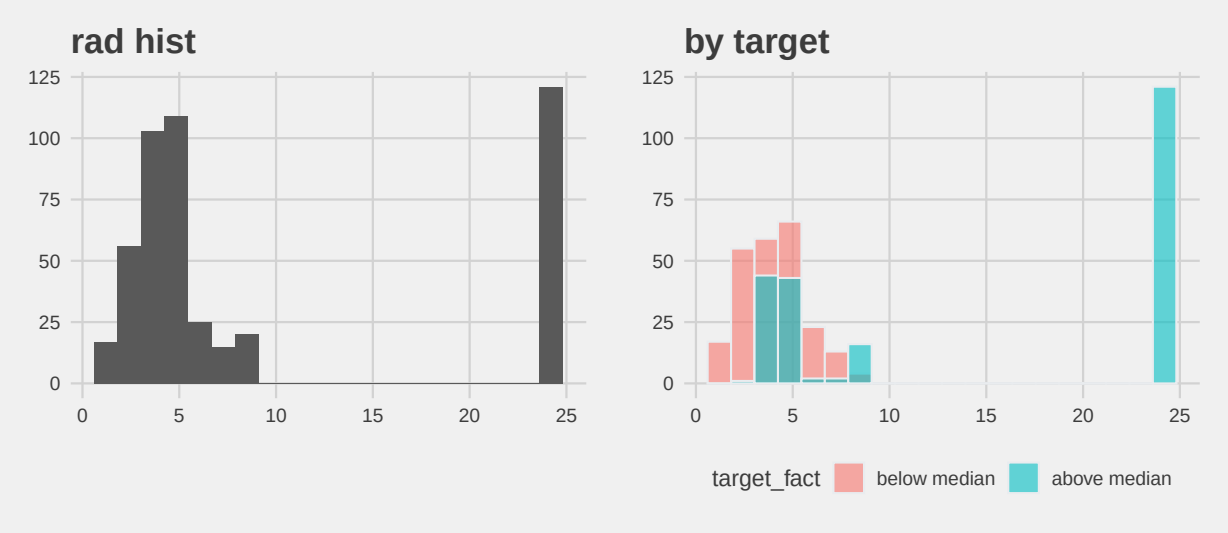


Not grouping by the target variable we can see a bimodal histogram. Grouping by target variable makes the bimodal distribution more interesting as the first peak seems to account for most of the neighborhoods below the median crime rate and the second peak is made up of mostly neighborhoods above the median crime rate. Let's re-code the age variable to a factor variable with two levels, below median age and above median age.

	below median	above median
Below median age	189	44
Above median age	48	185

Looking at the frequency table above, re-coding the age variable has a clear split between groups. 80.8% of neighborhoods with above median crime rate also have an above median proportion of owner-occupied units built prior to 1940. Conversely, 79.7% of neighborhoods below the median crime rate have a below median proportion of owner-occupied units built prior to 1940.

Another variable that makes sense to re-code is `rad`.



Looking at the original histogram we see that a lot of the neighborhoods have a rad index of 24. By also examining the histogram grouped by target variable we see that all almost all the values above the mean are neighborhoods with above median crime rate.

	below median	above median
<= mean rad	237	108
above mean rad	0	121

Examining the table of the new `rad_fact` variable with the `target_fact` variable we see that all of the neighborhoods below the median crime rate fall below the mean rad index, while 52.8% of the neighborhoods with above median crime rates also have an above the mean rad index.

Transformations

nox PPM to PPH

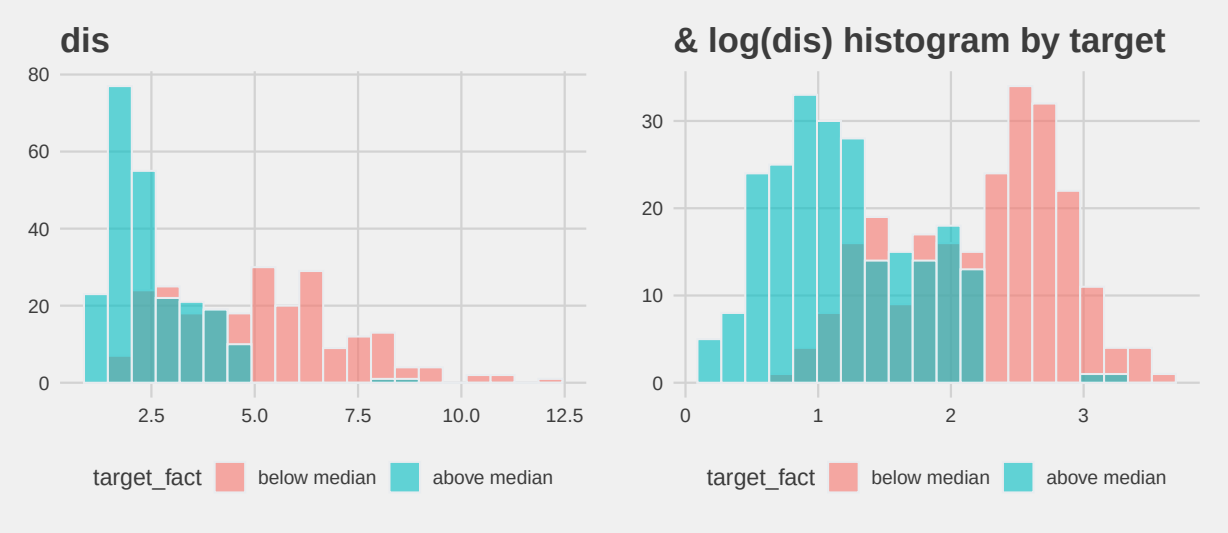
Transforming the `nox` PPM to a smaller unit such as PPH will help with understanding the odd factors of the regressions.

model	odds_factor
nox pp10m model	5.719360e+12
nox pph model	1.341447e+00

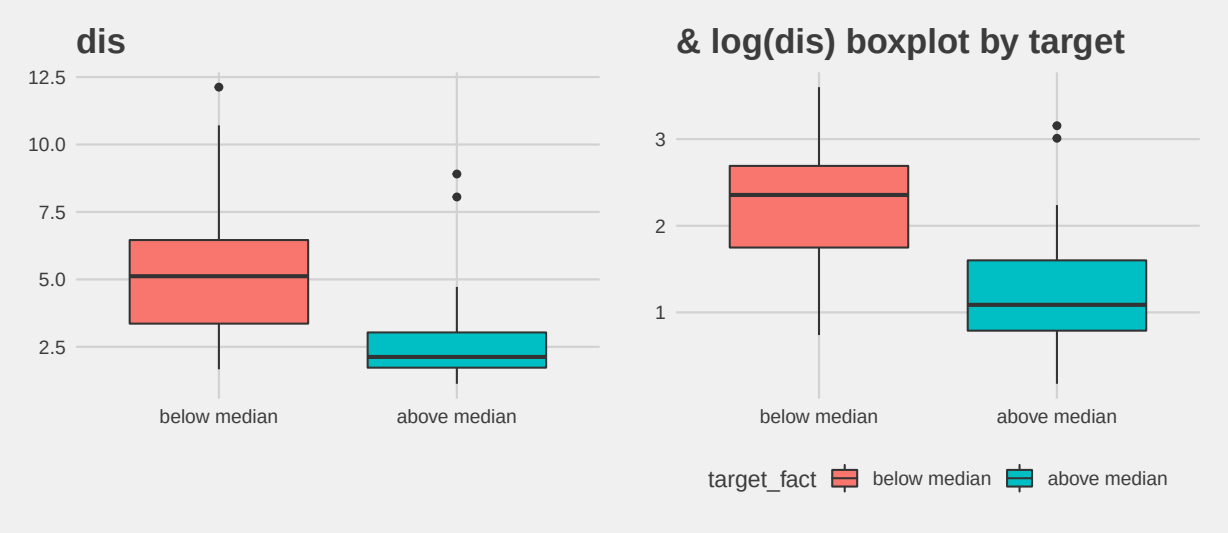
Quickly looking at odds factor for a logit model using `target` and `nox`, the original value of PP10M would be interpreted as, for every 10M parts of nitrogen oxide in the air crime increases 5,719,359,874,044 fold, an incomprehensible number. But when we convert the `nox` variable to PPH we can see the odds factor become much easier to understand, for every 100 parts of nitrogen oxide in the air we increase our odds of being in a neighborhood with above median crime rate by 1.341 fold.

log(dis)

Another variable that makes sense to transform is `dis`.



Taking the log of dis will give it a more normal distribution, as seen in the histograms above. We also see a much clearer break within the groups than we did with the original variable.



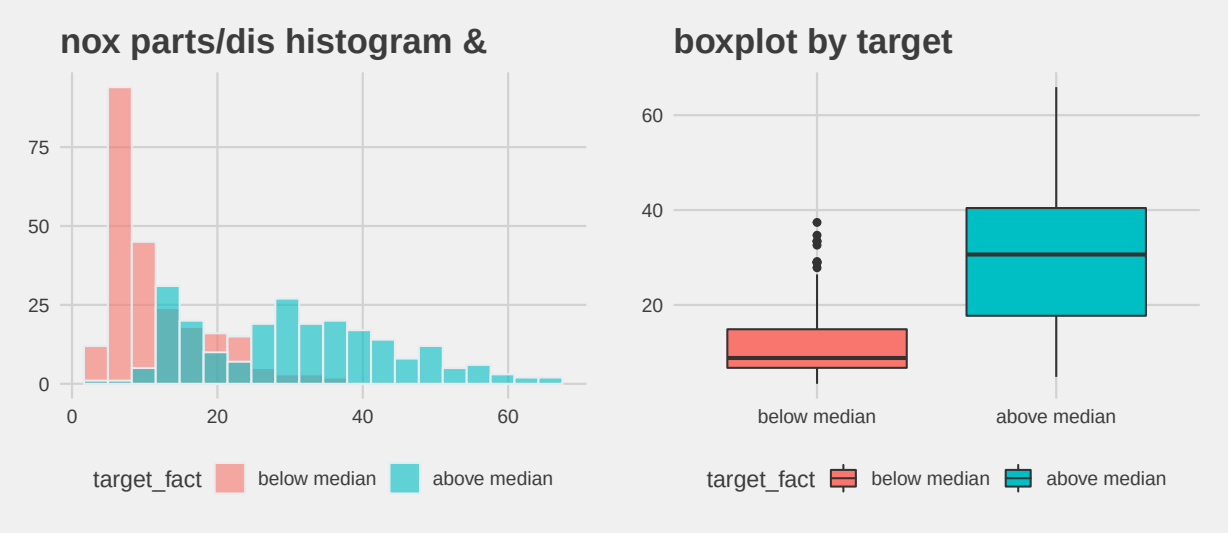
Transforming the `dis` variable does not alter the variance within the target groups. The median of `dis` for neighborhoods below median crime rates was 2.41 times larger than neighborhoods above median crime rates. Similarly, the median of `log_dis` for neighborhoods below median crime rates was 2.17 times larger than neighborhoods above median crime rates. Additionally, we wont lose too much interpretability when looking at the odds factor by transforming `dis`.

Interactions

In this section we will be interacting `nox_parts` and `dis` using division, `medv` and `rm` using division, and finally `tax` and `rad` using multiplication.

nox & dis

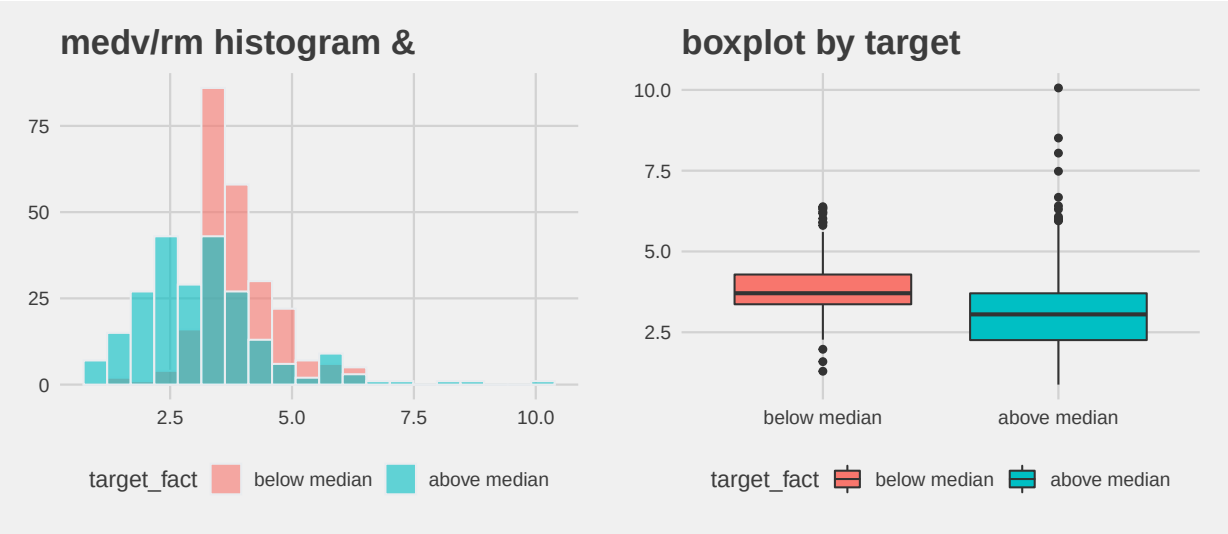
The first pair of variables to interact are the `nox_parts` and `dis`. An interaction here makes sense not only because of the high correlation between the two variables, but also because the air would be more polluted the closer a neighborhood is to an employment center.



There's a clear difference between crime rate groups and the ratio of nox parts to distance from an employment center. The median rate for neighborhoods with above median crime rate is 30.6 PPH for a distance unit from an employment center, compared to 8.8 in below median crime rate neighborhoods. This interaction preserves the relationship the original variables had with the target, and gets rid of any collinearity by combining the two variables.

medv & rm

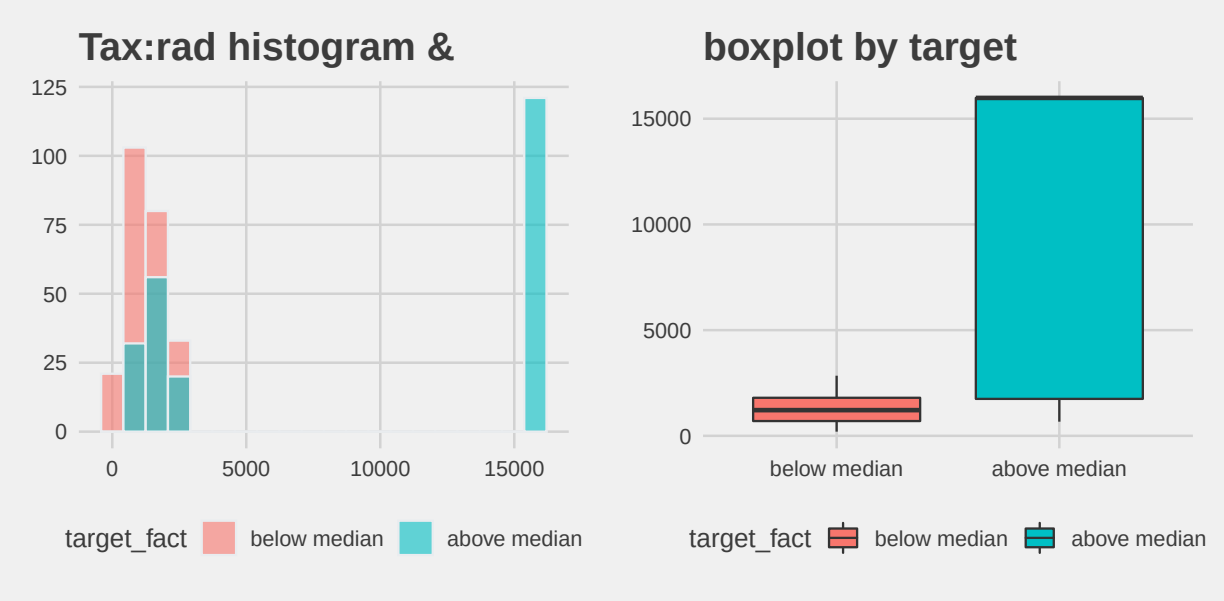
Next we will interact the `medv` and `rm` variables. This interaction makes theoretical sense as stated before, more rooms should mean higher value.



There seems to be a slight difference between the median house value per $1,000 per room between the target variable groups. Neighborhoods with below median crime rates tend to have higher median house value per room with a value of $3,708per average number of rooms compared to $3,052 per average number of rooms for neighborhoods with above median crime rates.

tax & rad

Another method to deal with collinearity is to interact two highly correlated terms. It makes sense to interact `tax` and `rad` since they were the two most correlated variables,0.906, in the data, and had almost identical histograms.



After interacting the terms we see that the new variables histogram and boxplot graphs are nearly identical to the tax and rad histograms and boxplots from earlier.

Building Models

To build models we will take the following steps:

- Start with all the original values
- Use stepwise regression with backward direction to find the model with the best AIC using the original variables
- Run a regression using the newly re-coded variables
- Run a stepwise regression with backward direction to find the model with best AIC
- Run a regression using the newly transformed variables
- Run a stepwise regression with backward direction to find the model with best AIC
- Run a regression with just the interaction terms
- Run a stepwise regression with forward direction to get the model with the best AIC
- Take the information and create 3 thought out models to compare as our final models

Base model

This is our baseline model. Without any data munging, we get a sense of how the data performs at predicting the target.

Let's take a closer look at some diagnostic statistics.

Table 9: Base model results

	Above Median Crime Rate	
	(1)Original Vars	(2)Model 1 with stepwise
zn	−0.1008**	−0.0981**
	(0.0468)	(0.0444)
indus	−0.0663	
	(0.0543)	
nox	55.9742***	51.4249***
	(9.7883)	(8.8191)
rm	−1.2624	−1.2012
	(0.8732)	(0.8109)
age	0.0543***	0.0514***
	(0.0171)	(0.0152)
dis	0.9997***	0.9692***
	(0.2979)	(0.2932)
rad	0.7742***	0.8333***
	(0.1965)	(0.1917)
tax	−0.0056	−0.0070**
	(0.0035)	(0.0031)
ptratio	0.4528***	0.4418***
	(0.1517)	(0.1513)
lstat	−0.0070	
	(0.0602)	
medv	0.2591***	0.2580***
	(0.0896)	(0.0907)
Constant	−45.3107***	−43.2508***
	(8.3600)	(7.9792)
Observations	372	372
Log Likelihood	-72.1384	-72.9595
Akaike Inf. Crit.	168.2769	165.9190
Note:		*p<0.1; **p<0.05; ***p<0.01

model	predictors	precision	auc	AIC	BIC
model1: base variables	11	0.91	0.91	168.28	215.30
model2: model 1 stepwise	9	0.89	0.89	165.92	205.11

Model 1 and Model 2 seem to predict the target variables extremely well with an AUC of 0.915 and 0.893, respectively. Model 2 does, however, have a better AIC score indicating a better fit. Next, we will look at our tranformation models.

Re-coded vars model

Model 3 looks at a baseline for the re-coded variables. Running a stepwise on Model 3 does seem to improve the AIC value, meaning Model 4 fits the data better, while using less predictors. Additionally, we don't see a decrease in AUC score between Model 3 and Model 4.

Table 11: Models with re-codede vars

	Target Variable	
	(3)Re-Coded Vars	(4)Model 3 Stepwise
indus	-0.1555^{***}	-0.1494^{***}
	(0.0530)	(0.0499)
nox	57.4519^{***}	56.9777^{***}
	(9.0333)	(8.9910)
rm	−0.1448	
	(0.6943)	
dis	1.0424^{***}	1.0582^{***}
	(0.2665)	(0.2613)
tax	0.0003	
	(0.0026)	
ptratio	0.1281	
	(0.1224)	
lstat	0.0920	0.0963^{*}
	(0.0563)	(0.0519)
medv	0.2150^{***}	0.1921^{***}
	(0.0700)	(0.0469)
age_factAbove median age	1.4478^{***}	1.3649^{***}
	(0.5286)	(0.4790)
zn_factLarge lot	-3.4352^{***}	-3.8199^{***}
	(1.0412)	(0.9817)
rad_factabove mean rad	19.4866	19.6011
	(1,336.6820)	(1,339.1620)
Constant	-41.3281^{***}	-39.0651^{***}
	(7.1826)	(6.2016)
Observations	372	372
Log Likelihood	-80.8487	-81.3919
Akaike Inf. Crit.	185.6975	180.7838
Note:		*p<0.1; **p<0.05; ***p<0.01

model	predictors	precision	auc	AIC	BIC
model3: re-coded vars	11	0.83	0.86	185.70	232.72
model4: model3 stepwise	8	0.83	0.86	180.78	216.05

However, Model 4 did not perform better than Model 2 as the AUC dropped from 0.893 for Model 2 to 0.863 and the AIC increased by 14.9 from and AIC of 165.9 for Model 2 to 180.8 for Model 4.

Transformation Models

To get a sense of how our transformed variables performed we ran a regression substituting the originals with the transformed. Similarly to our previous models the stepwise regression provided us with the model with the best fit, meaning Model 6 performs better than Model 5.

Table 13: Models with transformed variables

	Target Variable	
	(5)Transformed Vars	(6)Model 5 Stepwise
zn	-0.0710^{*}	-0.0726^{*}
	(0.0407)	(0.0396)
indus	-0.0283	
	(0.0551)	
rm	-1.4321	-1.3761^{*}
	(0.8972)	(0.8265)
age	0.0594^{***}	0.0576^{***}
	(0.0178)	(0.0160)
rad	0.7924^{***}	0.8187^{***}
	(0.2057)	(0.1979)
tax	-0.0044	-0.0050
	(0.0035)	(0.0033)
ptratio	0.5188^{***}	0.5153^{***}
	(0.1604)	(0.1590)
lstat	-0.0119	
	(0.0596)	
medv	0.2939^{***}	0.2979^{***}
	(0.0936)	(0.0942)
nox_parts	0.5883^{***}	0.5751^{***}
	(0.0983)	(0.0949)
log_dis	3.1515^{***}	3.1947^{***}
	(0.8215)	(0.8152)
Constant	-50.9062^{***}	-50.8651^{***}
	(9.1461)	(9.0828)
Observations	372	372
Log Likelihood	-69.7908	-69.9596
Akaike Inf. Crit.	163.5816	159.9192
Note:		*p<0.1; **p<0.05; ***p<0.01

model	predictors	precision	auc	AIC	BIC
model5: transformed vars	11	0.89	0.92	163.58	210.61
model6: model5 stepwise	9	0.89	0.90	159.92	199.11

Looking at more diagnostic statistics we see that even though Model 6 was a better fit to the data with an AIC of 159.92 compared to 163.58, both models had an equal precision of 0.89.

Interaction term

In this section we are checking how the interaction terms perform within a model by first taking the original variables and adding the interaction terms and then by running a stepwise regression to get the results with the best AIC measure. Model 8 seems to be a very strong model with the lowest AIC we've seen so far at 156.98. Let's taker a closer look at these two models.

Table 15: Models with interaction terms

	Target Variable	
	(7)Interaction Terms	(8)Model 7 Stepwise
zn	−0.0604	
	(0.0521)	
indus	0.0102	
	(0.0636)	
nox	64.8735***	67.7099***
	(11.4577)	(11.3361)
rm	−5.2149**	−3.6988**
	(2.1452)	(1.6611)
age	0.0603***	0.0534***
	(0.0184)	(0.0154)
dis	0.3115	
	(0.4312)	
rad	1.4637***	1.3535***
	(0.4189)	(0.3478)
tax	−0.0016	
	(0.0040)	
ptratio	0.5246***	0.5682***
	(0.1680)	(0.1513)
lstat	−0.0621	
	(0.0664)	
medv	1.1424**	0.8361**
	(0.4522)	(0.3737)
nox_dis_ratio	−0.1773*	−0.2330***
	(0.0953)	(0.0591)
medv_rm_ratio	−5.7381**	−3.7237
	(2.9129)	(2.3635)
tax_rad_int	−0.0015**	−0.0014***
	(0.0007)	(0.0006)
Constant	−23.4713	−33.8356***
	(14.8139)	(11.4155)
Observations	372	372
Log Likelihood	-67.0235	-68.4924
Akaike Inf. Crit.	164.0470	156.9849
Note:		*p<0.1; **p<0.05; ***p<0.01

model	predictors	precision	auc	AIC	BIC
model7: interaction terms	14	0.89	0.92	164.05	222.83
model8: model7 stepwise	9	0.89	0.92	156.98	196.17

Based on the diagnostic measures, the precision and AUC scores are identical for both models. If we were to pick one, however, it would have to be model 8 as the AIC score is 7.06 smaller than Model 7's AIC.

Combined models

Taking all the information from the previous models we will try and build 2 models to best fit the data and make the most precise predictions. In the first model we will use the following variables: `rad`, `tax`, `ptratio`, `age_fact`, `zn_fact`, `nox_parts`, `log_dis`, `medv_rm_ratio`. In the second model we are dropping the `ptratio` and `tax` columns to deal with some of the multicollinearity that they cause. Quickly looking AIC we see that model 10 is worse than model 9, with and AIC score 2.77 greater than model 9.

Table 17: Combined model results

	Target Variable	
	(9)Combined vars 1	(10)Combined vars 2
rad	0.757^{***}	0.583^{***}
	(0.182)	(0.154)
tax	-0.005	
	(0.003)	
ptratio	0.321^{**}	
	(0.147)	
age_factAbove median age	1.910^{***}	1.771^{***}
	(0.543)	(0.515)
zn_factLarge lot	-1.981^{**}	-2.996^{***}
	(0.836)	(0.784)
nox_parts	0.562^{***}	0.531^{***}
	(0.089)	(0.085)
log_dis	2.913^{***}	3.339^{***}
	(0.783)	(0.731)
medv_rm_ratio	1.389^{***}	1.234^{***}
	(0.451)	(0.344)
Constant	-49.244^{***}	-42.325^{***}
	(8.633)	(6.574)
Observations	372	372
Log Likelihood	-71.879	-75.262
Akaike Inf. Crit.	161.757	164.525
Note:		*p<0.1; **p<0.05; ***p<0.01

model	predictors	precision	auc	AIC	BIC
model9: combining vars 1	8	0.87	0.89	161.76	197.03
model10: combined vars model 2	6	0.82	0.86	164.52	191.96

Examining some of the other diagnostic measures we would come to the same conclusion since model 10's scores are worse than model 9's except for BIC. However, I would suggest that model 10 might be better off than model 9 because it deals with more of the multicullinearity in the data. If we consider the correlation coefficients of both `rad` and `tax` and `ptratio` and `medv`, 0.906 and -0.516 respectfully, then removing `tax` and `ptratio` makes theoretical sense. For this reason, Model 10 might perform better with new data.

Choose Model

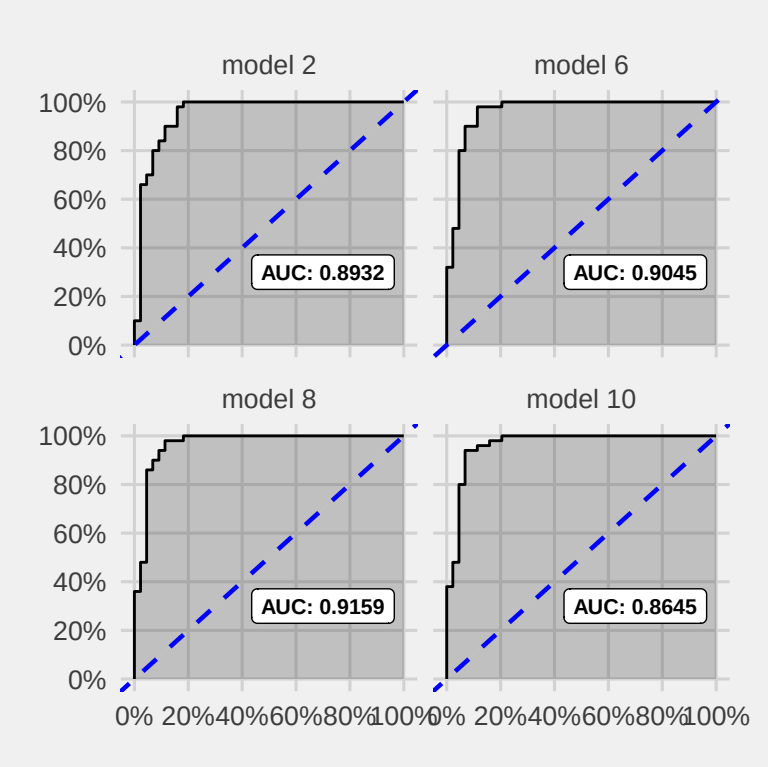
The first thing we want to do is narrow down the models we want to compare. Below are all the models we built and some of their diagnostic measures. We will compare model 2, the stepwise regression of all the original variables, with model 6, model 8, and model 10. We did not select any model with just the re-coded variables as the re-coded model with the lowest AIC was 16.26 higher than the model in our comparison selection with the highest AIC.

Table 19: Diagnostic measures for models to compare

model	predictors	precision	auc	AIC	BIC
model2: model 1 stepwise	9	0.8863636	0.8931818	165.9190	205.1079
model6: model5 stepwise	9	0.8888889	0.9045455	159.9192	199.1081
model8: model7 stepwise	9	0.8913043	0.9159091	156.9849	196.1738
model10: combined vars model 2	6	0.8163265	0.8645455	164.5248	191.9571

Looking at these statistics we should select model 8 as our best model. It has the lowest AIC, the highest precision score and the highest auc score. Before we select this model, however, lets take a look at their ROC curves and VIF values.

Roc



Examining the ROC curves doesn't help us narrow down which model could preform better on new data. All the curves look similar and we already compared AUC scores in the previous section noting that model 8 had the highest AUC score with 0.916. The final metric to check to narrow down which model to check is the VIF scores.

Model VIF

VIF is a measure of how much multicollinearity there is in a model. Usually anything above 5 should be considered and discussed, but anything above 10 would mean your model is not very effective. One of the assumptions of logistic regression is that there is little to no multicollinearity within the data. This is why transformations are vital.

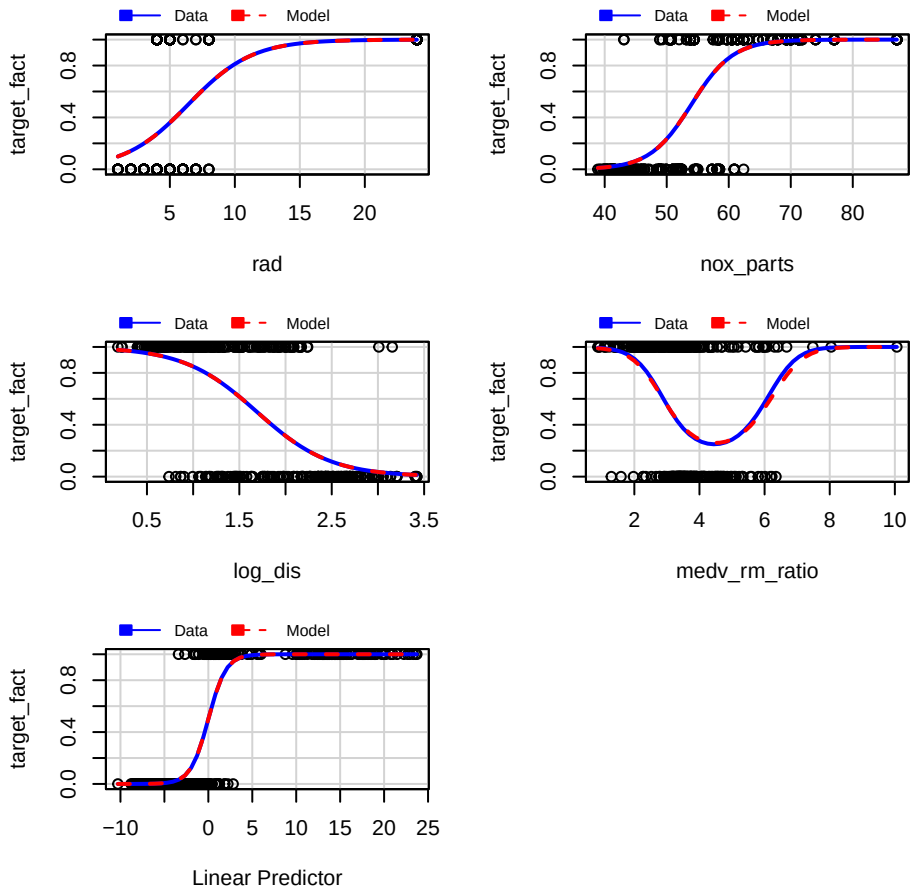
Table 20: Model VIF Scores

variables	model2	model6	model8	model10
medv	9.576469	9.212490	128.755461	
dis	5.256821			
rm	5.206987	4.825975	18.111113	
nox	3.967509		6.160550	
ptratio	2.643625	2.812356	2.311377	
age	2.336085	2.418471	2.134590	
zn	2.301730	1.645558		
rad	1.927693	1.891381	11.097519	1.163776
tax	1.826018	1.991347		
age_fact				1.563783
log_dis		5.073289		4.429158
medv_rm_ratio			70.921994	1.940869
nox_dis_ratio			5.773424	
nox_parts		4.355950		3.961190
tax_rad_int			10.524593	
zn_fact				1.717267

By examining the VIF score table above we see that `medv` has a VIF above 9 for model 2 and model 6, while it has a VIF score of over 100 for model 8. This suggests that dropping that variable and only using the interaction term of `medv_rm_ratio` is a better strategy. Additionally, model 10 is the only model that successfully keeps all the variable with a VIF below 5. Model 10 seems like the best theoretical model, which might suggest it would hold up better with newer data.

Marginal Model Plots

The final thing we want to check with our selected model (model 10) are the marginal model plots. Each data line, that is, the line created for the response on the predictor, mathces the model line, the line for the fitted values on the predictor. Since both lines match very closely the model can be determined to be adequate.



Conclusion

Table 21: Model 10: Confusion Matrix

	True Below Median	True Above Median
Predicted Below Median	41	4
Predicted Above Median	9	40

Looking at the confusion matrix and the diagnostic table we are happy with our chosen model. Yes, the precision is lower than some of the other models, but there were only 94 rows in our test data, with more data we might have been able to balance out the precision scores. Additionally, the AUC and AIC scores are not that much lower than the other models we looked at. Furthermore, our sensitivity is still above 90%. Therefore, we are happy with model 10 as our final model.

Diagnostic	Model 10
accuracy	0.86170
error_rate	0.13830
precision	0.81633
sensitivity	0.90909
specificity	0.82000
F1_score	0.86022
auc	0.86455

Predicting evaluation data

Table 23: Evaluation Data Predicted

zn	indus	chas	nox	rm	age	dis	rad	tax	ptratio	lstat	medv	target	target_prob
0	2.89	0	0.445	6.163	69.6	3.4952	2	276	18.0	11.34	21.4	0	0.001
0	18.10	0	0.655	6.209	65.4	2.9634	24	666	20.2	13.22	21.4	1	1.000
0	7.38	0	0.493	6.312	28.9	5.4159	5	287	19.6	6.15	23.0	0	0.355
0	19.58	0	0.605	6.101	93.0	2.2834	5	403	14.7	9.81	25.0	1	0.971
0	9.90	0	0.544	6.122	52.8	2.6403	4	304	18.4	5.98	22.1	0	0.122
33	2.18	0	0.472	6.616	58.1	3.3700	7	222	18.4	8.93	28.4	0	0.007
25	5.13	0	0.453	5.741	66.2	7.2254	8	284	19.7	13.15	18.7	0	0.045
0	8.14	0	0.538	5.950	82.0	3.9900	4	307	21.0	27.71	13.2	0	0.117
0	18.10	0	0.713	6.525	86.5	2.4358	24	666	20.2	18.13	14.1	1	1.000
0	18.10	0	0.740	6.219	100.0	2.0048	24	666	20.2	16.59	18.4	1	1.000