

Predicting Boston Housing Prices

Akinyemi Apampao, xxxx

David Fakolujo, xxxx

Joshua Ogunbo, xxxx

Prince Oloma, xxxx

Ravin Jayasuriya, xxxx

1. INTRODUCTION

1.1.1 Context

This project falls within the domains of real estate analytics, urban economics, and urban planning. It focuses on understanding the factors that influence housing prices in Boston by analyzing a dataset derived from a census survey conducted in the 1970s. The dataset includes 13 features that may impact the value of homes in different neighborhoods. Our goal is to build regression models to predict the median value of owner-occupied homes (measured in thousands of dollars) and to identify which features have the most significant effect on housing prices.

1.1.2 Problem

The problem we aim to address is the difficulty in understanding which features most influence housing prices in Boston. Without clear insights into what drives property values, it becomes challenging for prospective buyers, analysts, or stakeholders to make informed decisions. Our objective is to develop a predictive model that estimates housing prices and helps identify the key factors contributing to those predictions.

1.1.3 Challenges

// ADD Data normality issue

1.2 OBJECTIVES

1.2.1 Overview

The housing market has been unpredictable in recent years, with prices rising in many areas and making it harder for people, especially young or first-time buyers, to afford a home. In this project, we're working with housing data from Boston to build a model that can predict house prices. By exploring which features of a home are most strongly linked to its value, we hope to better understand what drives housing prices and help future buyers know what to look for.

1.2.2 Goals & Research Questions

The primary goal of this project is to build a predictive model that accurately estimates housing prices in Boston based on various property features. In doing so, we aim to uncover which features most strongly influence a home's value.

To guide this objective, we explore the following research questions:

- Can we develop a reliable model to predict the median value of homes in Boston?
- Which features are the most important in influencing housing prices?

2. METHODOLOGY

2.1 Data

The dataset used in this project consists of housing data collected from the Boston Standard Metropolitan Statistical Area (SMSA) in the 1970s. It contains 506 entries and includes 11 qualitative independent variables, 2 quantitative independent variables, and 1 quantitative dependent variable.

The dataset was originally collected as part of a census report and is considered open data. It is publicly available at:

<https://lib.stat.cmu.edu/datasets/boston>

Below is a brief description of each variable:

- **CRIM:** Per capita crime rate by town. Indicates the level of crime in the area.
- **ZN:** Proportion of residential land zoned for lots over 25,000 sq.ft. Reflects residential density.
- **INDUS:** Proportion of non-retail business acres per town. Indicates commercial land usage.
- **CHAS:** Charles River dummy variable (1 if tract bounds river; 0 otherwise). Indicates proximity to the Charles River.
- **NOX:** Nitric oxides concentration (parts per 10 million). Represents industrial pollution.
- **RM:** Average number of rooms per dwelling. Suggests spaciousness.
- **AGE:** Proportion of owner-occupied units built prior to 1940. Reflects the age of buildings in the area.
- **DIS:** Weighted distances to five Boston employment centres. Measures accessibility to work locations.
- **RAD:** Index of accessibility to radial highways. Higher values indicate better road access.
- **TAX:** Full-value property-tax rate per \$10,000. Indicates the annual property tax burden.
- **PTRATIO:** Pupil-teacher ratio by town. Lower values suggest better educational facilities.
- **B:** $1000(\text{Bk} - 0.63)^2$, where Bk is the proportion of Black residents by town.
- **LSTAT:** Percentage of the population considered lower status.
- **MEDV:** Median value of owner-occupied homes in \$1000s. This is the dependent variable we aim to predict.^[1]

2.2 Approach

In this project, we use a **predictive modeling approach** based on **multiple linear regression** to estimate the median value of homes in Boston. This method is well-suited for our goal of understanding how different housing features influence prices while also producing accurate predictions.

We believe multiple linear regression is an effective choice for this project for several reasons:

- **Interpretability:** The model provides clear and meaningful insights into how each variable affects housing prices, which is valuable for both analysis and decision-making.
- **No Multicollinearity:** VIF results confirmed the absence of multicollinearity, supporting the reliability and stability of the regression coefficients.
- **Well-Structured Data:** The dataset consists of numeric features that align well with the assumptions of linear regression, making it a natural modeling choice.
- **Proven Technique:** Linear regression is a widely accepted method in real estate analytics, with a long history of successful application in similar predictive tasks.
- **Strong Baseline:** This approach serves as a solid baseline for future comparisons with more complex models if needed, offering a balance of performance and simplicity.

2.3 Workflows

Below are the key steps:

A. Test for Multicollinearity B. Create the Best Additive Model C. Create the Best Interaction Model D. Explore Higher Order Terms E. Test Multiple Regression Assumptions (linearity, independence, equal variance, normality, outliers)

2.4 Contributions

1. Akinyemi Apampa -Introduction writing: Context, Problem, Challenges -Initial data preprocessing and cleaning -Creating the additive regression model and interpreting results -Assisting with final report compilation and proofreading
2. Ravin Jayasuriya -Multicollinearity testing and VIF analysis -Stepwise regression and reduced additive model selection -Writing and interpreting results of the additive model (t-tests, F-tests) -Assisting with final report compilation and proofreading
3. Joshua Ogunbo -Developing and refining the full and reduced interaction models -Conducting and interpreting ANOVA tests for interaction models -Writing the section on interaction terms and model selection justification -Assisting with final report compilation and proofreading
4. Prince Oloma Eworitsemoghan -Higher-order term exploration and model building (crim, zn, nox, rm, dis, rad, tax, lstat) -Identifying and incorporating significant polynomial terms -Writing the section on higher-order terms and their interpretation -Assisting with final report compilation and proofreading
5. David Fakojulo -Addressing model assumption violations: log, Box-Cox, and WLS transformations -Finalizing the best regression model using weighted least squares -Performing and interpreting Shapiro-Wilk and Breusch-Pagan tests -Assisting with final report compilation and proofreading

3. MAIN RESULT OF THE ANALYSIS

Data Import and Initial Inspection

Table 1: First Three Rows of the Boston Housing Dataset

crim	zn	indus	chas	nox	rm	age	dis	rad	tax	ptratio	b	lstat	medv
0.00632	18	2.31	0	0.538	6.575	65.2	4.0900	1	296	15.3	396.90	4.98	24.0
0.02731	0	7.07	0	0.469	6.421	78.9	4.9671	2	242	17.8	396.90	9.14	21.6
0.02729	0	7.07	0	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7

Recoding Categorical Variables for Model Compatibility

Table 2: First Three Rows of the Boston Housing Dataset

crim	zn	indus	chas	nox	rm	age	dis	rad	tax	ptratio	b	lstat	medv
0.00632	18	2.31	No	0.538	6.575	65.2	4.0900	1	296	15.3	396.90	4.98	24.0
0.02731	0	7.07	No	0.469	6.421	78.9	4.9671	2	242	17.8	396.90	9.14	21.6
0.02729	0	7.07	No	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7

A. Test for Multicollinearity

From the below, multicollinearity was not detected for any of the variables.

```
boston_additive <- lm(medv ~ crim + zn + indus + chas + nox + rm + age + dis + rad + tax + ptratio + b
```

Table 3: TEST FOR MULTICOLLINEARITY

Variable.Name	VIF	Detection
crim	1.7922	0
zn	2.2928	0
indus	3.9916	0
chas	1.0740	0
nox	4.3937	0
rm	1.9337	0
age	3.1008	0
dis	3.9559	0
rad	7.4845	0
tax	9.0086	0
ptratio	1.7991	0
b	1.3485	0
lstat	2.9415	0

The **Variance Inflation Factor (VIF)** values for all predictors in the model are below the commonly used threshold of **10**, indicating that **multicollinearity is not a concern**.

B. Create the Best Additive Model

We perform individual t -tests to assess the significance of each predictor in the model. The hypotheses are as follows:

- **Null Hypothesis:**

$$H_0 : \beta_1 = \beta_2 = \dots = \beta_p = 0$$

- **Alternative Hypothesis:**

$$H_a : \text{At least one } \beta_i \neq 0, \quad \text{where } i = 1, 2, \dots, p$$

These tests help determine whether each predictor contributes significantly to explaining the variability in the response variable.

Table 4: Coefficient Estimates from the Additive Model

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	36.4595	5.1035	7.1441	0.0000
crim	-0.1080	0.0329	-3.2865	0.0011
zn	0.0464	0.0137	3.3816	0.0008
indus	0.0206	0.0615	0.3343	0.7383
chasYes	2.6867	0.8616	3.1184	0.0019
nox	-17.7666	3.8197	-4.6513	0.0000
rm	3.8099	0.4179	9.1161	0.0000
age	0.0007	0.0132	0.0524	0.9582
dis	-1.4756	0.1995	-7.3980	0.0000
rad	0.3060	0.0663	4.6129	0.0000

	Estimate	Std. Error	t value	Pr(> t)
tax	-0.0123	0.0038	-3.2800	0.0011
ptratio	-0.9527	0.1308	-7.2825	0.0000
b	0.0093	0.0027	3.4668	0.0006
lstat	-0.5248	0.0507	-10.3471	0.0000

From the regression output, we observe:

- **Significant predictors** (p-value < 0.05): *crim*, *zn*, *chas*, *nox*, *rm*, *dis*, *rad*, *tax*, *ptratio*, *b*, *lstat*
- **Insignificant predictors**: *indus* (p = 0.7383) and *age* (p = 0.9582)

Because the p-values of *indus* and *age* are greater than 0.05, we fail to reject the null hypotheses that their coefficients are zero. Thus, these variables do not significantly contribute to the prediction of *medv* in the presence of other variables.

The fitted multiple linear regression model is:

$$\begin{aligned}\hat{medv} = & 3.646 - 0.108 \cdot crim + 0.04642 \cdot zn \\ & + 0.02056 \cdot indus + 2.687 \cdot chas - 17.7666 \cdot nox \\ & + 3.8099 \cdot rm + 0.0007 \cdot age - 1.4756 \cdot dis \\ & + 0.3060 \cdot rad - 0.0123 \cdot tax - 0.9527 \cdot ptratio \\ & + 0.0093 \cdot b - 0.5248 \cdot lstat\end{aligned}$$

Building the Reduced Additive model

We remove the variables *indus* and *age*, which were found to be statistically insignificant in the full model, and test the following hypotheses:

- **Null Hypothesis:**

$$H_0 : \beta_{indus} = \beta_{age} = 0$$

- **Alternative Hypothesis:**

$$H_a : \text{At least one of } \beta_{indus}, \beta_{age} \neq 0$$

A high p-value would indicate that removing *indus* and *age* does not significantly worsen the model, and thus the reduced model is preferred for its simplicity.

```
reduced_additive_model <- lm(
  formula = medv ~ crim + zn + factor(chas) + nox + rm +
    dis + rad + tax + ptratio + b + lstat,
  data = boston_data
)
```

Table 5: Coefficient Estimates from the Reduced Additive Model

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	36.3411	5.0675	7.1714	0.0000
crim	-0.1084	0.0328	-3.3074	0.0010
zn	0.0458	0.0135	3.3902	0.0008
factor(chas)Yes	2.7187	0.8542	3.1826	0.0016
nox	-17.3760	3.5352	-4.9151	0.0000
rm	3.8016	0.4063	9.3562	0.0000
dis	-1.4927	0.1857	-8.0370	0.0000
rad	0.2996	0.0634	4.7255	0.0000
tax	-0.0118	0.0034	-3.4925	0.0005
ptratio	-0.9465	0.1291	-7.3337	0.0000
b	0.0093	0.0027	3.4746	0.0006
lstat	-0.5226	0.0474	-11.0187	0.0000

The reduced model shows that **all included variables have p-values less than 0.05**, indicating that they are statistically significant.

We will now run a **global F-test (ANOVA)** to assess whether removing the variables **indus** and **age** significantly worsens the model fit.

Table 6: ANOVA Comparison of Reduced and Full Additive Model

Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
494	11081.36	NA	NA	NA	NA
492	11078.78	2	2.5794	0.0573	0.9443

Since the p-value is **0.9443**, which is much greater than the threshold of **0.05**, we **fail to reject the null hypothesis**. This indicates that **indus** and **age** do not significantly improve the model. Hence, the reduced model is more appropriate as it maintains model quality while eliminating unnecessary predictors.

Stepwise Model Selection**

To further validate the choice of predictors and identify the most parsimonious model, we used the `ols_step_both_p()` function from the **olsrr** package to perform stepwise selection based on p-values.

Table 7: Coefficient Estimates from the Stepwise-Selected Model

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	36.3411	5.0675	7.1714	0.0000
lstat	-0.5226	0.0474	-11.0187	0.0000
rm	3.8016	0.4063	9.3562	0.0000
ptratio	-0.9465	0.1291	-7.3337	0.0000
dis	-1.4927	0.1857	-8.0370	0.0000
nox	-17.3760	3.5352	-4.9151	0.0000
chasYes	2.7187	0.8542	3.1826	0.0016
b	0.0093	0.0027	3.4746	0.0006
zn	0.0458	0.0135	3.3902	0.0008
crim	-0.1084	0.0328	-3.3074	0.0010

	Estimate	Std. Error	t value	Pr(> t)
rad	0.2996	0.0634	4.7255	0.0000
tax	-0.0118	0.0034	-3.4925	0.0005

The selected model is the same as the reduced model obtained earlier. Hence, the **final additive model** is:

$$\begin{aligned}\hat{medv}_i = & 36.3411 - 0.5226 \cdot lstat_i + 3.8016 \cdot rm_i \\ & - 0.9465 \cdot ptratio_i - 1.4927 \cdot dis_i - 17.3760 \cdot nox_i + 2.7187 \cdot chas_i \\ & + 0.0093 \cdot b_i + 0.0458 \cdot zn_i - 0.1084 \cdot crim_i + 0.2996 \cdot rad_i - 0.0118 \cdot tax_i\end{aligned}$$

Where:

$$chas_i = 1 \quad \text{if the tract bounds the Charles River, and 0 otherwise}$$

C. Create the Best Interaction Model

A full two-way interaction model was constructed by including all possible interaction terms among the predictors in the final additive model obtained from stepwise selection.

Table 8: Coefficient Estimates from the Full Two-Way Interaction Model

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-61.6850	56.6084	-1.0897	0.2765
crim	-7.4081	5.9098	-1.2535	0.2107
zn	-0.2549	0.4276	-0.5960	0.5515
chasYes	42.5330	18.6500	2.2806	0.0231
nox	-43.7416	56.4059	-0.7755	0.4385
rm	21.1759	5.1167	4.1386	0.0000
dis	-7.8290	4.0307	-1.9423	0.0527
rad	4.6098	2.1044	2.1905	0.0290
tax	-0.1399	0.0953	-1.4684	0.1427
ptratio	2.5864	2.3119	1.1188	0.2639
b	0.0897	0.0754	1.1903	0.2346
lstat	2.1385	0.8089	2.6437	0.0085
crim:zn	0.3038	0.1600	1.8983	0.0583
crim:chasYes	2.5732	0.6020	4.2748	0.0000
crim:nox	-1.9333	0.9640	-2.0054	0.0455
crim:rm	0.1951	0.0503	3.8746	0.0001
crim:dis	-0.1912	0.0941	-2.0321	0.0427
crim:rad	-0.5192	0.1850	-2.8067	0.0052
crim:tax	0.0340	0.0108	3.1494	0.0017
crim:ptratio	-0.1428	0.2304	-0.6198	0.5357
crim:b	-0.0004	0.0002	-2.5069	0.0125
crim:lstat	0.0239	0.0069	3.4591	0.0006
zn:chasYes	-0.0448	0.0540	-0.8286	0.4078
zn:nox	-0.4167	0.3969	-1.0499	0.2943
zn:rm	0.0157	0.0246	0.6358	0.5252
zn:dis	0.0138	0.0058	2.3906	0.0172
zn:rad	-0.0050	0.0067	-0.7493	0.4541
zn:tax	0.0004	0.0002	2.4499	0.0147
zn:ptratio	0.0012	0.0062	0.1916	0.8482

	Estimate	Std. Error	t value	Pr(> t)
zn:b	0.0004	0.0007	0.5477	0.5842
zn:lstat	-0.0054	0.0039	-1.3666	0.1725
chasYes:nox	-31.0752	12.3380	-2.5187	0.0121
chasYes:rm	-4.3065	1.1439	-3.7649	0.0002
chasYes:dis	0.5251	1.3608	0.3858	0.6998
chasYes:rad	-0.5426	0.4017	-1.3507	0.1775
chasYes:tax	0.0230	0.0260	0.8830	0.3777
chasYes:ptratio	-0.5936	0.6712	-0.8843	0.3770
chasYes:b	0.0251	0.0154	1.6298	0.1039
chasYes:lstat	-0.3566	0.1681	-2.1213	0.0345
nox:rm	3.2483	5.2601	0.6175	0.5372
nox:dis	4.1538	2.9521	1.4071	0.1601
nox:rad	-2.3556	1.2452	-1.8918	0.0592
nox:tax	0.1479	0.0692	2.1370	0.0331
nox:ptratio	-1.4380	2.0307	-0.7081	0.4793
nox:b	-0.0556	0.0356	-1.5616	0.1191
nox:lstat	0.3077	0.6089	0.5054	0.6135
rm:dis	0.5584	0.2886	1.9352	0.0536
rm:rad	-0.1058	0.1229	-0.8611	0.3896
rm:tax	-0.0102	0.0067	-1.5122	0.1312
rm:ptratio	-0.5236	0.2146	-2.4400	0.0151
rm:b	-0.0072	0.0035	-2.0529	0.0407
rm:lstat	-0.3198	0.0436	-7.3395	0.0000
dis:rad	-0.1617	0.0551	-2.9348	0.0035
dis:tax	0.0003	0.0023	0.1085	0.9137
dis:ptratio	0.1152	0.0909	1.2675	0.2056
dis:b	-0.0022	0.0055	-0.3951	0.6929
dis:lstat	0.0778	0.0388	2.0053	0.0455
rad:tax	-0.0003	0.0006	-0.5635	0.5734
rad:ptratio	-0.0274	0.0799	-0.3430	0.7318
rad:b	-0.0018	0.0022	-0.8171	0.4143
rad:lstat	-0.0196	0.0147	-1.3373	0.1818
tax:ptratio	0.0035	0.0020	1.7336	0.0837
tax:b	0.0001	0.0002	0.9948	0.3204
tax:lstat	-0.0015	0.0009	-1.6672	0.0962
ptratio:b	-0.0014	0.0025	-0.5611	0.5750
ptratio:lstat	0.0050	0.0260	0.1909	0.8487
b:lstat	-0.0013	0.0004	-3.0059	0.0028

A full two-way interaction model was constructed by including all possible interaction terms among the predictors in the final additive model obtained from stepwise selection.

The full interaction model includes several interaction terms with **p-values greater than 0.05**, indicating that they are not statistically significant. These insignificant interactions were dropped to create a **reduced interaction model**.

Table 9: Coefficient Estimates from the Reduced Interaction Model

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-90.6158	17.8213	-5.0847	0.0000
crim	-9.2738	2.7963	-3.3164	0.0010
zn	-0.1636	0.0494	-3.3130	0.0010

	Estimate	Std. Error	t value	Pr(> t)
chasYes	45.8055	9.1403	5.0114	0.0000
nox	-24.4974	13.2152	-1.8537	0.0644
rm	21.9919	2.4669	8.9148	0.0000
dis	-8.6346	1.2444	-6.9387	0.0000
rad	2.2883	0.4760	4.8077	0.0000
tax	-0.0632	0.0229	-2.7586	0.0060
ptratio	5.3595	0.7317	7.3245	0.0000
b	0.0542	0.0249	2.1811	0.0297
lstat	1.6647	0.2839	5.8644	0.0000
crim:zn	0.2722	0.1074	2.5353	0.0116
crim:chasYes	2.0022	0.3306	6.0571	0.0000
crim:nox	0.4397	0.8537	0.5150	0.6068
crim:rm	0.0882	0.0486	1.8136	0.0704
crim:dis	-0.1332	0.0849	-1.5693	0.1173
crim:rad	-0.5622	0.1381	-4.0709	0.0001
crim:tax	0.0332	0.0089	3.7340	0.0002
crim:b	-0.0003	0.0002	-1.8384	0.0666
crim:lstat	-0.0044	0.0061	-0.7090	0.4787
zn:dis	0.0165	0.0043	3.8431	0.0001
zn:tax	0.0002	0.0001	1.5833	0.1140
chasYes:nox	-30.4724	6.0576	-5.0304	0.0000
chasYes:rm	-4.1716	1.1015	-3.7871	0.0002
chasYes:lstat	-0.2236	0.1440	-1.5524	0.1212
nox:rad	-2.5610	0.7488	-3.4202	0.0007
nox:tax	0.0802	0.0416	1.9281	0.0544
rm:dis	1.0300	0.1710	6.0237	0.0000
rm:ptratio	-0.9099	0.1138	-7.9928	0.0000
rm:b	-0.0029	0.0035	-0.8188	0.4133
rm:lstat	-0.3691	0.0404	-9.1312	0.0000
dis:rad	-0.1151	0.0322	-3.5722	0.0004
dis:lstat	0.1614	0.0216	7.4692	0.0000
b:lstat	-0.0015	0.0004	-3.9483	0.0001

The output of the reduced interactive model showed more insignificant interactions, which were further dropped.

Table 10: Table: Coefficient Estimates from the Final Reduced Interaction Model

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-75.3241	14.9000	-5.0553	0.0000
crim	-9.5963	2.7403	-3.5019	0.0005
zn	-0.0998	0.0313	-3.1898	0.0015
chasYes	38.2243	6.8141	5.6096	0.0000
nox	-17.5242	11.4888	-1.5253	0.1278
rm	19.3263	2.0435	9.4576	0.0000
dis	-7.8883	1.1719	-6.7312	0.0000
rad	2.1207	0.3763	5.6351	0.0000
tax	-0.0516	0.0180	-2.8597	0.0044
ptratio	4.7316	0.7148	6.6191	0.0000
b	0.0292	0.0062	4.6879	0.0000

	Estimate	Std. Error	t value	Pr(> t)
lstat	1.2139	0.2083	5.8276	0.0000
crim:zn	0.2777	0.1071	2.5937	0.0098
crim:chasYes	2.2138	0.3262	6.7860	0.0000
crim:rad	-0.5802	0.1397	-4.1522	0.0000
crim:tax	0.0351	0.0090	3.9025	0.0001
zn:dis	0.0162	0.0043	3.7206	0.0002
chasYes:nox	-33.5288	5.9660	-5.6200	0.0000
chasYes:rm	-3.1508	0.7824	-4.0269	0.0001
nox:rad	-2.3342	0.6145	-3.7983	0.0002
nox:tax	0.0627	0.0342	1.8329	0.0674
rm:dis	0.9348	0.1624	5.7565	0.0000
rm:ptratio	-0.8125	0.1115	-7.2861	0.0000
rm:lstat	-0.3117	0.0346	-9.0205	0.0000
dis:rad	-0.1304	0.0268	-4.8642	0.0000
dis:lstat	0.1592	0.0201	7.9221	0.0000
b:lstat	-0.0012	0.0003	-4.1913	0.0000

The further reduced model then showed nox:tax as insignificant, which was also dropped.

Table 11: Table: Coefficient Estimates from the Third Reduced Interaction Model

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-83.3729	14.2731	-5.8413	0.0000
crim	-11.1718	2.6084	-4.2829	0.0000
zn	-0.0897	0.0309	-2.9052	0.0038
chasYes	38.1964	6.8308	5.5918	0.0000
nox	1.2029	5.2663	0.2284	0.8194
rm	19.0011	2.0407	9.3109	0.0000
dis	-8.0659	1.1708	-6.8895	0.0000
rad	1.6045	0.2502	6.4124	0.0000
tax	-0.0190	0.0030	-6.4174	0.0000
ptratio	4.6552	0.7154	6.5072	0.0000
b	0.0292	0.0063	4.6741	0.0000
lstat	1.2469	0.2080	5.9935	0.0000
crim:zn	0.2484	0.1061	2.3409	0.0196
crim:chasYes	2.2155	0.3270	6.7746	0.0000
crim:rad	-0.6602	0.1331	-4.9616	0.0000
crim:tax	0.0403	0.0085	4.7219	0.0000
zn:dis	0.0149	0.0043	3.4648	0.0006
chasYes:nox	-33.6871	5.9800	-5.6333	0.0000
chasYes:rm	-3.1157	0.7841	-3.9736	0.0001
nox:rad	-1.3857	0.3323	-4.1705	0.0000
rm:dis	0.9675	0.1618	5.9796	0.0000
rm:ptratio	-0.7967	0.1115	-7.1481	0.0000
rm:lstat	-0.3161	0.0346	-9.1449	0.0000
dis:rad	-0.1239	0.0266	-4.6515	0.0000
dis:lstat	0.1581	0.0201	7.8530	0.0000
b:lstat	-0.0012	0.0003	-4.1852	0.0000

Now that all interactions were significant, an f-test was run to compare reduced interactive model and full

interaction model

Table 12: ANOVA Comparison of Third Reduced Interaction Model and Full Interaction Model

Res.Df	RSS	Df	Sum.of.Sq	F	Pr..F.
480	5573.760	NA	NA	NA	NA
439	4159.123	41	1414.638	3.641868	0

Based on the summary above, the p-value is $8.242e - 12 < 0.05$, suggesting the null hypothesis should be rejected. Also, the adjusted R^2_{adj} and RSE of the full interaction model are 0.888 and 3.078 respectively, while the adjusted R^2_{adj} and RSE of the reduced interaction model are 0.8627 and 3.408 respectively.

These suggest the full interaction model should be preferred. However, the full interaction model has a number of insignificant interactions, while the reduced interaction model has only significant interactions. Even though the anova test, adjusted R^2_{adj} and RSE suggest preferring the full interaction model, there isn't a major difference between the adjusted R^2_{adj} and RSE of the two models.

We would choose the reduced model because it retains all significant interactions while eliminating insignificant ones, ensuring better interpretability and avoiding unnecessary complexity without a substantial loss in explanatory power.

D. Explore Higher Order Terms

To check for possible higher order relationships, we explored all pairwise combinations of continuous variables in scatterplots to see how the response variable looked with respect to each of the continuous additive predictors

```
higher_order_data = data.frame(boston_data$medv, boston_data$crim, boston_data$zn, boston_data$chas, boston_data$nox, boston_data$rm)

ggpairs(higher_order_data, lower = list(continuous = wrap("smooth_loess", color = "red"), combo = "facet_wrap"))

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : pseudoinverse used at -0.5

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : neighborhood radius 13

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : reciprocal condition number 0

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : There are other near singularities as well. 156.25

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : pseudoinverse used at
## -0.5
```

```

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : neighborhood radius 13

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : reciprocal condition
## number 0

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : There are other near
## singularities as well. 156.25

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : pseudoinverse used at -0.5

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : neighborhood radius 13

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : reciprocal condition number 0

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : There are other near singularities as well. 156.25

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : pseudoinverse used at
## -0.5

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : neighborhood radius 13

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : reciprocal condition
## number 0

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : There are other near
## singularities as well. 156.25

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : pseudoinverse used at -0.5

```

```

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : neighborhood radius 13

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : reciprocal condition number 0

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : There are other near singularities as well. 156.25

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : pseudoinverse used at
## -0.5

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : neighborhood radius 13

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : reciprocal condition
## number 0

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : There are other near
## singularities as well. 156.25

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : pseudoinverse used at -0.5

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : neighborhood radius 13

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : reciprocal condition number 0

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : There are other near singularities as well. 156.25

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : pseudoinverse used at
## -0.5

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : neighborhood radius 13

```

```

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : reciprocal condition
## number 0

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : There are other near
## singularities as well. 156.25

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : pseudoinverse used at -0.5

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : neighborhood radius 13

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : reciprocal condition number 0

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : There are other near singularities as well. 156.25

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : pseudoinverse used at
## -0.5

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : neighborhood radius 13

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : reciprocal condition
## number 0

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : There are other near
## singularities as well. 156.25

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : pseudoinverse used at -0.5

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : neighborhood radius 13

```

```

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : reciprocal condition number 0

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : There are other near singularities as well. 156.25

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : pseudoinverse used at
## -0.5

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : neighborhood radius 13

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : reciprocal condition
## number 0

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : There are other near
## singularities as well. 156.25

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : pseudoinverse used at -0.5

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : neighborhood radius 13

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : reciprocal condition number 0

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : There are other near singularities as well. 156.25

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : pseudoinverse used at
## -0.5

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : neighborhood radius 13

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : reciprocal condition
## number 0

```



```

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : There are other near
## singularities as well. 156.25

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : pseudoinverse used at -0.5

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : neighborhood radius 13

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : reciprocal condition number 0

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = parametric,
## : There are other near singularities as well. 156.25

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : pseudoinverse used at
## -0.5

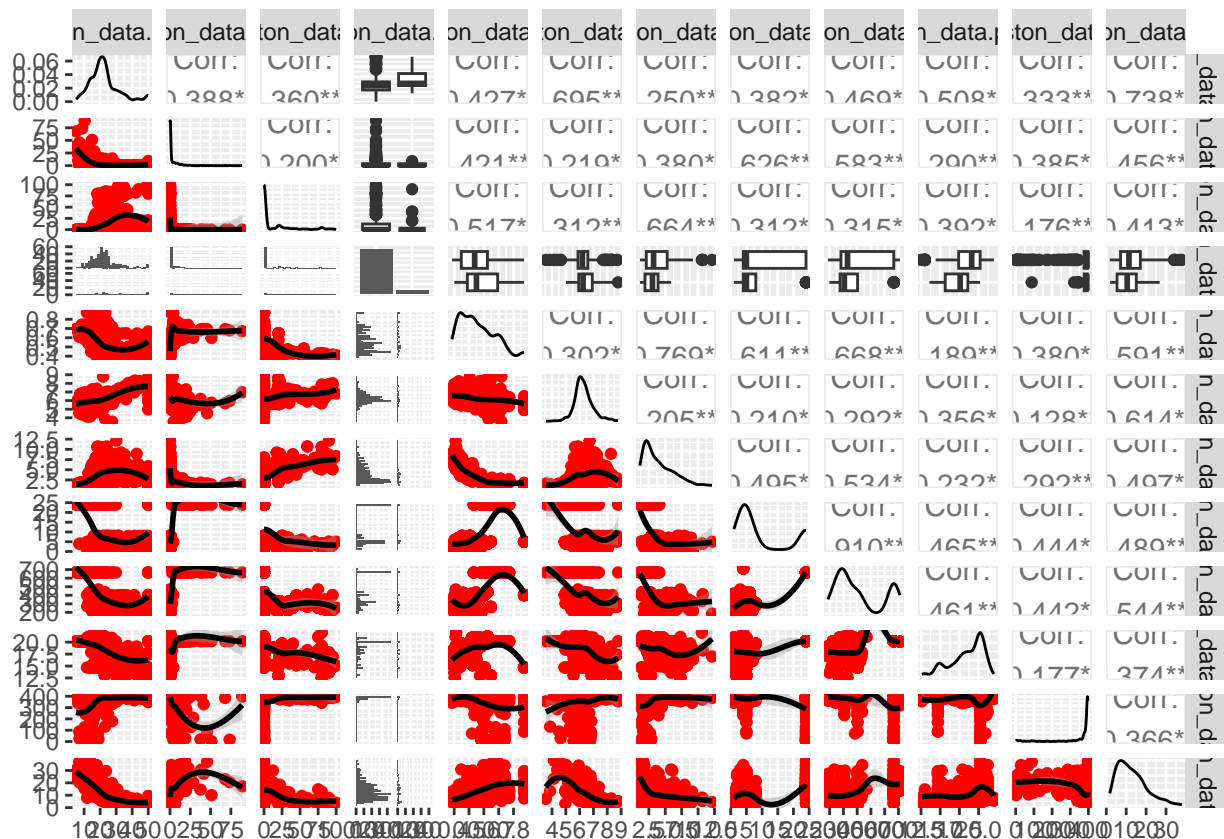
## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : neighborhood radius 13

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : reciprocal condition
## number 0

## Warning in predLoess(object$y, object$x, newx = if (is.null(newdata)) object$x
## else if (is.data.frame(newdata))
## as.matrix(model.frame(delete.response(terms(object))), : There are other near
## singularities as well. 156.25

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

```



It looks like the variables that might be worth exploring for possible higher order relationships with medv are crim, zn, nox, rm, dis, rad, tax, lstat. Each of these variables were tested as follows:

Possible higher order relationship exploration for crim

```
higher_order_crim_2 = lm(formula = medv ~ crim + I(crim^2) + zn + chas + nox + rm + dis + rad + tax + ptratio, data = boston_data)
summary(higher_order_crim_2)
```

```
##
## Call:
## lm(formula = medv ~ crim + I(crim^2) + zn + chas + nox + rm +
##     dis + rad + tax + ptratio + b + lstat + crim:zn + crim:chas +
##     crim:rad + crim:tax + zn:dis + chas:nox + chas:rm + nox:rad +
##     rm:dis + rm:ptratio + rm:lstat + dis:rad + dis:lstat + b:lstat,
##     data = boston_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.7617  -1.8923  -0.0132   1.7740  21.2679
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -8.701e+01  1.409e+01  -6.176 1.40e-09 ***
```

```
## crim      -1.145e+01  2.570e+00 -4.456 1.04e-05 ***
## I(crim^2)   3.972e-03  1.002e-03   3.966 8.43e-05 ***
## zn        -8.964e-02  3.041e-02 -2.948 0.003357 **
## chasYes     3.797e+01  6.729e+00   5.643 2.86e-08 ***
## nox         6.202e-01  5.189e+00   0.120 0.904927
## rm          1.950e+01  2.014e+00   9.680 < 2e-16 ***
## dis        -7.743e+00  1.156e+00 -6.697 5.97e-11 ***
## rad          1.837e+00  2.533e-01   7.250 1.68e-12 ***
## tax        -1.975e-02  2.918e-03 -6.769 3.80e-11 ***
## ptratio     4.781e+00  7.054e-01   6.779 3.58e-11 ***
## b           2.820e-02  6.162e-03   4.576 6.05e-06 ***
## lstat       1.411e+00  2.091e-01   6.750 4.28e-11 ***
## crim:zn     2.626e-01  1.046e-01   2.511 0.012370 *
## crim:chasYes 2.141e+00  3.227e-01   6.636 8.75e-11 ***
## crim:rad    -6.820e-01  1.312e-01 -5.199 2.98e-07 ***
## crim:tax     4.108e-02  8.415e-03   4.882 1.43e-06 ***
## zn:dis       1.449e-02  4.237e-03   3.419 0.000683 ***
## chasYes:nox -3.314e+01  5.892e+00 -5.624 3.17e-08 ***
## chasYes:rm  -3.138e+00  7.724e-01 -4.063 5.66e-05 ***
## nox:rad      -1.429e+00  3.275e-01 -4.364 1.56e-05 ***
## rm:dis        9.558e-01  1.594e-01   5.996 3.98e-09 ***
## rm:ptratio   -8.159e-01  1.099e-01 -7.424 5.23e-13 ***
## rm:lstat     -3.347e-01  3.437e-02 -9.740 < 2e-16 ***
## dis:rad      -1.602e-01  2.779e-02 -5.766 1.46e-08 ***
## dis:lstat     1.487e-01  1.997e-02   7.443 4.60e-13 ***
## b:lstat      -1.249e-03  2.912e-04 -4.287 2.19e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.357 on 479 degrees of freedom
## Multiple R-squared:  0.8737, Adjusted R-squared:  0.8668
## F-statistic: 127.4 on 26 and 479 DF,  p-value: < 2.2e-16
```

```
higher_order_crim_3 = lm(formula = medv ~ crim + I(crim^2) + I(crim^3) + zn + chas + nox + rm + dis + rad,
summary(higher_order_crim_3)
```

```
##
## Call:
## lm(formula = medv ~ crim + I(crim^2) + I(crim^3) + zn + chas +
##     nox + rm + dis + rad + tax + ptratio + b + lstat + crim:zn +
##     crim:chas + crim:rad + crim:tax + zn:dis + chas:nox + chas:rm +
##     nox:rad + rm:dis + rm:ptratio + rm:lstat + dis:rad + dis:lstat +
##     b:lstat, data = boston_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.496  -1.897  -0.030   1.714   21.192
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -8.834e+01  1.407e+01  -6.279 7.67e-10 ***
## crim         -1.163e+01  2.565e+00  -4.534 7.32e-06 ***
## I(crim^2)      1.305e-02  4.937e-03   2.643 0.008488 **
## I(crim^3)     -7.637e-05  4.068e-05  -1.877 0.061088 .
```

```
## zn          -8.873e-02  3.033e-02  -2.925  0.003603 **
## chasYes     3.843e+01  6.715e+00   5.722  1.86e-08 ***
## nox         4.730e-01  5.176e+00   0.091  0.927233
## rm         1.954e+01  2.009e+00   9.726  < 2e-16 ***
## dis        -7.547e+00  1.158e+00  -6.519  1.80e-10 ***
## rad         1.965e+00  2.618e-01   7.507  2.98e-13 ***
## tax        -1.998e-02  2.913e-03  -6.861  2.12e-11 ***
## ptratio     4.784e+00  7.035e-01   6.800  3.14e-11 ***
## b          2.917e-02  6.168e-03   4.729  2.98e-06 ***
## lstat       1.411e+00  2.085e-01   6.765  3.91e-11 ***
## crim:zn      2.638e-01  1.043e-01   2.528  0.011778 *
## crim:chasYes 2.122e+00  3.220e-01   6.592  1.15e-10 ***
## crim:rad     -7.056e-01  1.314e-01  -5.368  1.24e-07 ***
## crim:tax      4.181e-02  8.402e-03   4.977  9.04e-07 ***
## zn:dis       1.427e-02  4.227e-03   3.377  0.000793 ***
## chasYes:nox  -3.330e+01  5.877e+00  -5.666  2.53e-08 ***
## chasYes:rm   -3.202e+00  7.711e-01  -4.153  3.88e-05 ***
## nox:rad      -1.448e+00  3.268e-01  -4.431  1.16e-05 ***
## rm:dis       9.433e-01  1.591e-01   5.928  5.88e-09 ***
## rm:ptratio   -8.154e-01  1.096e-01  -7.440  4.71e-13 ***
## rm:lstat     -3.286e-01  3.443e-02  -9.543  < 2e-16 ***
## dis:rad      -1.785e-01  2.938e-02  -6.076  2.51e-09 ***
## dis:lstat     1.451e-01  2.001e-02   7.251  1.68e-12 ***
## b:lstat      -1.286e-03  2.912e-04  -4.418  1.23e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.348 on 478 degrees of freedom
## Multiple R-squared:  0.8746, Adjusted R-squared:  0.8675
## F-statistic: 123.5 on 27 and 478 DF,  p-value: < 2.2e-16
```

Possible higher order relationship exploration for zn

```
higher_order_zn_2 = lm(formula = medv ~ crim + zn + I(zn^2) + chas + nox + rm + dis + rad + tax + ptratio,
summary(higher_order_zn_2)
```

```
##
## Call:
## lm(formula = medv ~ crim + zn + I(zn^2) + chas + nox + rm + dis +
##     rad + tax + ptratio + b + lstat + crim:zn + crim:chas + crim:rad +
##     crim:tax + zn:dis + chas:nox + chas:rm + nox:rad + rm:dis +
##     rm:ptratio + rm:lstat + dis:rad + dis:lstat + b:lstat, data = boston_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.1579  -1.7754   0.0946   1.7047  21.7092
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -8.146e+01  1.434e+01  -5.680 2.34e-08 ***
## crim        -1.181e+01  2.653e+00  -4.451 1.07e-05 ***
```

```
## zn          -1.167e-01  3.733e-02  -3.127  0.00187 **
## I(zn^2)      4.961e-04  3.855e-04   1.287  0.19878
## chasYes      3.814e+01  6.826e+00   5.587  3.88e-08 ***
## nox          1.073e+00  5.264e+00   0.204  0.83849
## rm          1.867e+01  2.056e+00   9.082  < 2e-16 ***
## dis        -7.891e+00  1.178e+00  -6.700  5.86e-11 ***
## rad          1.617e+00  2.502e-01   6.462  2.55e-10 ***
## tax        -1.907e-02  2.954e-03  -6.455  2.65e-10 ***
## ptratio      4.537e+00  7.208e-01   6.294  6.99e-10 ***
## b           2.918e-02  6.247e-03   4.671  3.90e-06 ***
## lstat        1.251e+00  2.079e-01   6.018  3.51e-09 ***
## crim:zn      3.203e-01  1.199e-01   2.672  0.00779 **
## crim:chasYes  2.216e+00  3.268e-01   6.781  3.52e-11 ***
## crim:rad     -6.899e-01  1.350e-01  -5.112  4.62e-07 ***
## crim:tax      4.235e-02  8.679e-03   4.880  1.45e-06 ***
## zn:dis        1.240e-02  4.717e-03   2.628  0.00886 **
## chasYes:nox  -3.385e+01  5.977e+00  -5.663  2.57e-08 ***
## chasYes:rm   -3.083e+00  7.840e-01  -3.933  9.62e-05 ***
## nox:rad      -1.394e+00  3.321e-01  -4.198  3.22e-05 ***
## rm:dis        9.569e-01  1.619e-01   5.911  6.48e-09 ***
## rm:ptratio   -7.781e-01  1.123e-01  -6.928  1.38e-11 ***
## rm:lstat     -3.160e-01  3.454e-02  -9.151  < 2e-16 ***
## dis:rad      -1.247e-01  2.662e-02  -4.684  3.68e-06 ***
## dis:lstat     1.563e-01  2.017e-02   7.750  5.55e-14 ***
## b:lstat      -1.235e-03  2.954e-04  -4.181  3.46e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.405 on 479 degrees of freedom
## Multiple R-squared:  0.87, Adjusted R-squared:  0.8629
## F-statistic: 123.3 on 26 and 479 DF, p-value: < 2.2e-16
```

Possible higher order relationship exploration for nox

```
higher_order_nox_2 = lm(formula = medv ~ crim + zn + chas + nox + I(nox^2) + rm + dis + rad + tax + ptratio,
summary(higher_order_nox_2)
```

```
##
## Call:
## lm(formula = medv ~ crim + zn + chas + nox + I(nox^2) + rm +
##     dis + rad + tax + ptratio + b + lstat + crim:zn + crim:chas +
##     crim:rad + crim:tax + zn:dis + chas:nox + chas:rm + nox:rad +
##     rm:dis + rm:ptratio + rm:lstat + dis:rad + dis:lstat + b:lstat,
##     data = boston_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.1009  -1.7510  -0.0096   1.7135  21.5882
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept) -7.818e+01 1.444e+01 -5.413 9.82e-08 ***
## crim -1.051e+01 2.619e+00 -4.012 6.98e-05 ***
## zn -1.112e-01 3.247e-02 -3.426 0.000666 ***
## chasYes 3.820e+01 6.807e+00 5.612 3.39e-08 ***
## nox -4.294e+01 2.191e+01 -1.960 0.050613 .
## I(nox^2) 3.502e+01 1.688e+01 2.075 0.038523 *
## rm 2.020e+01 2.115e+00 9.554 < 2e-16 ***
## dis -8.421e+00 1.179e+00 -7.141 3.46e-12 ***
## rad 1.652e+00 2.504e-01 6.597 1.11e-10 ***
## tax -1.821e-02 2.967e-03 -6.137 1.76e-09 ***
## ptratio 5.156e+00 7.528e-01 6.850 2.27e-11 ***
## b 2.942e-02 6.230e-03 4.723 3.06e-06 ***
## lstat 1.181e+00 2.097e-01 5.632 3.04e-08 ***
## crim:zn 2.788e-01 1.068e-01 2.612 0.009297 **
## crim:chasYes 2.255e+00 3.265e-01 6.907 1.58e-11 ***
## crim:rad -6.153e-01 1.344e-01 -4.579 5.97e-06 ***
## crim:tax 3.771e-02 8.604e-03 4.383 1.44e-05 ***
## zn:dis 1.711e-02 4.416e-03 3.874 0.000122 ***
## chasYes:nox -3.529e+01 6.009e+00 -5.872 8.05e-09 ***
## chasYes:rm -2.998e+00 7.835e-01 -3.827 0.000147 ***
## nox:rad -1.478e+00 3.341e-01 -4.424 1.20e-05 ***
## rm:dis 9.826e-01 1.614e-01 6.088 2.35e-09 ***
## rm:ptratio -8.672e-01 1.162e-01 -7.466 3.94e-13 ***
## rm:lstat -3.074e-01 3.469e-02 -8.861 < 2e-16 ***
## dis:rad -1.221e-01 2.655e-02 -4.600 5.42e-06 ***
## dis:lstat 1.647e-01 2.032e-02 8.109 4.32e-15 ***
## b:lstat -1.237e-03 2.946e-04 -4.198 3.20e-05 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.396 on 479 degrees of freedom
## Multiple R-squared: 0.8707, Adjusted R-squared: 0.8637
## F-statistic: 124 on 26 and 479 DF, p-value: < 2.2e-16
```

```
higher_order_nox_3 = lm(formula = medv ~ crim + zn + chas + nox + I(nox^2) + I(nox^3) + rm + dis + rad +
summary(higher_order_nox_3)
```

```
##
## Call:
## lm(formula = medv ~ crim + zn + chas + nox + I(nox^2) + I(nox^3) +
##   rm + dis + rad + tax + ptratio + b + lstat + crim:zn + crim:chas +
##   crim:rad + crim:tax + zn:dis + chas:nox + chas:rm + nox:rad +
##   rm:dis + rm:ptratio + rm:lstat + dis:rad + dis:lstat + b:lstat,
##   data = boston_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.0322  -1.7615  -0.0327   1.7776  21.6156
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.034e+02  4.287e+01  -2.412 0.016254 *
## crim        -1.091e+01  2.699e+00  -4.043 6.15e-05 ***
## zn          -1.119e-01  3.251e-02  -3.442 0.000628 ***
```

```
## chasYes      3.787e+01  6.833e+00   5.542 4.95e-08 ***
## nox          8.184e+01  2.010e+02   0.407 0.684004
## I(nox^2)     -1.746e+02  3.360e+02  -0.520 0.603555
## I(nox^3)      1.113e+02  1.782e+02   0.625 0.532494
## rm           2.042e+01  2.145e+00   9.520 < 2e-16 ***
## dis         -8.421e+00  1.180e+00  -7.136 3.58e-12 ***
## rad          1.517e+00  3.311e-01   4.581 5.91e-06 ***
## tax         -1.784e-02  3.030e-03  -5.886 7.43e-09 ***
## ptratio      5.241e+00  7.654e-01   6.848 2.31e-11 ***
## b            2.932e-02  6.236e-03   4.701 3.39e-06 ***
## lstat        1.168e+00  2.109e-01   5.538 5.07e-08 ***
## crim:zn       3.099e-01  1.178e-01   2.630 0.008817 **
## crim:chasYes  2.256e+00  3.267e-01   6.907 1.59e-11 ***
## crim:rad      -6.390e-01  1.397e-01  -4.573 6.12e-06 ***
## crim:tax       3.918e-02  8.921e-03   4.391 1.39e-05 ***
## zn:dis         1.755e-02  4.476e-03   3.922 0.000101 ***
## chasYes:nox   -3.525e+01  6.013e+00  -5.862 8.51e-09 ***
## chasYes:rm    -2.955e+00  7.870e-01  -3.755 0.000195 ***
## nox:rad       -1.250e+00  4.952e-01  -2.524 0.011925 *
## rm:dis         9.811e-01  1.615e-01   6.073 2.56e-09 ***
## rm:ptratio    -8.804e-01  1.181e-01  -7.453 4.31e-13 ***
## rm:lstat      -3.049e-01  3.496e-02  -8.721 < 2e-16 ***
## dis:rad       -1.213e-01  2.661e-02  -4.558 6.58e-06 ***
## dis:lstat      1.638e-01  2.039e-02   8.032 7.53e-15 ***
## b:lstat       -1.229e-03  2.951e-04  -4.164 3.71e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.398 on 478 degrees of freedom
## Multiple R-squared:  0.8708, Adjusted R-squared:  0.8635
## F-statistic: 119.3 on 27 and 478 DF,  p-value: < 2.2e-16
```

Possible higher order relationship exploration for rm

```
higher_order_rm_2 = lm(formula = medv ~ crim + zn + chas + nox + rm + I(rm^2) + dis + rad + tax + ptratio
summary(higher_order_rm_2)
```

```
##
## Call:
## lm(formula = medv ~ crim + zn + chas + nox + rm + I(rm^2) + dis +
##      rad + tax + ptratio + b + lstat + crim:zn + crim:chas + crim:rad +
##      crim:tax + zn:dis + chas:nox + chas:rm + nox:rad + rm:dis +
##      rm:ptratio + rm:lstat + dis:rad + dis:lstat + b:lstat, data = boston_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.8887  -1.7711   0.0397   1.6668  22.8240
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -2.626e+01  2.057e+01  -1.276 0.202417
```

```
## crim      -1.165e+01  2.576e+00 -4.523 7.68e-06 ***
## zn        -7.511e-02  3.069e-02 -2.448 0.014736 *
## chasYes    4.305e+01  6.856e+00  6.279 7.66e-10 ***
## nox       -3.452e-01  5.210e+00 -0.066 0.947196
## rm         3.765e+00  4.479e+00  0.841 0.400954
## I(rm^2)    9.812e-01  2.577e-01  3.808 0.000158 ***
## dis       -7.234e+00  1.175e+00 -6.156 1.58e-09 ***
## rad        1.493e+00  2.485e-01  6.010 3.67e-09 ***
## tax       -1.850e-02  2.917e-03 -6.341 5.27e-10 ***
## ptratio    3.906e+00  7.324e-01  5.333 1.49e-07 ***
## b          2.945e-02  6.165e-03  4.778 2.36e-06 ***
## lstat      7.579e-01  2.420e-01  3.131 0.001848 **
## crim:zn    2.049e-01  1.053e-01  1.947 0.052169 .
## crim:chasYes 2.437e+00  3.277e-01  7.436 4.84e-13 ***
## crim:rad   -6.756e-01  1.313e-01 -5.146 3.90e-07 ***
## crim:tax    4.161e-02  8.430e-03  4.936 1.10e-06 ***
## zn:dis      1.314e-02  4.266e-03  3.079 0.002194 **
## chasYes:nox -3.736e+01  5.976e+00 -6.251 9.02e-10 ***
## chasYes:rm  -3.588e+00  7.832e-01 -4.582 5.89e-06 ***
## nox:rad     -1.321e+00  3.281e-01 -4.025 6.63e-05 ***
## rm:dis      8.552e-01  1.623e-01  5.270 2.07e-07 ***
## rm:ptratio  -6.820e-01  1.140e-01 -5.983 4.28e-09 ***
## rm:lstat    -2.224e-01  4.204e-02 -5.289 1.87e-07 ***
## dis:rad     -1.046e-01  2.675e-02 -3.912 0.000105 ***
## dis:lstat    1.400e-01  2.042e-02  6.856 2.19e-11 ***
## b:lstat     -1.228e-03  2.916e-04 -4.210 3.05e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.361 on 479 degrees of freedom
## Multiple R-squared:  0.8734, Adjusted R-squared:  0.8665
## F-statistic: 127 on 26 and 479 DF, p-value: < 2.2e-16
```

```
higher_order_rm_3 = lm(formula = medv ~ crim + zn + chas + nox + rm + I(rm^2) + I(rm^3) + dis + rad + tax
summary(higher_order_rm_3)
```

```
##
## Call:
## lm(formula = medv ~ crim + zn + chas + nox + rm + I(rm^2) + I(rm^3) +
##   dis + rad + tax + ptratio + b + lstat + crim:zn + crim:chas +
##   crim:rad + crim:tax + zn:dis + chas:nox + chas:rm + nox:rad +
##   rm:dis + rm:ptratio + rm:lstat + dis:rad + dis:lstat + b:lstat,
##   data = boston_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.8903  -1.7694   0.0387   1.6676  22.8214
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -26.730781  34.371242  -0.778 0.437126
## crim         -11.653876   2.586368  -4.506 8.32e-06 ***
## zn           -0.075100   0.030721  -2.445 0.014861 *
## chasYes       43.059324   6.893981   6.246 9.32e-10 ***
```



```
## nox          -0.350820    5.225251   -0.067  0.946499
## rm           4.011523   14.979763    0.268  0.788972
## I(rm^2)       0.940527    2.373196    0.396  0.692051
## I(rm^3)       0.002150    0.124714    0.017  0.986253
## dis          -7.240139    1.225258   -5.909  6.54e-09 ***
## rad           1.493544    0.248828    6.002  3.85e-09 ***
## tax          -0.018502    0.002927   -6.322  5.93e-10 ***
## ptratio       3.905921    0.733216    5.327  1.54e-07 ***
## b             0.029464    0.006201    4.752  2.67e-06 ***
## lstat         0.757310    0.244473    3.098  0.002065 **
## crim:zn       0.205081    0.105739    1.939  0.053029 .
## crim:chasYes  2.436718    0.328097    7.427  5.15e-13 ***
## crim:rad      -0.675768    0.131876   -5.124  4.34e-07 ***
## crim:tax       0.041624    0.008467    4.916  1.22e-06 ***
## zn:dis         0.013137    0.004271    3.076  0.002220 **
## chasYes:nox   -37.362749    5.995800   -6.231  1.01e-09 ***
## chasYes:rm    -3.589523    0.786749   -4.562  6.43e-06 ***
## nox:rad       -1.320427    0.328657   -4.018  6.83e-05 ***
## rm:dis         0.856120    0.170990    5.007  7.79e-07 ***
## rm:ptratio    -0.681968    0.114097   -5.977  4.45e-09 ***
## rm:lstat      -0.222256    0.042471   -5.233  2.50e-07 ***
## dis:rad       -0.104677    0.026909   -3.890  0.000114 ***
## dis:lstat      0.140001    0.020452    6.845  2.35e-11 ***
## b:lstat       -0.001228    0.000294   -4.178  3.50e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.364 on 478 degrees of freedom
## Multiple R-squared:  0.8734, Adjusted R-squared:  0.8662
## F-statistic: 122.1 on 27 and 478 DF,  p-value: < 2.2e-16
```

Possible higher order relationship exploration for dis

```
higher_order_dis_2 = lm(formula = medv ~ crim + zn + chas + nox + rm + dis + I(dis^2) + rad + tax + ptratio, data = boston_data)
summary(higher_order_dis_2)
```

```
##
## Call:
## lm(formula = medv ~ crim + zn + chas + nox + rm + dis + I(dis^2) +
##     rad + tax + ptratio + b + lstat + crim:zn + crim:chas + crim:rad +
##     crim:tax + zn:dis + chas:nox + chas:rm + nox:rad + rm:dis +
##     rm:ptratio + rm:lstat + dis:rad + dis:lstat + b:lstat, data = boston_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.2598  -1.7797  -0.0041   1.7395  20.9960
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -7.418e+01  1.434e+01  -5.174 3.37e-07 ***
## crim         -7.737e+00  2.751e+00  -2.812 0.005120 **
```

```
## zn          3.468e-02  4.633e-02   0.748  0.454537
## chasYes     3.807e+01  6.749e+00   5.641  2.89e-08 ***
## nox        -4.497e+00  5.443e+00  -0.826  0.409136
## rm          1.905e+01  2.016e+00   9.450 < 2e-16 ***
## dis        -1.049e+01  1.341e+00  -7.819  3.42e-14 ***
## I(dis^2)    2.511e-01  7.042e-02   3.566  0.000399 ***
## rad         1.293e+00  2.622e-01   4.932  1.13e-06 ***
## tax        -1.770e-02  2.941e-03  -6.017  3.54e-09 ***
## ptratio     4.609e+00  7.069e-01   6.519  1.79e-10 ***
## b           2.995e-02  6.179e-03   4.848  1.69e-06 ***
## lstat       1.193e+00  2.061e-01   5.790  1.28e-08 ***
## crim:zn     -1.402e-02  1.281e-01  -0.109  0.912879
## crim:chasYes 2.138e+00  3.238e-01   6.603  1.07e-10 ***
## crim:rad    -4.972e-01  1.392e-01  -3.572  0.000390 ***
## crim:tax     2.929e-02  8.988e-03   3.259  0.001198 **
## zn:dis      -4.534e-03  6.911e-03  -0.656  0.512062
## chasYes:nox -3.309e+01  5.911e+00  -5.598  3.66e-08 ***
## chasYes:rm  -3.141e+00  7.747e-01  -4.055  5.86e-05 ***
## nox:rad     -1.078e+00  3.394e-01  -3.175  0.001597 **
## rm:dis       9.376e-01  1.601e-01   5.857  8.78e-09 ***
## rm:ptratio  -8.032e-01  1.101e-01  -7.292  1.27e-12 ***
## rm:lstat    -3.092e-01  3.420e-02  -9.042 < 2e-16 ***
## dis:rad     -8.441e-02  2.854e-02  -2.957  0.003256 **
## dis:lstat    1.663e-01  2.002e-02   8.304  1.04e-15 ***
## b:lstat     -1.287e-03  2.924e-04  -4.401  1.33e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.367 on 479 degrees of freedom
## Multiple R-squared:  0.8729, Adjusted R-squared:  0.866
## F-statistic: 126.5 on 26 and 479 DF,  p-value: < 2.2e-16
```

```
higher_order_dis_3 = lm(formula = medv ~ crim + zn + chas + nox + rm + dis + I(dis^2)+ I(dis^3)+ rad +
summary(higher_order_dis_3)
```

```
##
## Call:
## lm(formula = medv ~ crim + zn + chas + nox + rm + dis + I(dis^2) +
##     I(dis^3) + rad + tax + ptratio + b + lstat + crim:zn + crim:chas +
##     crim:rad + crim:tax + zn:dis + chas:nox + chas:rm + nox:rad +
##     rm:dis + rm:ptratio + rm:lstat + dis:rad + dis:lstat + b:lstat,
##     data = boston_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.362  -1.810   0.018   1.710  20.581
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -7.072e+01  1.452e+01  -4.872 1.51e-06 ***
## crim         -7.018e+00  2.792e+00  -2.514  0.01228 *
## zn            1.712e-02  4.783e-02   0.358  0.72045
## chasYes       3.874e+01  6.757e+00   5.733 1.75e-08 ***
## nox          -6.396e+00  5.591e+00  -1.144  0.25323
```

```
## rm          1.913e+01  2.015e+00   9.496 < 2e-16 ***
## dis        -1.220e+01  1.782e+00  -6.845 2.36e-11 ***
## I(dis^2)     5.761e-01  2.342e-01   2.460 0.01424 *
## I(dis^3)    -1.873e-02  1.287e-02  -1.455 0.14632
## rad         1.177e+00  2.737e-01   4.303 2.05e-05 ***
## tax        -1.808e-02  2.950e-03  -6.130 1.84e-09 ***
## ptratio     4.617e+00  7.061e-01   6.538 1.60e-10 ***
## b          3.014e-02  6.173e-03   4.882 1.44e-06 ***
## lstat       1.195e+00  2.059e-01   5.806 1.17e-08 ***
## crim:zn     -2.776e-02  1.283e-01  -0.216 0.82878
## crim:chasYes 2.084e+00  3.256e-01   6.401 3.70e-10 ***
## crim:rad    -4.640e-01  1.409e-01  -3.293 0.00106 **
## crim:tax     2.701e-02  9.114e-03   2.964 0.00319 **
## zn:dis      -1.743e-03  7.164e-03  -0.243 0.80787
## chasYes:nox -3.327e+01  5.905e+00  -5.634 3.01e-08 ***
## chasYes:rm  -3.209e+00  7.752e-01  -4.140 4.11e-05 ***
## nox:rad     -9.583e-01  3.488e-01  -2.747 0.00624 **
## rm:dis       9.187e-01  1.604e-01   5.727 1.81e-08 ***
## rm:ptratio  -8.030e-01  1.100e-01  -7.300 1.21e-12 ***
## rm:lstat    -3.103e-01  3.417e-02  -9.082 < 2e-16 ***
## dis:rad     -6.671e-02  3.100e-02  -2.152 0.03188 *
## dis:lstat    1.676e-01  2.002e-02   8.371 6.36e-16 ***
## b:lstat     -1.309e-03  2.925e-04  -4.476 9.52e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.363 on 478 degrees of freedom
## Multiple R-squared:  0.8735, Adjusted R-squared:  0.8663
## F-statistic: 122.2 on 27 and 478 DF,  p-value: < 2.2e-16
```

Possible higher order relationship exploration for rad

```
higher_order_rad_2 = lm(formula = medv ~ crim + zn + chas + nox + rm + dis + rad + I(rad^2) + tax + ptratio,
summary(higher_order_rad_2)
```

```
##
## Call:
## lm(formula = medv ~ crim + zn + chas + nox + rm + dis + rad +
##      I(rad^2) + tax + ptratio + b + lstat + crim:zn + crim:chas +
##      crim:rad + crim:tax + zn:dis + chas:nox + chas:rm + nox:rad +
##      rm:dis + rm:ptratio + rm:lstat + dis:rad + dis:lstat + b:lstat,
##      data = boston_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.5754  -1.8727  -0.0668   1.7989  21.4461
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -8.456e+01  1.427e+01  -5.924 6.00e-09 ***
## crim        -1.167e+01  2.625e+00  -4.447 1.09e-05 ***
```

```
## zn          -8.892e-02  3.083e-02  -2.884  0.004100 **
## chasYes     3.797e+01  6.823e+00   5.566  4.35e-08 ***
## nox         6.351e-01  5.272e+00   0.120  0.904164
## rm          1.901e+01  2.038e+00   9.330  < 2e-16 ***
## dis        -8.111e+00  1.169e+00  -6.936  1.31e-11 ***
## rad         1.879e+00  3.071e-01   6.120  1.95e-09 ***
## I(rad^2)    -1.101e-02  7.154e-03  -1.539  0.124451
## tax        -1.891e-02  2.951e-03  -6.409  3.52e-10 ***
## ptratio     4.698e+00  7.149e-01   6.571  1.31e-10 ***
## b           2.929e-02  6.242e-03   4.693  3.52e-06 ***
## lstat       1.228e+00  2.081e-01   5.901  6.85e-09 ***
## crim:zn     2.400e-01  1.061e-01   2.262  0.024135 *
## crim:chasYes 2.201e+00  3.267e-01   6.737  4.66e-11 ***
## crim:rad    -6.777e-01  1.334e-01  -5.082  5.38e-07 ***
## crim:tax     4.171e-02  8.575e-03   4.863  1.57e-06 ***
## zn:dis       1.506e-02  4.295e-03   3.506  0.000498 ***
## chasYes:nox -3.354e+01  5.972e+00  -5.616  3.31e-08 ***
## chasYes:rm  -3.104e+00  7.830e-01  -3.964  8.48e-05 ***
## nox:rad     -1.292e+00  3.373e-01  -3.831  0.000145 ***
## rm:dis       9.786e-01  1.617e-01   6.051  2.91e-09 ***
## rm:ptratio  -8.001e-01  1.113e-01  -7.188  2.54e-12 ***
## rm:lstat    -3.140e-01  3.454e-02  -9.091  < 2e-16 ***
## dis:rad     -1.393e-01  2.841e-02  -4.902  1.30e-06 ***
## dis:lstat    1.613e-01  2.021e-02   7.981  1.08e-14 ***
## b:lstat     -1.248e-03  2.953e-04  -4.225  2.86e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.403 on 479 degrees of freedom
## Multiple R-squared:  0.8702, Adjusted R-squared:  0.8631
## F-statistic: 123.5 on 26 and 479 DF,  p-value: < 2.2e-16
```

Possible higher order relationship exploration for tax

```
higher_order_tax_2 = lm(formula = medv ~ crim + zn + chas + nox + rm + dis + rad + tax + I(tax^2) + ptratio, data = boston_data)
summary(higher_order_tax_2)
```

```
##
## Call:
## lm(formula = medv ~ crim + zn + chas + nox + rm + dis + rad +
##     tax + I(tax^2) + ptratio + b + lstat + crim:zn + crim:chas +
##     crim:rad + crim:tax + zn:dis + chas:nox + chas:rm + nox:rad +
##     rm:dis + rm:ptratio + rm:lstat + dis:rad + dis:lstat + b:lstat,
##     data = boston_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.2319  -1.7603  -0.0322   1.6989  21.5852
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept) -7.953e+01 1.444e+01 -5.508 5.93e-08 ***
## crim -1.057e+01 2.630e+00 -4.020 6.76e-05 ***
## zn -9.053e-02 3.082e-02 -2.937 0.003471 **
## chasYes 3.843e+01 6.820e+00 5.635 2.99e-08 ***
## nox 1.270e+00 5.257e+00 0.242 0.809228
## rm 1.884e+01 2.039e+00 9.239 < 2e-16 ***
## dis -7.962e+00 1.170e+00 -6.802 3.08e-11 ***
## rad 1.577e+00 2.503e-01 6.299 6.79e-10 ***
## tax -3.368e-02 9.439e-03 -3.569 0.000395 ***
## I(tax^2) 1.907e-05 1.162e-05 1.642 0.101336
## ptratio 4.572e+00 7.159e-01 6.387 4.01e-10 ***
## b 2.944e-02 6.241e-03 4.717 3.14e-06 ***
## lstat 1.243e+00 2.077e-01 5.984 4.26e-09 ***
## crim:zn 2.464e-01 1.059e-01 2.325 0.020466 *
## crim:chasYes 2.198e+00 3.266e-01 6.731 4.83e-11 ***
## crim:rad -6.538e-01 1.329e-01 -4.919 1.20e-06 ***
## crim:tax 3.919e-02 8.554e-03 4.582 5.88e-06 ***
## zn:dis 1.497e-02 4.293e-03 3.487 0.000534 ***
## chasYes:nox -3.343e+01 5.972e+00 -5.597 3.67e-08 ***
## chasYes:rm -3.172e+00 7.835e-01 -4.048 6.02e-05 ***
## nox:rad -1.427e+00 3.326e-01 -4.290 2.16e-05 ***
## rm:dis 9.538e-01 1.617e-01 5.897 7.00e-09 ***
## rm:ptratio -7.860e-01 1.115e-01 -7.052 6.18e-12 ***
## rm:lstat -3.159e-01 3.450e-02 -9.157 < 2e-16 ***
## dis:rad -1.219e-01 2.661e-02 -4.580 5.93e-06 ***
## dis:lstat 1.583e-01 2.010e-02 7.878 2.25e-14 ***
## b:lstat -1.243e-03 2.951e-04 -4.212 3.03e-05 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.402 on 479 degrees of freedom
## Multiple R-squared: 0.8702, Adjusted R-squared: 0.8632
## F-statistic: 123.6 on 26 and 479 DF, p-value: < 2.2e-16
```

Possible higher order relationship exploration for lstat

```
higher_order_lstat_2 = lm(formula = medv ~ crim + zn + chas + nox + rm + dis + rad + tax + ptratio + b
summary(higher_order_lstat_2)
```

```
##
## Call:
## lm(formula = medv ~ crim + zn + chas + nox + rm + dis + rad +
##     tax + ptratio + b + lstat + I(lstat^2) + crim:zn + crim:chas +
##     crim:rad + crim:tax + zn:dis + chas:nox + chas:rm + nox:rad +
##     rm:dis + rm:ptratio + rm:lstat + dis:rad + dis:lstat + b:lstat,
##     data = boston_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.9778  -1.9930  -0.0558   1.6721  21.7134
##
```

```
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept) -7.078e+01 1.441e+01 -4.911 1.25e-06 ***
## crim        -1.071e+01 2.572e+00 -4.165 3.69e-05 ***
## zn          -9.722e-02 3.047e-02 -3.191 0.001511 **
## chasYes      3.748e+01 6.731e+00  5.568 4.29e-08 ***
## nox          3.399e+00 5.217e+00  0.652 0.514963
## rm           1.792e+01 2.028e+00  8.837 < 2e-16 ***
## dis         -8.844e+00 1.170e+00 -7.560 2.07e-13 ***
## rad          1.566e+00 2.466e-01  6.351 4.98e-10 ***
## tax         -1.785e-02 2.924e-03 -6.105 2.13e-09 ***
## ptratio      4.905e+00 7.075e-01  6.934 1.33e-11 ***
## b            2.090e-02 6.504e-03  3.214 0.001399 **
## lstat       -3.365e-01 4.487e-01 -0.750 0.453632
## I(lstat^2)    1.789e-02 4.509e-03  3.967 8.39e-05 ***
## crim:zn       2.410e-01 1.045e-01  2.306 0.021551 *
## crim:chasYes  2.126e+00 3.229e-01  6.582 1.22e-10 ***
## crim:rad     -6.153e-01 1.316e-01 -4.677 3.79e-06 ***
## crim:tax      3.803e-02 8.433e-03  4.509 8.18e-06 ***
## zn:dis        1.610e-02 4.247e-03  3.792 0.000169 ***
## chasYes:nox   -3.275e+01 5.895e+00 -5.555 4.62e-08 ***
## chasYes:rm    -3.059e+00 7.725e-01 -3.959 8.65e-05 ***
## nox:rad       -1.435e+00 3.275e-01 -4.381 1.45e-05 ***
## rm:dis        1.039e+00 1.604e-01  6.476 2.33e-10 ***
## rm:ptratio    -8.334e-01 1.102e-01 -7.565 2.01e-13 ***
## rm:lstat     -1.868e-01 4.713e-02 -3.963 8.53e-05 ***
## dis:rad       -1.085e-01 2.652e-02 -4.092 5.02e-05 ***
## dis:lstat     1.818e-01 2.071e-02  8.778 < 2e-16 ***
## b:lstat      -7.696e-04 3.142e-04 -2.450 0.014662 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.357 on 479 degrees of freedom
## Multiple R-squared:  0.8737, Adjusted R-squared:  0.8668
## F-statistic: 127.4 on 26 and 479 DF, p-value: < 2.2e-16
```

```
higher_order_lstat_3 = lm(formula = medv ~ crim + zn + chas + nox + rm + dis + rad + tax + ptratio + b +
summary(higher_order_lstat_3)
```

```
##
## Call:
## lm(formula = medv ~ crim + zn + chas + nox + rm + dis + rad +
##     tax + ptratio + b + lstat + I(lstat^2) + I(lstat^3) + crim:zn +
##     crim:chas + crim:rad + crim:tax + zn:dis + chas:nox + chas:rm +
##     nox:rad + rm:dis + rm:ptratio + rm:lstat + dis:rad + dis:lstat +
##     b:lstat, data = boston_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.8970  -1.9732   0.0132   1.7517  22.1711
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept) -6.513e+01 1.445e+01 -4.509 8.22e-06 ***
```

```
## crim      -1.118e+01  2.559e+00 -4.369 1.53e-05 ***
## zn        -1.072e-01  3.045e-02 -3.521 0.000471 ***
## chasYes    3.624e+01  6.696e+00  5.413 9.84e-08 ***
## nox        2.399e+00  5.191e+00  0.462 0.644216
## rm         1.748e+01  2.020e+00  8.656 < 2e-16 ***
## dis       -8.706e+00  1.162e+00 -7.491 3.33e-13 ***
## rad        1.528e+00  2.452e-01  6.232 1.01e-09 ***
## tax       -1.804e-02  2.904e-03 -6.213 1.13e-09 ***
## ptratio    4.757e+00  7.043e-01  6.754 4.19e-11 ***
## b          2.262e-02  6.485e-03  3.488 0.000531 ***
## lstat     -8.109e-01  4.759e-01 -1.704 0.089032 .
## I(lstat^2)  5.946e-02  1.534e-02  3.875 0.000121 ***
## I(lstat^3) -7.983e-04  2.819e-04 -2.832 0.004815 **
## crim:zn     2.827e-01  1.048e-01  2.697 0.007240 **
## crim:chasYes 2.089e+00  3.208e-01  6.511 1.89e-10 ***
## crim:rad    -6.269e-01  1.307e-01 -4.798 2.15e-06 ***
## crim:tax     3.915e-02  8.381e-03  4.671 3.90e-06 ***
## zn:dis       1.693e-02  4.226e-03  4.007 7.14e-05 ***
## chasYes:nox -3.136e+01  5.873e+00 -5.339 1.45e-07 ***
## chasYes:rm  -2.999e+00  7.672e-01 -3.910 0.000106 ***
## nox:rad     -1.391e+00  3.255e-01 -4.275 2.31e-05 ***
## rm:dis       1.007e+00  1.596e-01  6.311 6.34e-10 ***
## rm:ptratio  -8.109e-01  1.097e-01 -7.394 6.42e-13 ***
## rm:lstat    -2.058e-01  4.726e-02 -4.354 1.64e-05 ***
## dis:rad     -1.038e-01  2.638e-02 -3.937 9.49e-05 ***
## dis:lstat    1.836e-01  2.057e-02  8.924 < 2e-16 ***
## b:lstat     -8.654e-04  3.137e-04 -2.758 0.006032 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.332 on 478 degrees of freedom
## Multiple R-squared:  0.8758, Adjusted R-squared:  0.8687
## F-statistic: 124.8 on 27 and 478 DF, p-value: < 2.2e-16
```

```
higher_order_lstat_4 = lm(formula = medv ~ crim + zn + chas + nox + rm + dis + rad + tax + ptratio + b +
summary(higher_order_lstat_4)
```

```
##
## Call:
## lm(formula = medv ~ crim + zn + chas + nox + rm + dis + rad +
##     tax + ptratio + b + lstat + I(lstat^2) + I(lstat^3) + I(lstat^4) +
##     crim:zn + crim:chas + crim:rad + crim:tax + zn:dis + chas:nox +
##     chas:rm + nox:rad + rm:dis + rm:ptratio + rm:lstat + dis:rad +
##     dis:lstat + b:lstat, data = boston_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.5305  -1.9086   0.0193   1.6720  22.6857
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -5.497e+01  1.478e+01  -3.719 0.000224 ***
## crim         -1.112e+01  2.540e+00  -4.379 1.46e-05 ***
## zn           -1.131e-01  3.030e-02  -3.732 0.000213 ***
```

```
## chasYes      3.537e+01  6.654e+00   5.315 1.64e-07 ***
## nox          2.030e+00  5.155e+00   0.394 0.693887
## rm           1.665e+01  2.027e+00   8.214 2.03e-15 ***
## dis         -8.558e+00  1.155e+00  -7.410 5.79e-13 ***
## rad          1.528e+00  2.434e-01   6.278 7.70e-10 ***
## tax         -1.843e-02  2.886e-03  -6.386 4.05e-10 ***
## ptratio      4.579e+00  7.019e-01   6.524 1.75e-10 ***
## b            2.277e-02  6.438e-03   3.537 0.000445 ***
## lstat       -2.405e+00  7.337e-01  -3.278 0.001122 **
## I(lstat^2)    2.059e-01  5.378e-02   3.829 0.000146 ***
## I(lstat^3)   -6.678e-03  2.089e-03  -3.196 0.001486 **
## I(lstat^4)    7.931e-05  2.793e-05   2.840 0.004711 **
## crim:zn       3.024e-01  1.043e-01   2.900 0.003904 **
## crim:chasYes  2.099e+00  3.185e-01   6.590 1.17e-10 ***
## crim:rad     -6.213e-01  1.297e-01  -4.789 2.24e-06 ***
## crim:tax      3.886e-02  8.320e-03   4.671 3.91e-06 ***
## zn:dis        1.725e-02  4.197e-03   4.110 4.65e-05 ***
## chasYes:nox  -3.081e+01  5.833e+00  -5.283 1.94e-07 ***
## chasYes:rm   -2.932e+00  7.619e-01  -3.849 0.000135 ***
## nox:rad      -1.393e+00  3.231e-01  -4.310 1.98e-05 ***
## rm:dis        9.916e-01  1.585e-01   6.255 8.85e-10 ***
## rm:ptratio   -7.884e-01  1.091e-01  -7.224 2.02e-12 ***
## rm:lstat     -1.721e-01  4.839e-02  -3.556 0.000414 ***
## dis:rad      -1.037e-01  2.618e-02  -3.962 8.56e-05 ***
## dis:lstat     1.792e-01  2.048e-02   8.753 < 2e-16 ***
## b:lstat      -8.467e-04  3.115e-04  -2.718 0.006803 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.308 on 477 degrees of freedom
## Multiple R-squared:  0.8778, Adjusted R-squared:  0.8706
## F-statistic: 122.4 on 28 and 477 DF,  p-value: < 2.2e-16
```

```
higher_order_lstat_5 = lm(formula = medv ~ crim + zn + chas + nox + rm + dis + rad + tax + ptratio + b +
summary(higher_order_lstat_5)
```

```
##
## Call:
## lm(formula = medv ~ crim + zn + chas + nox + rm + dis + rad +
##     tax + ptratio + b + lstat + I(lstat^2) + I(lstat^3) + I(lstat^4) +
##     I(lstat^5) + crim:zn + crim:chas + crim:rad + crim:tax +
##     zn:dis + chas:nox + chas:rm + nox:rad + rm:dis + rm:ptratio +
##     rm:lstat + dis:rad + dis:lstat + b:lstat, data = boston_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.176  -1.863   -0.012    1.559   22.618
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -4.979e+01  1.480e+01  -3.365 0.000827 ***
## crim         -1.059e+01  2.530e+00  -4.187 3.37e-05 ***
## zn           -1.146e-01  3.009e-02  -3.808 0.000158 ***
## chasYes       3.663e+01  6.624e+00   5.530 5.30e-08 ***
```



```
## nox          2.011e+00  5.119e+00   0.393 0.694661
## rm           1.659e+01  2.013e+00   8.243 1.64e-15 ***
## dis         -8.204e+00  1.154e+00  -7.109 4.31e-12 ***
## rad          1.554e+00  2.419e-01   6.425 3.21e-10 ***
## tax         -1.848e-02  2.866e-03  -6.448 2.78e-10 ***
## ptratio      4.518e+00  6.974e-01   6.479 2.31e-10 ***
## b            2.261e-02  6.394e-03   3.537 0.000444 ***
## lstat       -4.913e+00  1.161e+00  -4.231 2.79e-05 ***
## I(lstat^2)    6.011e-01  1.521e-01   3.951 8.95e-05 ***
## I(lstat^3)   -3.303e-02  9.724e-03  -3.397 0.000738 ***
## I(lstat^4)    8.527e-04  2.801e-04   3.044 0.002466 **
## I(lstat^5)   -8.207e-06  2.958e-06  -2.774 0.005750 **
## crim:zn       3.048e-01  1.036e-01   2.943 0.003406 **
## crim:chasYes  2.101e+00  3.163e-01   6.642 8.46e-11 ***
## crim:rad      -5.889e-01  1.294e-01  -4.552 6.74e-06 ***
## crim:tax       3.690e-02  8.293e-03   4.449 1.07e-05 ***
## zn:dis         1.718e-02  4.168e-03   4.122 4.43e-05 ***
## chasYes:nox   -3.121e+01  5.794e+00  -5.386 1.13e-07 ***
## chasYes:rm    -3.097e+00  7.590e-01  -4.081 5.26e-05 ***
## nox:rad       -1.410e+00  3.209e-01  -4.392 1.38e-05 ***
## rm:dis         9.620e-01  1.578e-01   6.096 2.25e-09 ***
## rm:ptratio    -7.797e-01  1.084e-01  -7.190 2.52e-12 ***
## rm:lstat      -1.674e-01  4.809e-02  -3.481 0.000546 ***
## dis:rad       -1.102e-01  2.610e-02  -4.220 2.92e-05 ***
## dis:lstat      1.674e-01  2.078e-02   8.057 6.33e-15 ***
## b:lstat       -8.449e-04  3.093e-04  -2.731 0.006542 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.285 on 476 degrees of freedom
## Multiple R-squared:  0.8798, Adjusted R-squared:  0.8724
## F-statistic: 120.1 on 29 and 476 DF,  p-value: < 2.2e-16
```

```
higher_order_lstat_6 = lm(formula = medv ~ crim + zn + chas + nox + rm + dis + rad + tax + ptratio + b +
summary(higher_order_lstat_6)
```

```
##
## Call:
## lm(formula = medv ~ crim + zn + chas + nox + rm + dis + rad +
##     tax + ptratio + b + lstat + I(lstat^2) + I(lstat^3) + I(lstat^4) +
##     I(lstat^5) + I(lstat^6) + crim:zn + crim:chas + crim:rad +
##     crim:tax + zn:dis + chas:nox + chas:rm + nox:rad + rm:dis +
##     rm:ptratio + rm:lstat + dis:rad + dis:lstat + b:lstat, data = boston_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.2222  -1.8315   0.0012   1.6134  22.5069
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -4.676e+01  1.505e+01  -3.108 0.002000 **
## crim         -1.070e+01  2.531e+00  -4.227 2.85e-05 ***
## zn           -1.158e-01  3.011e-02  -3.846 0.000136 ***
## chasYes       3.686e+01  6.626e+00   5.563 4.43e-08 ***
```

```
## nox          2.108e+00  5.118e+00   0.412 0.680672
## rm           1.658e+01  2.012e+00   8.240 1.69e-15 ***
## dis         -8.247e+00  1.154e+00  -7.144 3.43e-12 ***
## rad          1.547e+00  2.420e-01   6.394 3.88e-10 ***
## tax         -1.847e-02  2.865e-03  -6.446 2.83e-10 ***
## ptratio      4.523e+00  6.973e-01   6.487 2.21e-10 ***
## b            2.245e-02  6.394e-03   3.511 0.000489 ***
## lstat       -6.756e+00  2.038e+00  -3.316 0.000984 ***
## I(lstat^2)    9.793e-01  3.758e-01   2.606 0.009454 **
## I(lstat^3)   -6.888e-02  3.399e-02  -2.026 0.043282 *
## I(lstat^4)    2.547e-03  1.564e-03   1.628 0.104231
## I(lstat^5)   -4.691e-05  3.529e-05  -1.329 0.184415
## I(lstat^6)    3.398e-07  3.088e-07   1.101 0.271668
## crim:zn       3.085e-01  1.036e-01   2.978 0.003051 **
## crim:chasYes  2.100e+00  3.162e-01   6.640 8.57e-11 ***
## crim:rad     -5.898e-01  1.293e-01  -4.560 6.51e-06 ***
## crim:tax      3.708e-02  8.293e-03   4.472 9.70e-06 ***
## zn:dis        1.738e-02  4.170e-03   4.166 3.68e-05 ***
## chasYes:nox  -3.130e+01  5.794e+00  -5.402 1.04e-07 ***
## chasYes:rm   -3.129e+00  7.593e-01  -4.121 4.45e-05 ***
## nox:rad      -1.402e+00  3.209e-01  -4.367 1.54e-05 ***
## rm:dis        9.689e-01  1.579e-01   6.137 1.78e-09 ***
## rm:ptratio   -7.793e-01  1.084e-01  -7.188 2.57e-12 ***
## rm:lstat     -1.659e-01  4.810e-02  -3.450 0.000611 ***
## dis:rad      -1.099e-01  2.610e-02  -4.213 3.02e-05 ***
## dis:lstat     1.676e-01  2.077e-02   8.068 5.89e-15 ***
## b:lstat      -8.411e-04  3.093e-04  -2.719 0.006781 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.284 on 475 degrees of freedom
## Multiple R-squared:  0.8801, Adjusted R-squared:  0.8725
## F-statistic: 116.2 on 30 and 475 DF,  p-value: < 2.2e-16
```

Based on the exploration above, the significant higher order relationships were then added to the reduced interactive model

```
higher_order_interaction_model = lm(formula = medv ~ crim + I(crim^2) + zn + chas + nox + I(nox^2) + rm + I(rm^2) + dis + I(dis^2) + rad + tax + ptratio + b + lstat + I(lstat^2) + I(lstat^3) + I(lstat^4) + I(lstat^5) + crim:zn + crim:chas + crim:rad + crim:tax + zn:dis + chas:nox + chas:rm + nox:rad + rm:dis + rm:ptratio + rm:lstat + dis:rad + dis:lstat, data = boston_data)
```

```
##
## Call:
## lm(formula = medv ~ crim + I(crim^2) + zn + chas + nox + I(nox^2) +
##     rm + I(rm^2) + dis + I(dis^2) + rad + tax + ptratio + b +
##     lstat + I(lstat^2) + I(lstat^3) + I(lstat^4) + I(lstat^5) +
##     crim:zn + crim:chas + crim:rad + crim:tax + zn:dis + chas:nox +
##     chas:rm + nox:rad + rm:dis + rm:ptratio + rm:lstat + dis:rad +
##     dis:lstat, data = boston_data)
##
## Residuals:
```

	Min	1Q	Median	3Q	Max
##	-11.4678	-1.6694	-0.0783	1.4242	22.6809

```
##
```

```
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  9.950e+00  2.065e+01   0.482 0.630071
## crim        -7.693e+00  2.626e+00  -2.930 0.003554 **
## I(crim^2)     3.267e-03  9.529e-04   3.429 0.000659 ***
## zn          -2.735e-02  4.488e-02  -0.609 0.542523
## chasYes       4.081e+01  6.472e+00   6.305 6.63e-10 ***
## nox          -5.100e+01  2.077e+01  -2.455 0.014438 *
## I(nox^2)      3.735e+01  1.597e+01   2.339 0.019748 *
## rm           5.116e+00  4.471e+00   1.144 0.253089
## I(rm^2)       8.483e-01  2.501e-01   3.391 0.000754 ***
## dis          -9.521e+00  1.326e+00  -7.178 2.77e-12 ***
## I(dis^2)      1.950e-01  6.684e-02   2.918 0.003688 **
## rad           1.465e+00  2.533e-01   5.782 1.34e-08 ***
## tax          -1.684e-02  2.796e-03  -6.023 3.44e-09 ***
## ptratio       4.506e+00  7.377e-01   6.109 2.10e-09 ***
## b             2.187e-02  6.137e-03   3.563 0.000404 ***
## lstat        -5.617e+00  1.150e+00  -4.885 1.42e-06 ***
## I(lstat^2)    6.351e-01  1.467e-01   4.330 1.83e-05 ***
## I(lstat^3)   -3.544e-02  9.360e-03  -3.787 0.000172 ***
## I(lstat^4)    9.242e-04  2.693e-04   3.432 0.000652 ***
## I(lstat^5)   -8.935e-06  2.841e-06  -3.145 0.001767 **
## crim:zn       1.002e-01  1.224e-01   0.819 0.413307
## crim:chasYes  2.214e+00  3.103e-01   7.136 3.65e-12 ***
## crim:rad      -4.371e-01  1.336e-01  -3.271 0.001149 **
## crim:tax       2.671e-02  8.604e-03   3.104 0.002022 **
## zn:dis         2.507e-03  6.585e-03   0.381 0.703551
## chasYes:nox   -3.538e+01  5.671e+00  -6.238 9.88e-10 ***
## chasYes:rm    -3.438e+00  7.404e-01  -4.643 4.45e-06 ***
## nox:rad       -1.261e+00  3.207e-01  -3.933 9.67e-05 ***
## rm:dis         8.557e-01  1.539e-01   5.560 4.53e-08 ***
## rm:ptratio    -7.816e-01  1.137e-01  -6.874 1.98e-11 ***
## rm:lstat      -7.299e-02  5.469e-02  -1.335 0.182638
## dis:rad       -9.128e-02  2.872e-02  -3.178 0.001581 **
## dis:lstat     1.575e-01  2.093e-02   7.525 2.68e-13 ***
## b:lstat       -8.328e-04  2.975e-04  -2.800 0.005326 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.148 on 472 degrees of freedom
## Multiple R-squared:  0.8905, Adjusted R-squared:  0.8828
## F-statistic: 116.3 on 33 and 472 DF,  p-value: < 2.2e-16
```

The new insignificant variables were then dropped to get a reduced higher order interaction model.

```
higher_order_interaction_model_2 = lm(formula = medv ~ crim + I(crim^2) + chas + nox + I(nox^2) + rm + I(
summary(higher_order_interaction_model_2)
```

```
##
## Call:
## lm(formula = medv ~ crim + I(crim^2) + chas + nox + I(nox^2) +
##     rm + I(rm^2) + dis + I(dis^2) + rad + tax + ptratio + b +
##     lstat + I(lstat^2) + I(lstat^3) + I(lstat^4) + I(lstat^5) +
```

```
##      crim:chas + crim:rad + crim:tax + chas:nox + chas:rm + nox:rad +
##      rm:dis + rm:ptratio + dis:rad + dis:lstat + b:lstat, data = boston_data)
##
## Residuals:
##      Min        1Q      Median        3Q        Max
## -11.6250  -1.7162  -0.1331   1.4736  22.9873
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.112e+01  1.815e+01   1.164 0.245085
## crim        -6.809e+00  2.307e+00  -2.951 0.003322 **
## I(crim^2)     3.002e-03  9.349e-04   3.211 0.001411 **
## chasYes       4.090e+01  6.456e+00   6.335 5.52e-10 ***
## nox          -4.569e+01  1.937e+01  -2.359 0.018737 *
## I(nox^2)      3.401e+01  1.502e+01   2.265 0.023987 *
## rm           2.071e+00  3.855e+00   0.537 0.591444
## I(rm^2)       1.044e+00  2.129e-01   4.903 1.30e-06 ***
## dis          -9.628e+00  1.281e+00  -7.518 2.78e-13 ***
## I(dis^2)      2.099e-01  4.023e-02   5.217 2.71e-07 ***
## rad           1.436e+00  2.495e-01   5.756 1.55e-08 ***
## tax          -1.669e-02  2.735e-03  -6.101 2.19e-09 ***
## ptratio       4.554e+00  7.094e-01   6.420 3.31e-10 ***
## b            2.067e-02  6.078e-03   3.401 0.000728 ***
## lstat        -6.212e+00  9.861e-01  -6.300 6.79e-10 ***
## I(lstat^2)    6.456e-01  1.438e-01   4.489 9.00e-06 ***
## I(lstat^3)   -3.590e-02  9.249e-03  -3.882 0.000118 ***
## I(lstat^4)    9.344e-04  2.672e-04   3.497 0.000515 ***
## I(lstat^5)   -8.984e-06  2.827e-06  -3.178 0.001580 **
## crim:chasYes  2.277e+00  3.062e-01   7.437 4.85e-13 ***
## crim:rad     -3.893e-01  1.162e-01  -3.350 0.000872 ***
## crim:tax      2.370e-02  7.481e-03   3.168 0.001634 **
## chasYes:nox  -3.586e+01  5.643e+00  -6.355 4.88e-10 ***
## chasYes:rm   -3.418e+00  7.377e-01  -4.633 4.66e-06 ***
## nox:rad      -1.305e+00  3.164e-01  -4.125 4.38e-05 ***
## rm:dis        8.439e-01  1.468e-01   5.748 1.61e-08 ***
## rm:ptratio   -7.927e-01  1.070e-01  -7.407 5.91e-13 ***
## dis:rad      -7.699e-02  2.707e-02  -2.844 0.004642 **
## dis:lstat     1.559e-01  2.065e-02   7.550 2.24e-13 ***
## b:lstat      -7.380e-04  2.898e-04  -2.546 0.011195 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.145 on 476 degrees of freedom
## Multiple R-squared:  0.8898, Adjusted R-squared:  0.8831
## F-statistic: 132.6 on 29 and 476 DF,  p-value: < 2.2e-16
```

The higher order interaction model with significant variables is:

$$\widehat{medv} = 36.341145 - 0.108413crim_i + 0.045845zn_i + 2.718716chas_i - 17.376023nox_i + 3.801579rm_i - 1.492711dis_i + 0.299608rad_i - 0.011778tax_i - 0.946525ptratio_i + 0.009291b_i - 0.522553lstat_i$$

where $chas_i$ is 1 if the tract bounds Charles River and 0 if otherwise

E. Testing Multiple Regression Assumptions

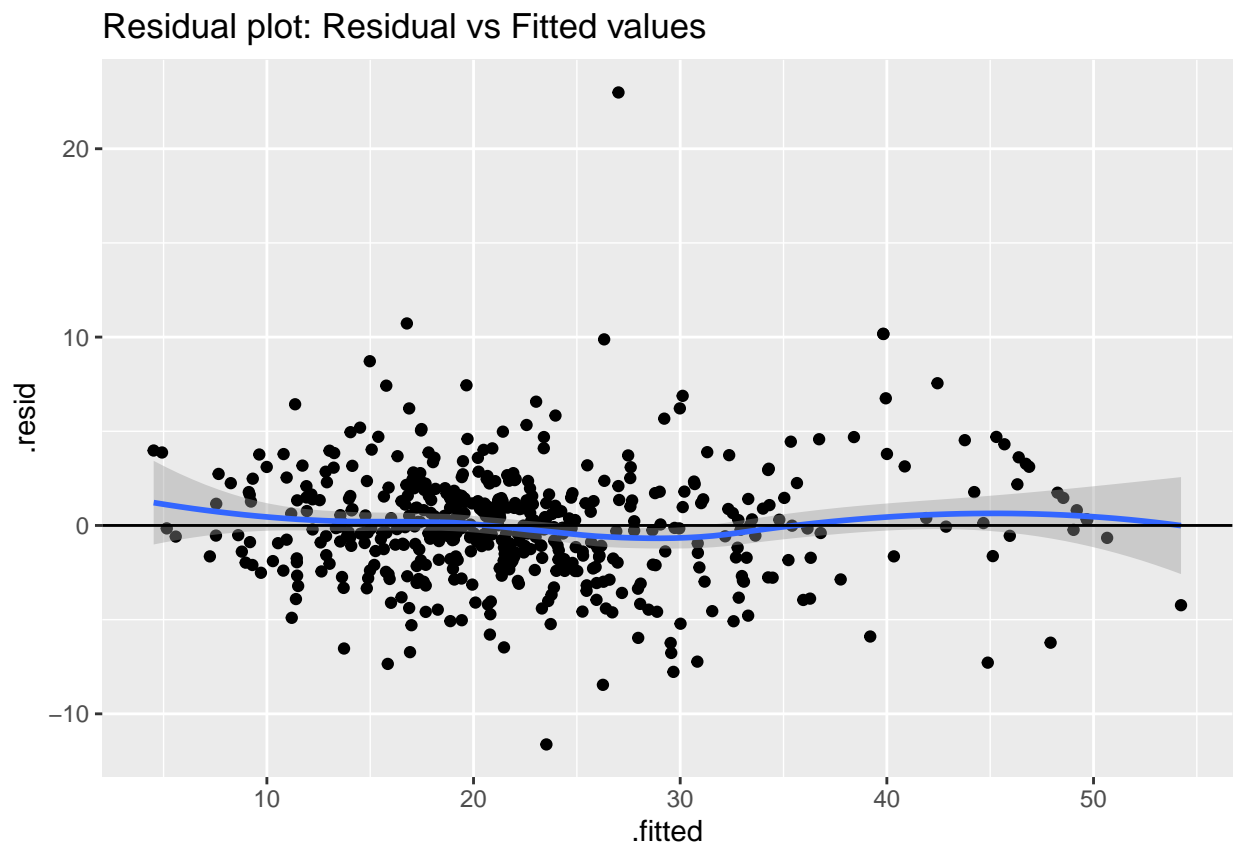
The following multiple regression assumptions were tested to check if the model is trustworthy

Linearity Assumption

A scatter plot of the distribution of residuals (errors) vs fitted values (predicted values) was plotted below

```
ggplot(higher_order_interaction_model_2, aes(x=.fitted, y=.resid)) +  
  geom_point() + geom_smooth() +  
  geom_hline(yintercept = 0) +  
  ggtitle("Residual plot: Residual vs Fitted values")
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```



There appears to be no pattern of the residuals at all, indicating that the model passes the linearity assumption that there is a straight-line (linear) relationship between the predictors and the response.

Independence Assumption

In the Boston housing dataset, the subjects were not related to time, space, or group, so we can be pretty sure that their measurements are independent.

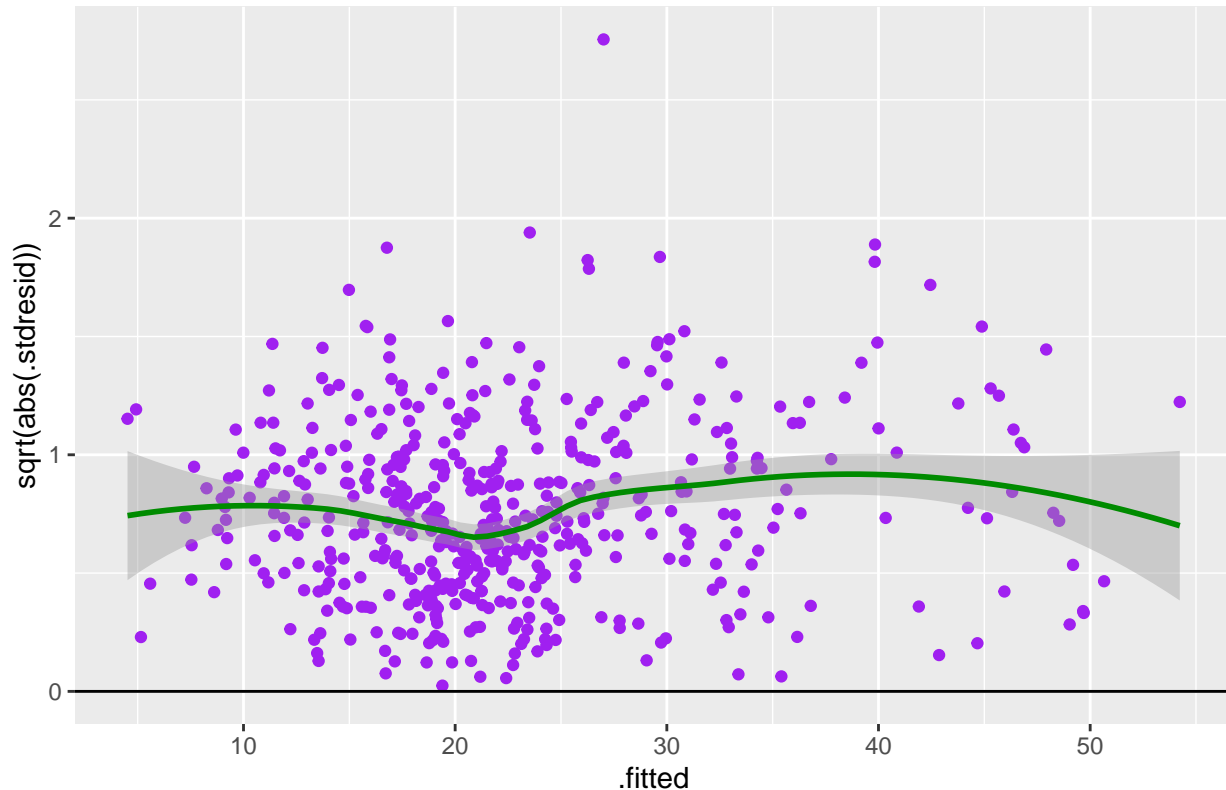
Equal Variance Assumption

The residuals plot in the linearity assumption section indicates a smooth fit to the residuals, which is good. In addition to the residuals plot, a scale-location plot between fitted values and standardized residuals was also plotted to show if the residuals are spread equally along the ranges of predictors

```
ggplot(higher_order_interaction_model_2, aes(x=.fitted, y=sqrt(abs(.stdresid)))) +
  geom_point(colour = "purple") +
  geom_hline(yintercept = 0) +
  geom_smooth( colour = "green4")+
  ggtitle("Scale-Location plot : Standardized Residual vs Fitted values")
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```

Scale–Location plot : Standardized Residual vs Fitted values



Based on the plot, we can see that the scale-location plot is quite horizontal, and there is not any funneling in the residual plot, indicating equal variance.

The Breusch-Pagan test was then run as a more formal way to assess if we have homo/heteroscedasticity using the following hypotheses:

H_0 : heteroscedasticity is not present (homoscedasticity)

H_a : heteroscedasticity is present

```
bptest(higher_order_interaction_model_2)
```

```
##
## studentized Breusch-Pagan test
##
## data: higher_order_interaction_model_2
## BP = 70.605, df = 29, p-value = 2.505e-05
```

The p-value of the Breusch-Pagan test is less than 0.05 ($2.505e-05$), so we fail to reject the null hypothesis, indicating we do have heteroscedasticity.

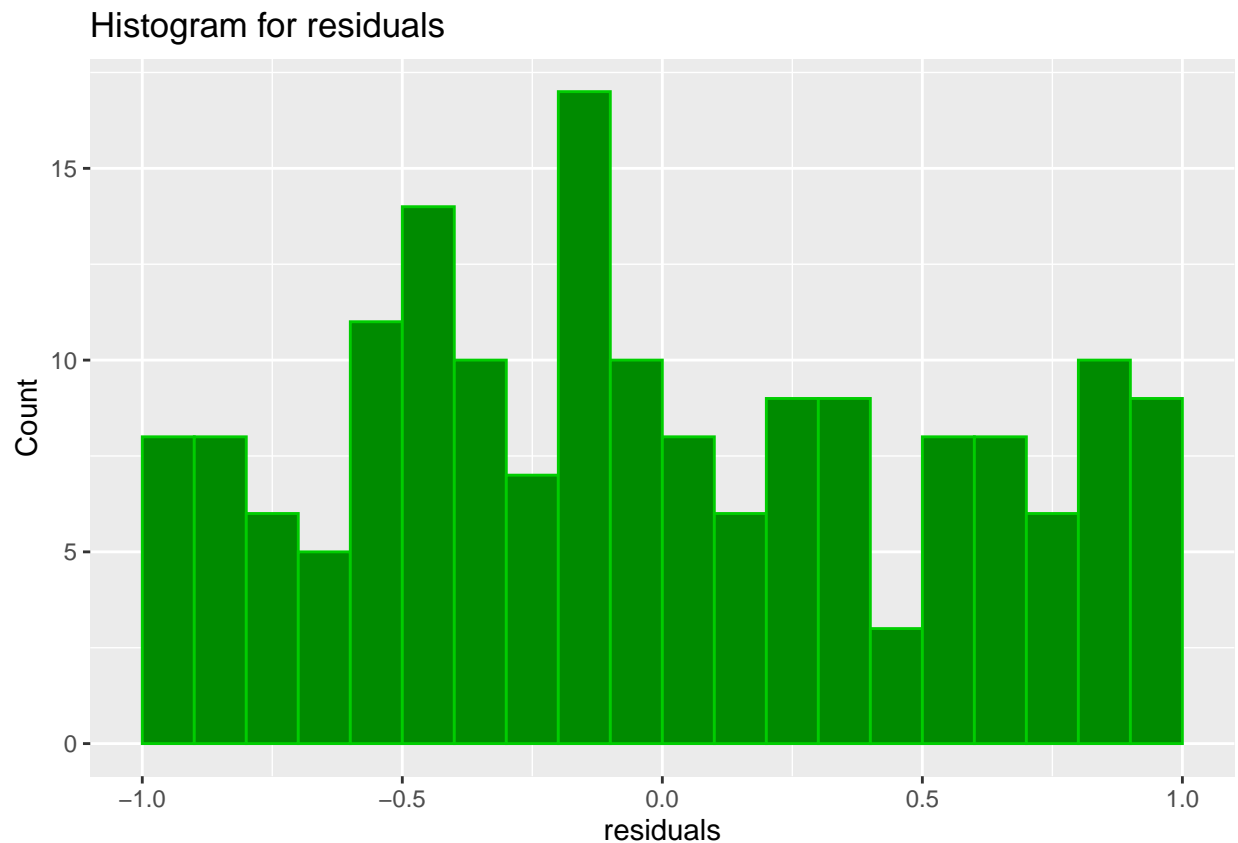
An attempt to address the heteroscedasticity, in addition to any other assumption failure will be done after testing all other assumptions.

NORMALITY

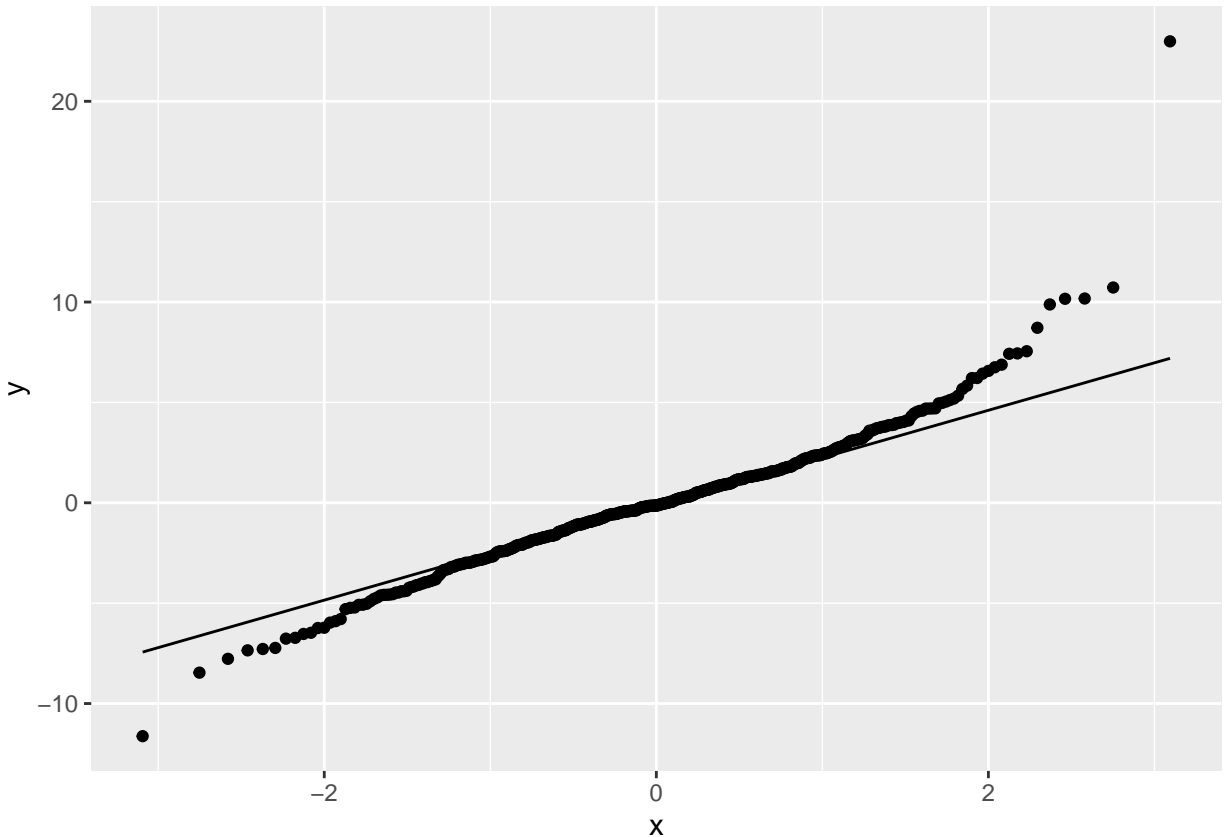
The multiple linear regression analysis requires that the errors between observed and predicted values (i.e., the residuals of the regression) should be normally distributed.

A histogram and q-q plot were developed to check if the residuals were normally distributed or not.

```
ggplot(data=boston_data, aes(residuals(higher_order_interaction_model_2))) +  
geom_histogram(breaks = seq(-1,1,by=0.1), col="green3", fill="green4") +  
labs(title="Histogram for residuals") +  
labs(x="residuals", y="Count")
```



```
ggplot(boston_data, aes(sample=higher_order_interaction_model_2$residuals)) +  
stat_qq() + stat_qq_line()
```



Based on the plots above, it appears the residuals are not normal.

A Shapiro-Wilk normality statistical test was also conducted to confirm that the residuals are not normally distributed

Hypotheses

H_0 : the sample data are significantly normally distributed

H_a : the sample data are not significantly normally distributed

```
#Testing for Normality
shapiro.test(residuals(higher_order_interaction_model_2))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(higher_order_interaction_model_2)
## W = 0.94232, p-value = 4.088e-13
```

The p-value of the Shapiro-Wilk normality test is less than 0.05 (4.088e-13), so we fail to reject the null hypothesis, indicating we do not have normality.

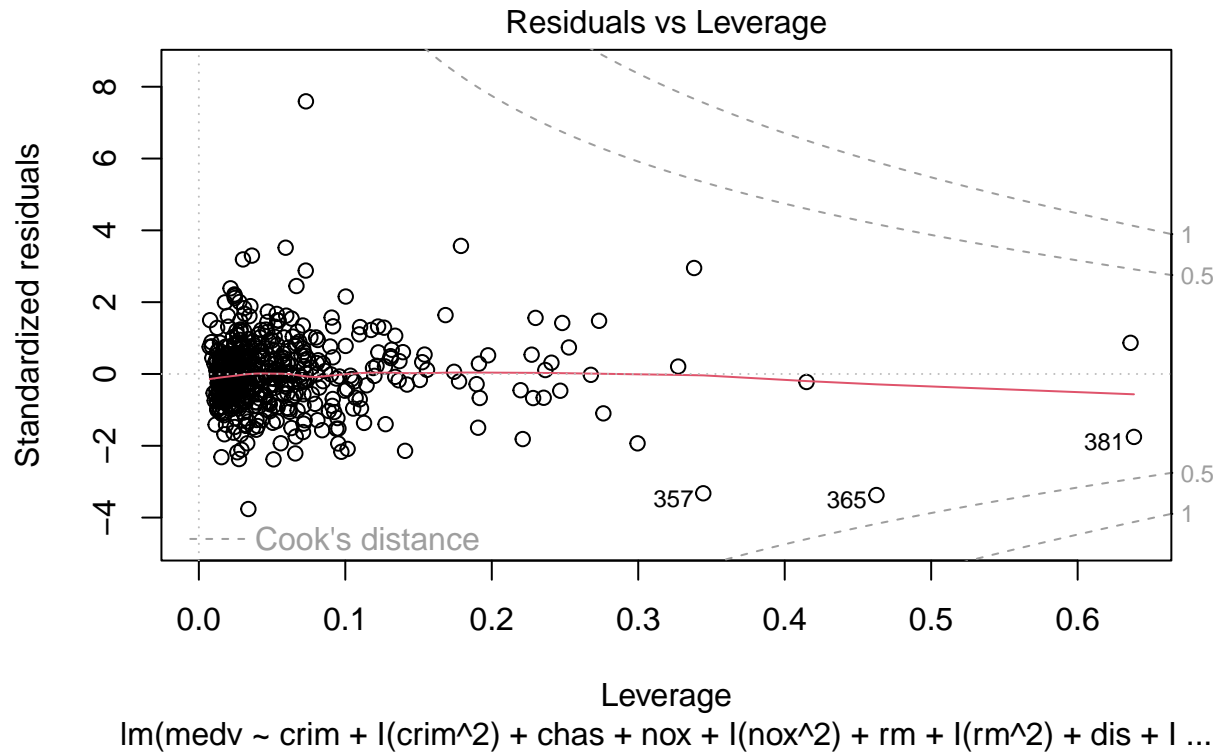
An attempt to address the normality will be done after testing all other assumptions.

OUTLIERS

The below approaches were explored to find and evaluate outliers or influential points.

1. Residuals vs Leverage Plot

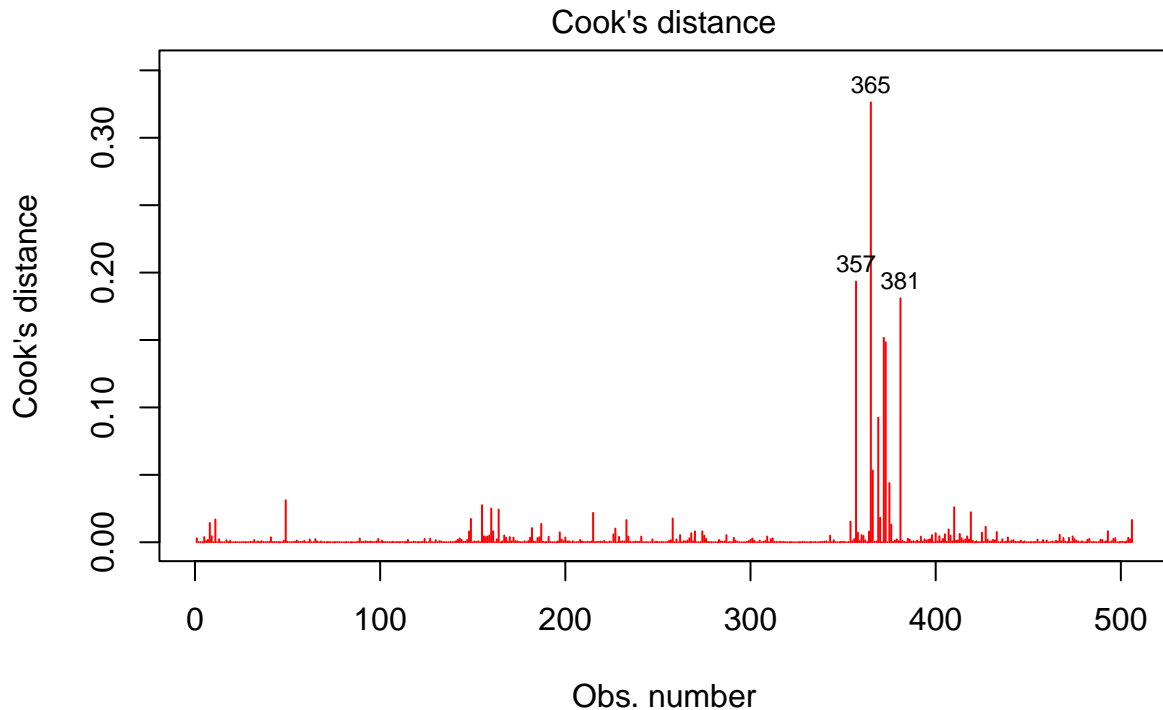
```
plot(higher_order_interaction_model_2, 5)
```



The plot above shows that all cases are well inside of the Cook's distance lines, indicating no outliers or no influential points.

2. Cook's Distance

```
plot(higher_order_interaction_model_2, pch=18, col="red", which=c(4))
```



$\text{lm}(\text{medv} \sim \text{crim} + \text{l}(\text{crim}^2) + \text{chas} + \text{nox} + \text{l}(\text{nox}^2) + \text{rm} + \text{l}(\text{rm}^2) + \text{dis} + \text{l} \dots$

Based on the consensus that a value of more than 1 indicates an influential value, the cook's distance plot above indicates there are no influential values.

3. Leverage points

```
lev=hatvalues(higher_order_interaction_model_2)
p = length(coef(higher_order_interaction_model_2))
n = nrow(boston_data)
outlier2p = lev[lev>(2*p/n)]
outlier3p = lev[lev>(3*p/n)]
print("h_I>2p/n, outliers are")
```

```
## [1] "h_I>2p/n, outliers are"
```

```
print(outlier2p)
```

```
##          9          49          103          142          143          144          145          148
## 0.2354513 0.2211416 0.1189061 0.2196870 0.2272326 0.1541038 0.1270919 0.1264748
##          153          155          156          160          162          163          164          166
## 0.3272566 0.2732434 0.2282326 0.1407945 0.1557776 0.1974755 0.2298756 0.1369647
##          210          212          226          254          258          263          266          274
## 0.1509071 0.1421784 0.1340614 0.2676467 0.1906601 0.1340792 0.1309452 0.1223700
##          283          284          311          353          354          355          356          357
## 0.1225426 0.1912266 0.2467671 0.1213298 0.2761866 0.1525040 0.1316790 0.3444005
##          359          365          366          368          369          370          371          373
```

```
## 0.1311273 0.4626355 0.2995252 0.1896458 0.1789528 0.1684122 0.2365689 0.3381414
##      374      375      381      399      406      407      411      413
## 0.1742488 0.6361285 0.6384477 0.1198824 0.1776729 0.1275039 0.2407394 0.2526555
##      415      419      428      439      491
## 0.4148876 0.2481005 0.1393357 0.1917905 0.1364389
```

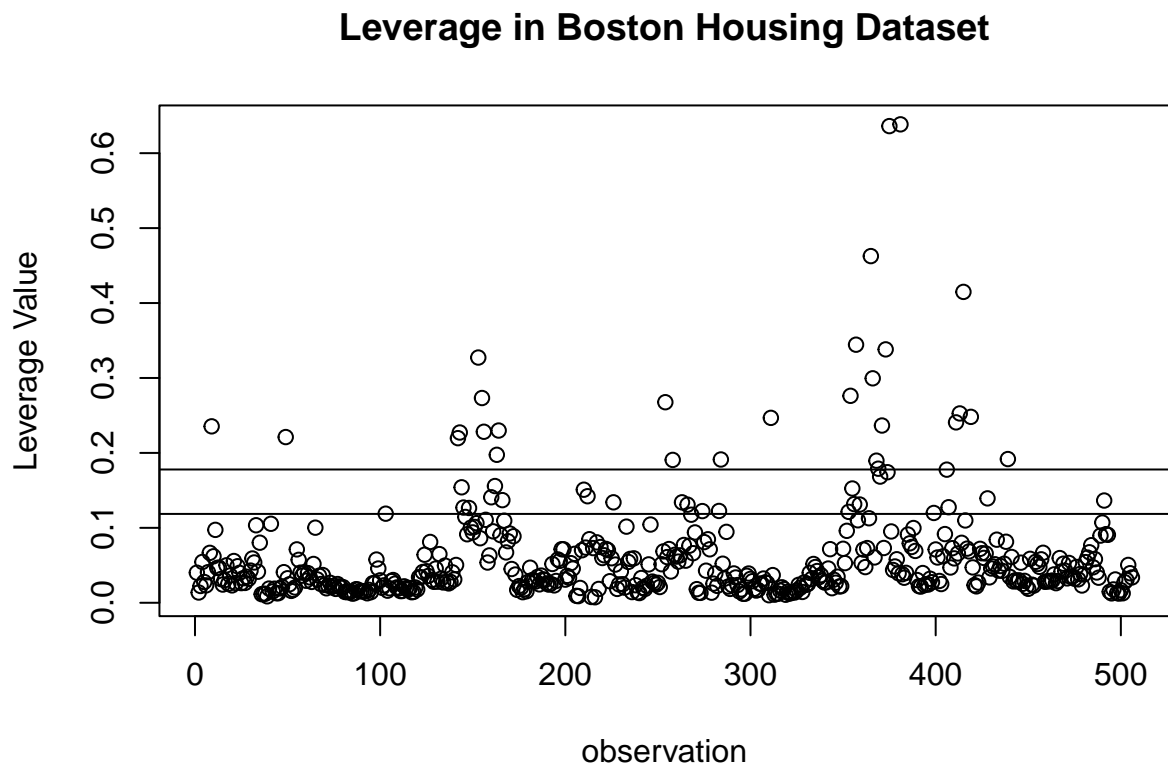
```
print("h_I>3p/n, outliers are")
```

```
## [1] "h_I>3p/n, outliers are"
```

```
print(outlier3p)
```

```
##      9      49      142      143      153      155      156      163
## 0.2354513 0.2211416 0.2196870 0.2272326 0.3272566 0.2732434 0.2282326 0.1974755
##      164      254      258      284      311      354      357      365
## 0.2298756 0.2676467 0.1906601 0.1912266 0.2467671 0.2761866 0.3444005 0.4626355
##      366      368      369      371      373      375      381      411
## 0.2995252 0.1896458 0.1789528 0.2365689 0.3381414 0.6361285 0.6384477 0.2407394
##      413      415      419      439
## 0.2526555 0.4148876 0.2481005 0.1917905
```

```
plot(rownames(boston_data),lev, main = "Leverage in Boston Housing Dataset", xlab="observation", ylab =
abline(h = 2 *p/n, lty = 1)
abline(h = 3 *p/n, lty = 1)
```



Based on the above plot, it appears we have high leverage points but none of them appear to be particularly influential (no points with a concerning cooks distance). Hence, it appears we do not have any outliers that could pose problems.

Dealing with Heteroscedasticity and Normality

Now that all multiple regression assumptions have been tested to check if the model is trustworthy, we then made attempts to address the heteroscedasticity and lack of normality.

We started by making a log transformation of the predictor so the difference between big and small numbers relatively becomes small.

```
higher_order_interaction_model_log = lm(formula = log(medv) ~ crim + I(crim^2) + chas + nox + I(nox^2) +
summary(higher_order_interaction_model_log)
```

```
##
## Call:
## lm(formula = log(medv) ~ crim + I(crim^2) + chas + nox + I(nox^2) +
##      rm + I(rm^2) + dis + I(dis^2) + rad + tax + ptratio + b +
##      lstat + I(lstat^2) + I(lstat^3) + I(lstat^4) + I(lstat^5) +
##      crim:chas + crim:rad + crim:tax + chas:nox + chas:rm + nox:rad +
##      rm:dis + rm:ptratio + dis:rad + dis:lstat + b:lstat, data = boston_data)
##
## Residuals:
```

	Min	1Q	Median	3Q	Max
	-0.64273	-0.07904	-0.00199	0.07810	0.69835

```
##
## Coefficients:
```

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	4.482e+00	8.708e-01	5.147	3.88e-07 ***
crim	-2.790e-01	1.107e-01	-2.521	0.012039 *
I(crim^2)	2.006e-04	4.486e-05	4.471	9.74e-06 ***
chasYes	1.559e+00	3.098e-01	5.031	6.91e-07 ***
nox	-1.865e+00	9.294e-01	-2.006	0.045393 *
I(nox^2)	1.341e+00	7.206e-01	1.861	0.063298 .
rm	-1.186e-01	1.850e-01	-0.641	0.521678
I(rm^2)	3.425e-02	1.021e-02	3.353	0.000863 ***
dis	-4.052e-01	6.144e-02	-6.595	1.13e-10 ***
I(dis^2)	8.126e-03	1.930e-03	4.210	3.06e-05 ***
rad	6.655e-02	1.197e-02	5.560	4.50e-08 ***
tax	-7.812e-04	1.312e-04	-5.953	5.13e-09 ***
ptratio	9.170e-02	3.404e-02	2.694	0.007306 **
b	7.433e-04	2.916e-04	2.549	0.011126 *
lstat	-1.671e-01	4.731e-02	-3.531	0.000454 ***
I(lstat^2)	1.603e-02	6.901e-03	2.323	0.020622 *
I(lstat^3)	-8.833e-04	4.438e-04	-1.990	0.047117 *
I(lstat^4)	2.139e-05	1.282e-05	1.668	0.095950 .
I(lstat^5)	-1.788e-07	1.356e-07	-1.318	0.188160
crim:chasYes	8.007e-02	1.469e-02	5.450	8.10e-08 ***
crim:rad	-1.575e-02	5.576e-03	-2.826	0.004917 **
crim:tax	9.480e-04	3.589e-04	2.641	0.008531 **
chasYes:nox	-1.343e+00	2.708e-01	-4.958	9.92e-07 ***
chasYes:rm	-1.302e-01	3.540e-02	-3.679	0.000261 ***

```
## nox:rad      -6.132e-02  1.518e-02  -4.039  6.25e-05 ***
## rm:dis       3.751e-02  7.044e-03   5.326  1.55e-07 ***
## rm:ptratio   -1.793e-02  5.135e-03  -3.493  0.000523 ***
## dis:rad      -2.618e-03  1.299e-03  -2.016  0.044392 *
## dis:lstat    5.984e-03  9.908e-04   6.040  3.11e-09 ***
## b:lstat      -2.205e-05  1.390e-05  -1.586  0.113403
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1509 on 476 degrees of freedom
## Multiple R-squared:  0.8716, Adjusted R-squared:  0.8638
## F-statistic: 111.4 on 29 and 476 DF,  p-value: < 2.2e-16
```

Some insignificant variables were observed after the log transformation, which were subsequently dropped from the model

```
higher_order_interaction_model_log_2 = lm(formula = log(medv) ~ crim + I(crim^2) + chas + nox + rm + I(rm^2) + dis + I(dis^2) + rad + tax + ptratio + b + lstat + I(lstat^2) + I(lstat^3) + I(lstat^4) + crim:chas + crim:rad + crim:tax + chas:nox + chas:rm + nox:rad + rm:dis + rm:ptratio + dis:lstat, data = boston_data)
summary(higher_order_interaction_model_log_2)
```

```
##
## Call:
## lm(formula = log(medv) ~ crim + I(crim^2) + chas + nox + rm +
##      I(rm^2) + dis + I(dis^2) + rad + tax + ptratio + b + lstat +
##      I(lstat^2) + I(lstat^3) + I(lstat^4) + crim:chas + crim:rad +
##      crim:tax + chas:nox + chas:rm + nox:rad + rm:dis + rm:ptratio +
##      dis:lstat, data = boston_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.65327 -0.07850 -0.00283  0.07869  0.76409
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.759e+00  8.451e-01   5.632 3.04e-08 ***
## crim        -3.149e-01  1.078e-01  -2.922 0.003637 **
## I(crim^2)     1.770e-04  4.308e-05   4.109 4.67e-05 ***
## chasYes       1.524e+00  3.111e-01   4.900 1.32e-06 ***
## nox          -3.748e-01  2.194e-01  -1.709 0.088182 .
## rm           -2.175e-01  1.789e-01  -1.215 0.224816
## I(rm^2)       3.738e-02  1.011e-02   3.698 0.000242 ***
## dis          -4.104e-01  5.850e-02  -7.015 7.89e-12 ***
## I(dis^2)      8.450e-03  1.793e-03   4.712 3.21e-06 ***
## rad           4.594e-02  8.105e-03   5.668 2.50e-08 ***
## tax          -7.783e-04  1.304e-04  -5.971 4.60e-09 ***
## ptratio       6.818e-02  3.254e-02   2.095 0.036670 *
## b             2.943e-04  8.825e-05   3.334 0.000921 ***
## lstat        -1.217e-01  2.434e-02  -5.000 8.07e-07 ***
## I(lstat^2)    7.392e-03  2.371e-03   3.118 0.001932 **
## I(lstat^3)   -3.012e-04  9.302e-05  -3.237 0.001289 **
## I(lstat^4)    4.395e-06  1.242e-06   3.540 0.000439 ***
## crim:chasYes  8.345e-02  1.469e-02   5.682 2.32e-08 ***
## crim:rad     -1.740e-02  5.327e-03  -3.267 0.001163 **
## crim:tax      1.065e-03  3.456e-04   3.081 0.002184 **
```

```
## chasYes:nox -1.302e+00 2.711e-01 -4.805 2.08e-06 ***
## chasYes:rm -1.282e-01 3.538e-02 -3.624 0.000321 ***
## nox:rad -4.075e-02 1.288e-02 -3.165 0.001651 **
## rm:dis 3.698e-02 7.021e-03 5.267 2.10e-07 ***
## rm:ptratio -1.488e-02 4.963e-03 -2.998 0.002861 **
## dis:lstat 5.632e-03 9.484e-04 5.939 5.52e-09 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1521 on 480 degrees of freedom
## Multiple R-squared: 0.8684, Adjusted R-squared: 0.8616
## F-statistic: 126.7 on 25 and 480 DF, p-value: < 2.2e-16
```

The log-transformed model was tested for homoscedasticity and normality.

```
#Testing for Homoscedasticity
bptest(higher_order_interaction_model_log_2)
```

```
##
## studentized Breusch-Pagan test
##
## data: higher_order_interaction_model_log_2
## BP = 76.436, df = 25, p-value = 4.085e-07
```

The p-value of the Breusch-Pagan test is less than 0.05 (4.085e-07), indicating we still have heteroscedasticity.

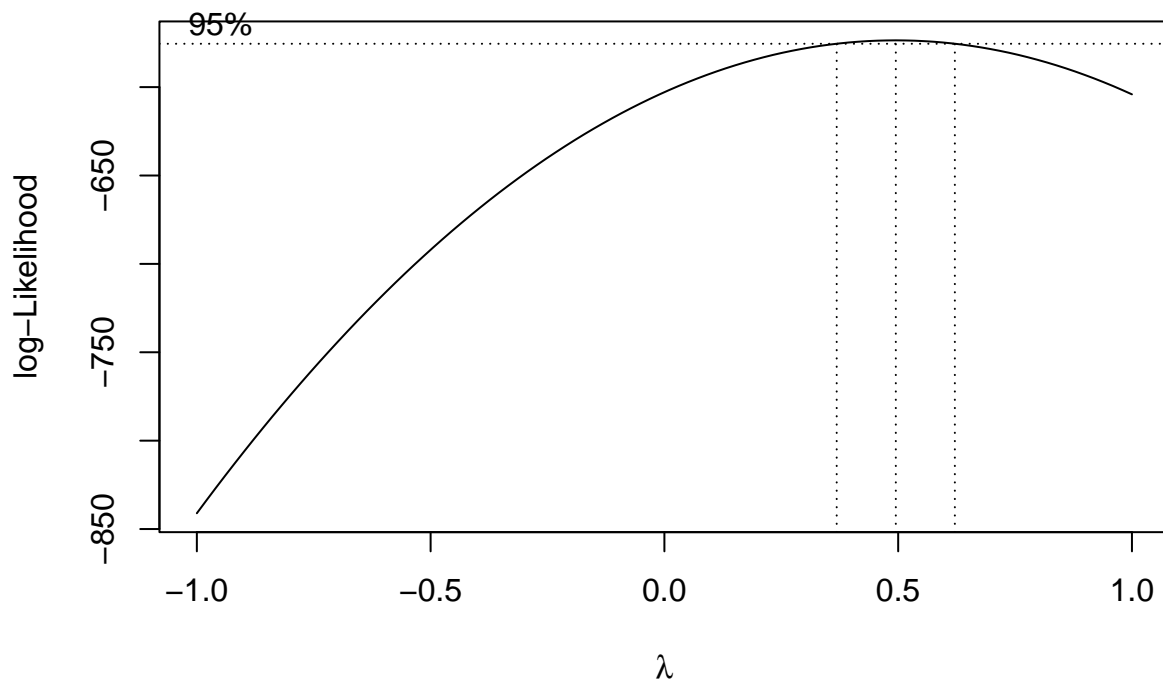
```
#Testing for Normality
shapiro.test(residuals(higher_order_interaction_model_log_2))
```

```
##
## Shapiro-Wilk normality test
##
## data: residuals(higher_order_interaction_model_log_2)
## W = 0.95209, p-value = 9.614e-12
```

The p-value of the Shapiro-Wilk normality test is less than 0.05 (9.614e-12), so we fail to reject the null hypothesis, indicating we do not have normality.

Since the log-transformation approach did not resolve the heteroscedasticity or lack of normality, we then explored the Box-Cox transformations approach by identifying $\hat{\lambda}$, the maximum likelihood estimate of λ to use in the power transformation

```
bc=boxcox(higher_order_interaction_model_2,lambda=seq(-1,1))
```



```
#extract best lambda
bestlambda=bc$x[which(bc$y==max(bc$y))]
bestlambda
```

```
## [1] 0.4949495
```

The best lambda estimate was then used to transform the model as follows:

```
higher_order_interaction_model_2_box = lm(formula = (((medv^0.4949495)-1)/0.4949495) ~ crim + I(crim^2) +
summary(higher_order_interaction_model_2_box)
```

```
##
## Call:
## lm(formula = (((medv^0.4949495) - 1)/0.4949495) ~ crim + I(crim^2) +
##   chas + nox + I(nox^2) + rm + I(rm^2) + dis + I(dis^2) + rad +
##   tax + ptratio + b + lstat + I(lstat^2) + I(lstat^3) + I(lstat^4) +
##   I(lstat^5) + crim:chas + crim:rad + crim:tax + chas:nox +
##   chas:rm + nox:rad + rm:dis + rm:ptratio + dis:rad + dis:lstat +
##   b:lstat, data = boston_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.6756 -0.3719 -0.0053  0.3438  3.8925
##
## Coefficients:
```

```
##           Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.044e+01  3.686e+00   2.831 0.004829 **
## crim        -1.301e+00  4.686e-01  -2.775 0.005730 **
## I(crim^2)     7.403e-04  1.899e-04   3.899 0.000111 ***
## chasYes       7.684e+00  1.311e+00   5.860 8.65e-09 ***
## nox          -8.847e+00  3.934e+00  -2.249 0.024983 *
## I(nox^2)      6.398e+00  3.050e+00   2.097 0.036501 *
## rm           -3.191e-02  7.830e-01  -0.041 0.967515
## I(rm^2)       1.788e-01  4.324e-02   4.135 4.19e-05 ***
## dis          -1.926e+00  2.601e-01  -7.405 6.00e-13 ***
## I(dis^2)      4.004e-02  8.171e-03   4.900 1.32e-06 ***
## rad           2.896e-01  5.067e-02   5.716 1.92e-08 ***
## tax          -3.453e-03  5.555e-04  -6.216 1.12e-09 ***
## ptratio       6.587e-01  1.441e-01   4.572 6.17e-06 ***
## b             3.889e-03  1.235e-03   3.150 0.001737 **
## lstat        -9.961e-01  2.003e-01  -4.973 9.20e-07 ***
## I(lstat^2)    1.004e-01  2.921e-02   3.438 0.000637 ***
## I(lstat^3)   -5.598e-03  1.879e-03  -2.980 0.003028 **
## I(lstat^4)    1.433e-04  5.427e-05   2.640 0.008575 **
## I(lstat^5)   -1.328e-06  5.742e-07  -2.314 0.021119 *
## crim:chasYes  4.094e-01  6.219e-02   6.583 1.22e-10 ***
## crim:rad      -7.339e-02  2.360e-02  -3.109 0.001987 **
## crim:tax       4.458e-03  1.519e-03   2.934 0.003509 **
## chasYes:nox   -6.666e+00  1.146e+00  -5.816 1.11e-08 ***
## chasYes:rm    -6.423e-01  1.498e-01  -4.287 2.19e-05 ***
## nox:rad       -2.647e-01  6.427e-02  -4.118 4.51e-05 ***
## rm:dis        1.742e-01  2.982e-02   5.843 9.49e-09 ***
## rm:ptratio    -1.198e-01  2.174e-02  -5.510 5.90e-08 ***
## dis:rad       -1.305e-02  5.498e-03  -2.374 0.017982 *
## dis:lstat     2.924e-02  4.194e-03   6.973 1.04e-11 ***
## b:lstat      -1.286e-04  5.886e-05  -2.184 0.029452 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6387 on 476 degrees of freedom
## Multiple R-squared:  0.8868, Adjusted R-squared:  0.8799
## F-statistic: 128.6 on 29 and 476 DF,  p-value: < 2.2e-16
```

The best lambda transformed model was then tested for homoscedasticity and normality.

```
bptest(higher_order_interaction_model_2_box)
```

```
##
## studentized Breusch-Pagan test
##
## data:  higher_order_interaction_model_2_box
## BP = 68.228, df = 29, p-value = 5.277e-05
```

The p-value of the Breusch-Pagan test is less than 0.05 (5.277e-05), indicating we still have heteroscedasticity.

```
#Testing for Normality
shapiro.test(residuals(higher_order_interaction_model_2_box))
```



```
##
## Shapiro-Wilk normality test
##
## data: residuals(higher_order_interaction_model_2_box)
## W = 0.96285, p-value = 5.297e-10
```

The p-value of the Shapiro-Wilk normality test is less than 0.05 (5.297e-10), so we fail to reject the null hypothesis, indicating we do not have normality.

Since the box-cox transformation approach did not resolve the heteroscedasticity or lack of normality also, we then explored the weighted least squares regression method, which is an application of the more general concept of generalized least squares. This method gives less weight to observations with high variance as follows:

```
#Determine absolute residuals
```

```
resid = resid(higher_order_interaction_model_2)
```

```
#Fit a model to the residuals to model the variance pattern
```

```
resid_model = lm(formula = log(resid^2) ~ crim + I(crim^2) + chas + nox + I(nox^2) + rm + I(rm^2) + dis + I(dis^2))
```

```
#Create weights using fitted variances
```

```
fitted_var = exp(fitted(resid_model))
```

```
weights = 1 / fitted_var
```

```
#Re-fit the model with weights
```

```
higher_order_interaction_model_wls = lm(formula = medv ~ crim + I(crim^2) + chas + nox + I(nox^2) + rm + I(rm^2) + dis + I(dis^2), data = boston_data, weights = weights)
summary(higher_order_interaction_model_wls)
```

```
##
## Call:
## lm(formula = medv ~ crim + I(crim^2) + chas + nox + I(nox^2) +
##     rm + I(rm^2) + dis + I(dis^2) + rad + tax + ptratio + b +
##     lstat + I(lstat^2) + I(lstat^3) + I(lstat^4) + I(lstat^5) +
##     crim:chas + crim:rad + crim:tax + chas:nox + chas:rm + nox:rad +
##     rm:dis + rm:ptratio + dis:rad + dis:lstat + b:lstat, data = boston_data,
##     weights = weights)
##
## Weighted Residuals:
##   Min      1Q  Median      3Q      Max
## -8.477 -1.287 -0.069  1.153  9.871
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.264e+01  1.853e+01   3.380 0.000784 ***
## crim        -5.864e+00  1.863e+00  -3.147 0.001751 **
## I(crim^2)     3.262e-03  1.083e-03   3.012 0.002729 **
## chasYes       2.596e+01  7.130e+00   3.641 0.000302 ***
## nox          -3.232e+01  1.479e+01  -2.185 0.029384 *
## I(nox^2)       1.951e+01  1.234e+01   1.581 0.114486
## rm           -1.009e+01  4.277e+00  -2.359 0.018718 *
## I(rm^2)        1.731e+00  2.554e-01   6.777 3.63e-11 ***
## dis          -7.457e+00  1.101e+00  -6.772 3.76e-11 ***
## I(dis^2)       1.581e-01  3.367e-02   4.695 3.49e-06 ***
```

```
## rad          8.730e-01  2.617e-01   3.336 0.000918 ***
## tax         -1.299e-02  2.122e-03  -6.122 1.94e-09 ***
## ptratio      3.106e+00  6.968e-01   4.458 1.03e-05 ***
## b            1.985e-02  5.604e-03   3.543 0.000435 ***
## lstat        -5.062e+00  8.233e-01  -6.149 1.65e-09 ***
## I(lstat^2)    5.210e-01  1.195e-01   4.361 1.59e-05 ***
## I(lstat^3)   -2.814e-02  7.712e-03  -3.649 0.000292 ***
## I(lstat^4)    7.116e-04  2.229e-04   3.193 0.001502 **
## I(lstat^5)   -6.684e-06  2.352e-06  -2.841 0.004684 **
## crim:chasYes  1.616e+00  3.828e-01   4.221 2.92e-05 ***
## crim:rad     -3.063e-01  1.045e-01  -2.931 0.003544 **
## crim:tax      1.925e-02  6.413e-03   3.002 0.002819 **
## chasYes:nox   -2.508e+01  5.996e+00  -4.182 3.44e-05 ***
## chasYes:rm    -1.907e+00  8.498e-01  -2.244 0.025312 *
## nox:rad       -6.934e-01  3.302e-01  -2.100 0.036237 *
## rm:dis        6.555e-01  1.270e-01   5.162 3.60e-07 ***
## rm:ptratio    -5.768e-01  1.088e-01  -5.303 1.75e-07 ***
## dis:rad       -3.352e-02  2.448e-02  -1.369 0.171583
## dis:lstat     1.010e-01  1.649e-02   6.123 1.93e-09 ***
## b:lstat       -6.805e-04  2.652e-04  -2.566 0.010583 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.199 on 476 degrees of freedom
## Multiple R-squared:  0.8832, Adjusted R-squared:  0.8761
## F-statistic: 124.1 on 29 and 476 DF,  p-value: < 2.2e-16
```

Some insignificant variables were observed, which were subsequently dropped from the model

```
higher_order_interaction_model_wls_2 = lm(formula = medv ~ crim + I(crim^2) + chas + nox + rm + I(rm^2) +
summary(higher_order_interaction_model_wls_2)
```

```
##
## Call:
## lm(formula = medv ~ crim + I(crim^2) + chas + nox + rm + I(rm^2) +
##     dis + I(dis^2) + rad + tax + ptratio + b + lstat + I(lstat^2) +
##     I(lstat^3) + I(lstat^4) + I(lstat^5) + crim:chas + crim:rad +
##     crim:tax + chas:nox + chas:rm + nox:rad + rm:dis + rm:ptratio +
##     dis:lstat, data = boston_data, weights = weights)
##
## Weighted Residuals:
##      Min       1Q   Median       3Q      Max
## -8.547 -1.227 -0.087  1.148 10.212
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.851e+01  1.828e+01   3.748 0.000200 ***
## crim        -6.236e+00  1.814e+00  -3.438 0.000636 ***
## I(crim^2)     2.789e-03  1.022e-03   2.728 0.006606 **
## chasYes       2.499e+01  7.131e+00   3.504 0.000501 ***
## nox          -1.231e+01  3.935e+00  -3.127 0.001871 **
## rm           -1.206e+01  4.154e+00  -2.902 0.003879 **
## I(rm^2)       1.811e+00  2.524e-01   7.175 2.77e-12 ***
```

```
## dis          -7.574e+00  1.083e+00 -6.994 9.03e-12 ***
## I(dis^2)      1.672e-01  3.292e-02  5.078 5.48e-07 ***
## rad          5.139e-01  1.532e-01  3.354 0.000861 ***
## tax         -1.278e-02  2.092e-03 -6.106 2.11e-09 ***
## ptratio      2.741e+00  6.697e-01  4.093 5.00e-05 ***
## b           1.939e-02  5.590e-03  3.468 0.000571 ***
## lstat       -4.996e+00  8.246e-01 -6.059 2.78e-09 ***
## I(lstat^2)    5.105e-01  1.196e-01  4.267 2.39e-05 ***
## I(lstat^3)   -2.744e-02  7.720e-03 -3.555 0.000416 ***
## I(lstat^4)    6.927e-04  2.231e-04  3.104 0.002019 **
## I(lstat^5)   -6.510e-06  2.355e-06 -2.764 0.005931 **
## crim:chasYes  1.565e+00  3.825e-01  4.091 5.04e-05 ***
## crim:rad     -3.414e-01  9.951e-02 -3.431 0.000654 ***
## crim:tax      2.115e-02  6.163e-03  3.431 0.000654 ***
## chasYes:nox   -2.337e+01  5.926e+00 -3.944 9.23e-05 ***
## chasYes:rm    -1.883e+00  8.507e-01 -2.214 0.027332 *
## nox:rad       -2.886e-01  2.507e-01 -1.151 0.250115
## rm:dis        6.478e-01  1.272e-01  5.091 5.13e-07 ***
## rm:ptratio    -5.280e-01  1.056e-01 -4.999 8.12e-07 ***
## dis:lstat     9.682e-02  1.638e-02  5.910 6.53e-09 ***
## b:lstat       -6.665e-04  2.640e-04 -2.525 0.011904 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.204 on 478 degrees of freedom
## Multiple R-squared:  0.8822, Adjusted R-squared:  0.8755
## F-statistic: 132.5 on 27 and 478 DF, p-value: < 2.2e-16
```

A further insignificant variables was observed, which was also dropped from the model

```
higher_order_interaction_model_wls_3 = lm(formula = medv ~ crim + I(crim^2) + chas + nox + rm + I(rm^2) +
summary(higher_order_interaction_model_wls_3)
```

```
##
## Call:
## lm(formula = medv ~ crim + I(crim^2) + chas + nox + rm + I(rm^2) +
##     dis + I(dis^2) + rad + tax + ptratio + b + lstat + I(lstat^2) +
##     I(lstat^3) + I(lstat^4) + I(lstat^5) + crim:chas + crim:rad +
##     crim:tax + chas:nox + chas:rm + rm:dis + rm:ptratio + dis:lstat +
##     b:lstat, data = boston_data, weights = weights)
##
## Weighted Residuals:
##      Min       1Q   Median       3Q      Max
## -8.5191 -1.2640 -0.0935  1.1285 10.2246
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.203e+01  1.803e+01   3.995 7.50e-05 ***
## crim        -6.281e+00  1.814e+00  -3.463 0.000582 ***
## I(crim^2)     3.005e-03  1.005e-03   2.989 0.002940 **
## chasYes      2.461e+01  7.126e+00   3.454 0.000602 ***
## nox         -1.537e+01  2.901e+00  -5.298 1.79e-07 ***
## rm          -1.246e+01  4.141e+00  -3.008 0.002767 **
```

```
## I(rm^2)      1.827e+00  2.520e-01   7.251 1.68e-12 ***
## dis         -7.739e+00  1.074e+00  -7.207 2.24e-12 ***
## I(dis^2)     1.715e-01  3.272e-02   5.242 2.38e-07 ***
## rad          3.466e-01  4.880e-02   7.104 4.42e-12 ***
## tax         -1.298e-02  2.085e-03  -6.227 1.04e-09 ***
## ptratio      2.704e+00  6.692e-01   4.041 6.21e-05 ***
## b            1.990e-02  5.575e-03   3.570 0.000392 ***
## lstat        -4.951e+00  8.240e-01  -6.009 3.70e-09 ***
## I(lstat^2)    5.050e-01  1.196e-01   4.223 2.89e-05 ***
## I(lstat^3)   -2.711e-02  7.717e-03  -3.513 0.000485 ***
## I(lstat^4)    6.835e-04  2.231e-04   3.064 0.002305 **
## I(lstat^5)   -6.415e-06  2.355e-06  -2.725 0.006672 **
## crim:chasYes  1.485e+00  3.762e-01   3.946 9.12e-05 ***
## crim:rad      -3.624e-01  9.786e-02  -3.704 0.000237 ***
## crim:tax       2.194e-02  6.126e-03   3.582 0.000376 ***
## chasYes:nox   -2.275e+01  5.904e+00  -3.853 0.000132 ***
## chasYes:rm    -1.872e+00  8.510e-01  -2.200 0.028256 *
## rm:dis         6.587e-01  1.269e-01   5.190 3.11e-07 ***
## rm:ptratio    -5.220e-01  1.055e-01  -4.946 1.05e-06 ***
## dis:lstat      9.832e-02  1.634e-02   6.018 3.51e-09 ***
## b:lstat       -6.878e-04  2.634e-04  -2.611 0.009320 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.205 on 479 degrees of freedom
## Multiple R-squared:  0.8818, Adjusted R-squared:  0.8754
## F-statistic: 137.5 on 26 and 479 DF,  p-value: < 2.2e-16
```

The model derived from the weighted least squares regression method was tested for homoscedasticity and normality.

```
bptest(higher_order_interaction_model_wls_3)
```

```
##
## studentized Breusch-Pagan test
##
## data:  higher_order_interaction_model_wls_3
## BP = 11.552, df = 26, p-value = 0.9934
```

The p-value of the Breusch-Pagan test is greater than 0.05 (0.9934), indicating we now finally have homoscedasticity.

```
#Testing for Normality
shapiro.test(residuals(higher_order_interaction_model_wls_3))
```

```
##
## Shapiro-Wilk normality test
##
## data:  residuals(higher_order_interaction_model_wls_3)
## W = 0.88798, p-value < 2.2e-16
```

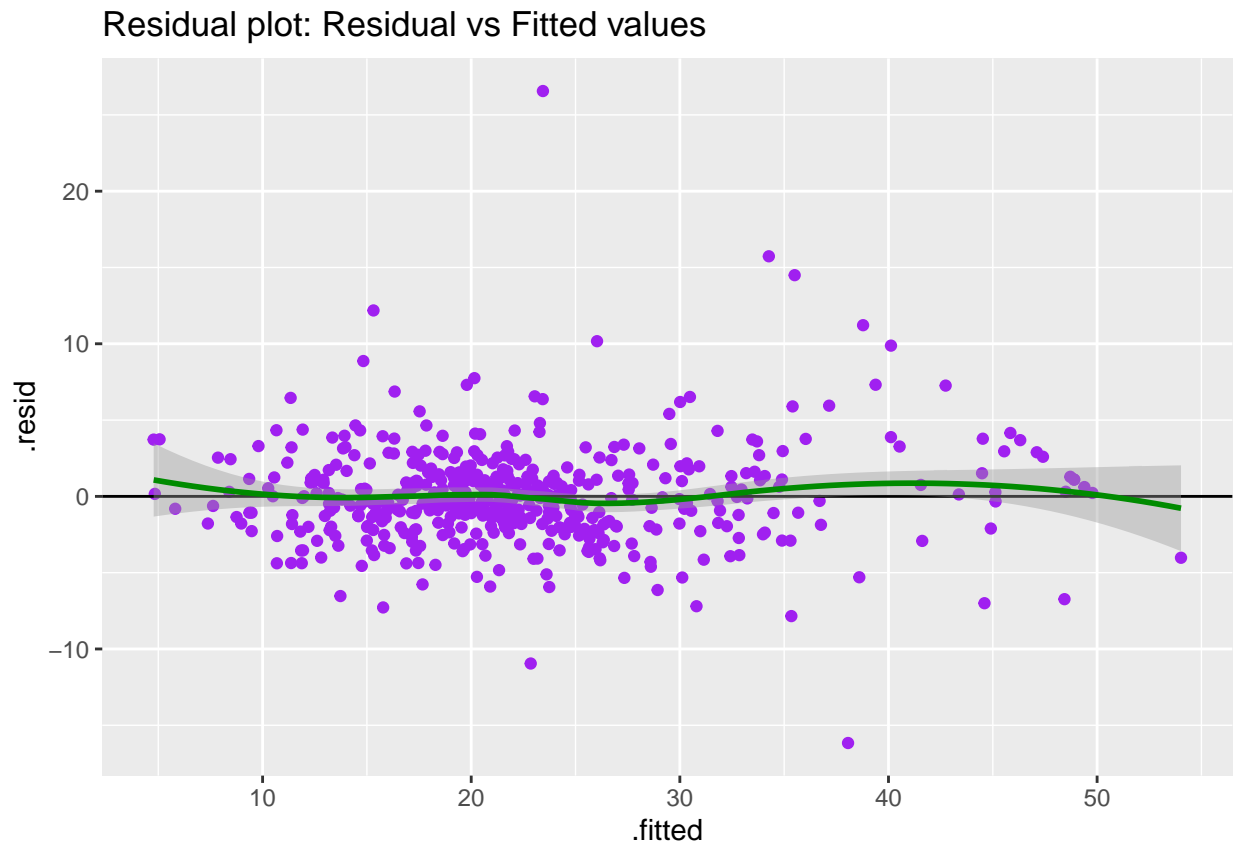
The p-value of the Shapiro-Wilk normality test is less than 0.05 (9.614e-12), indicating we still do not have normality.

The linearity and outliers assumptions would be tested again to confirm they are still met

Linearity Assumption

```
ggplot(higher_order_interaction_model_wls_3, aes(x=.fitted, y=.resid)) +  
  geom_point(colour = "purple") +  
  geom_hline(yintercept = 0) +  
  geom_smooth(colour = "green4")+  
  ggtitle("Residual plot: Residual vs Fitted values")
```

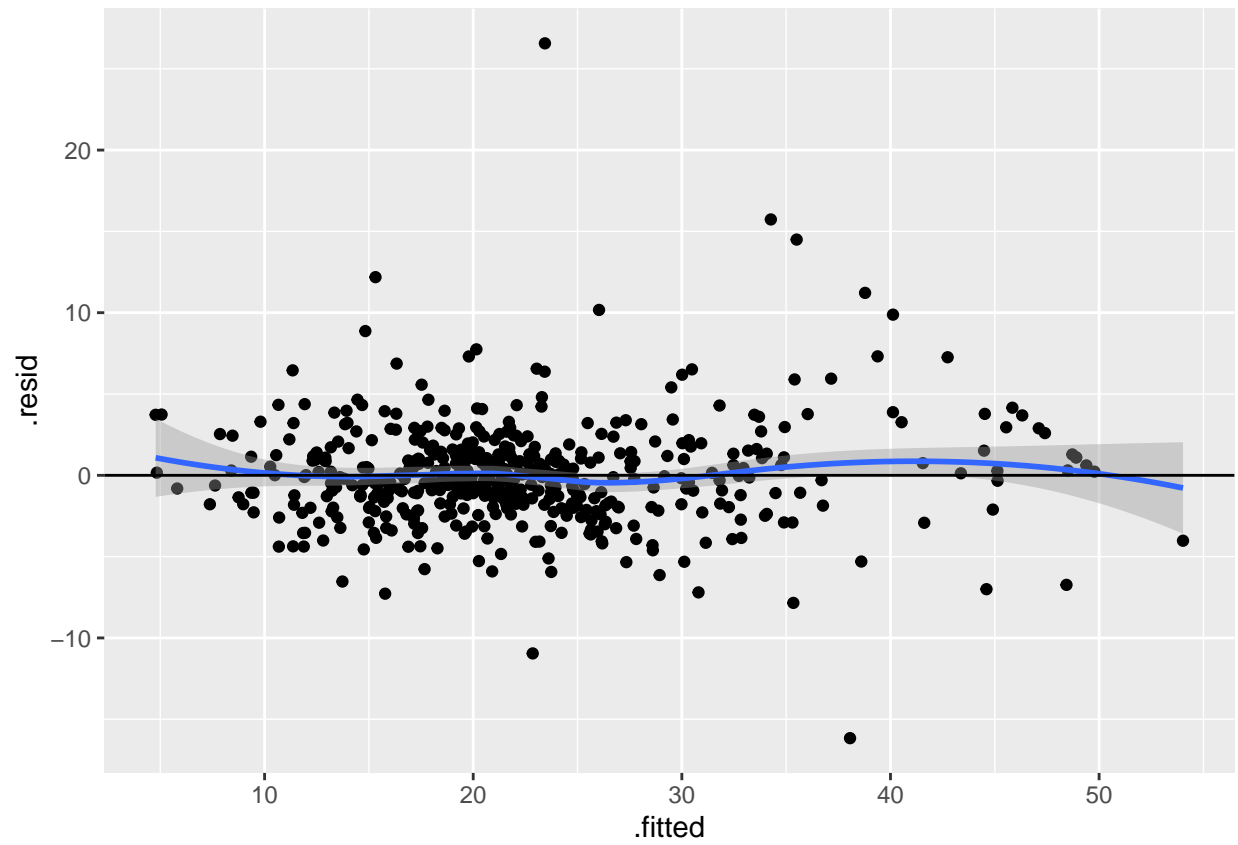
```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```



Based on the updated model, there appears to be no pattern of the residuals at all, indicating that the model still passes the linearity assumption that there is a straight-line (linear) relationship between the predictors and the response.

```
ggplot(higher_order_interaction_model_wls_3, aes(x=.fitted, y=.resid)) +  
  geom_point() +geom_smooth()+  
  geom_hline(yintercept = 0)
```

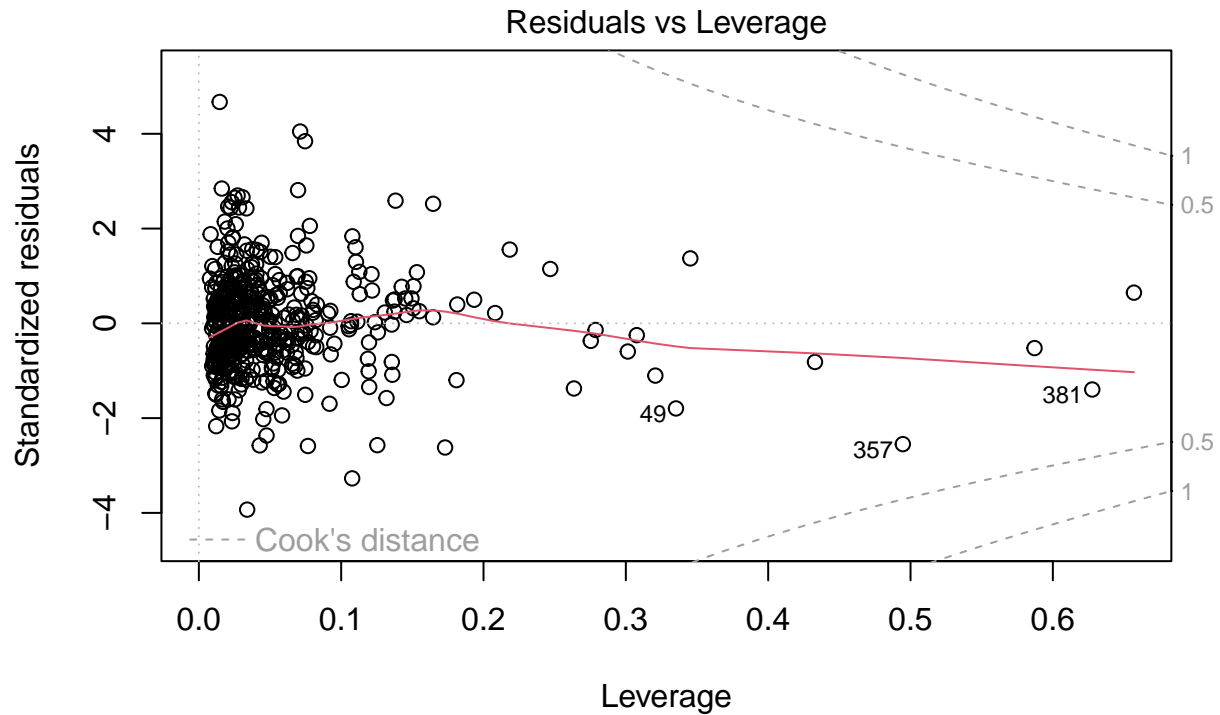
```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```



Outliers

1. Residuals vs Leverage Plot

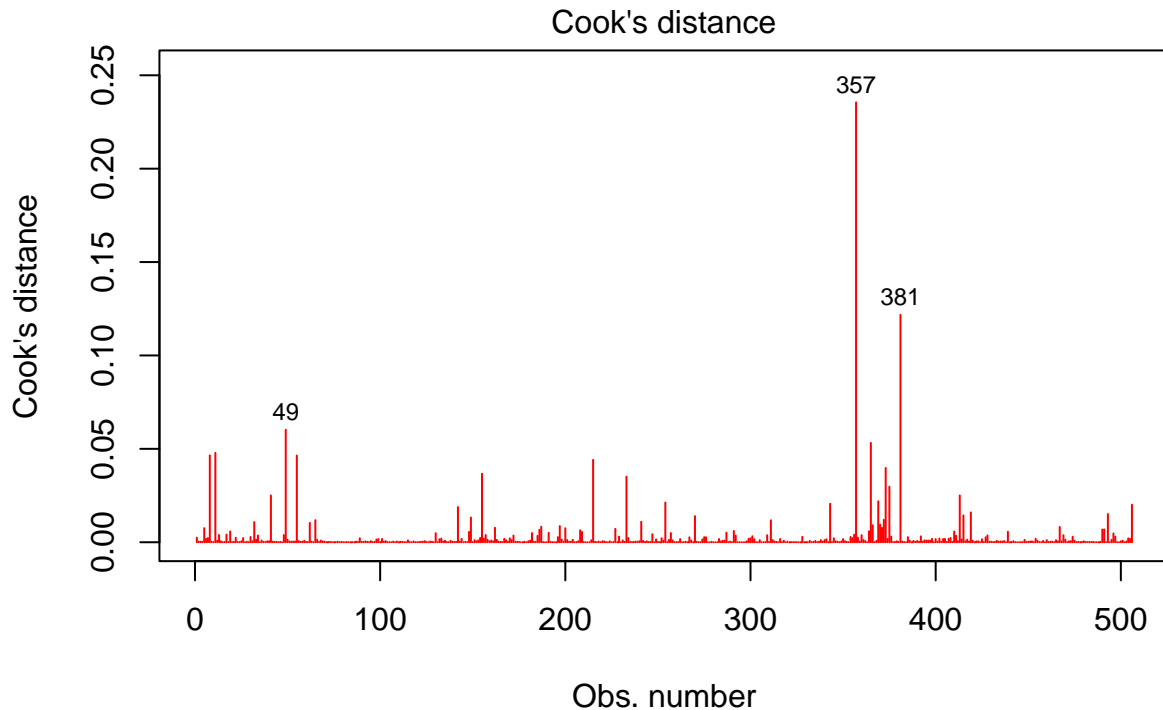
```
plot(higher_order_interaction_model_wls_3, 5)
```



The plot above shows that all cases are well inside of the Cook's distance lines, indicating no outliers or no influential points.

2. Cook's Distance

```
plot(higher_order_interaction_model_wls_3, pch=18, col="red", which=c(4))
```



$\text{lm}(\text{medv} \sim \text{crim} + \text{l}(\text{crim}^2) + \text{chas} + \text{nox} + \text{rm} + \text{l}(\text{rm}^2) + \text{dis} + \text{l}(\text{dis}^2) + \text{r} \dots$

Based on the consensus that a value of more than 1 indicates an influential value, the cook's distance plot above indicates there are still no influential values.

3. Leverage points

```
lev=hatvalues(higher_order_interaction_model_2)
p = length(coef(higher_order_interaction_model_2))
n = nrow(boston_data)
outlier2p = lev[lev>(2*p/n)]
outlier3p = lev[lev>(3*p/n)]
print("h_I>2p/n, outliers are")
```

```
## [1] "h_I>2p/n, outliers are"
```

```
print(outlier2p)
```

```
##          9          49          103          142          143          144          145          148
## 0.2354513 0.2211416 0.1189061 0.2196870 0.2272326 0.1541038 0.1270919 0.1264748
##          153          155          156          160          162          163          164          166
## 0.3272566 0.2732434 0.2282326 0.1407945 0.1557776 0.1974755 0.2298756 0.1369647
##          210          212          226          254          258          263          266          274
## 0.1509071 0.1421784 0.1340614 0.2676467 0.1906601 0.1340792 0.1309452 0.1223700
##          283          284          311          353          354          355          356          357
## 0.1225426 0.1912266 0.2467671 0.1213298 0.2761866 0.1525040 0.1316790 0.3444005
##          359          365          366          368          369          370          371          373
```



```
## 0.1311273 0.4626355 0.2995252 0.1896458 0.1789528 0.1684122 0.2365689 0.3381414
##      374      375      381      399      406      407      411      413
## 0.1742488 0.6361285 0.6384477 0.1198824 0.1776729 0.1275039 0.2407394 0.2526555
##      415      419      428      439      491
## 0.4148876 0.2481005 0.1393357 0.1917905 0.1364389
```

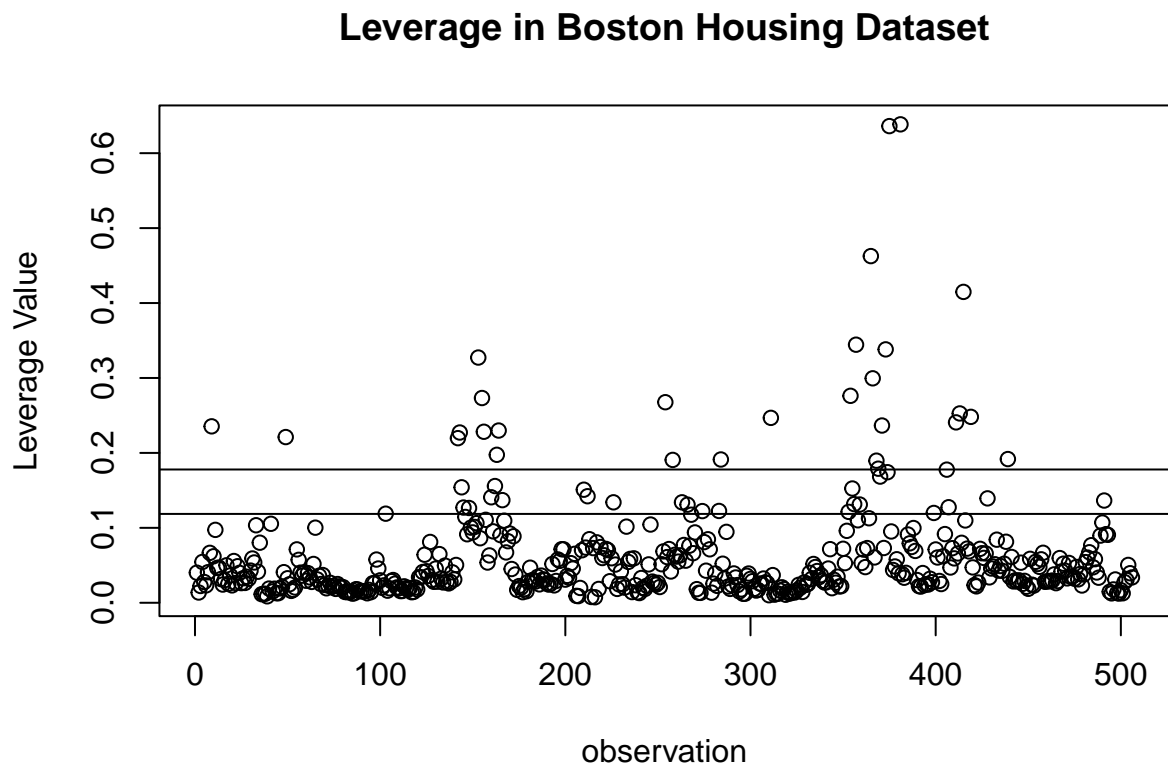
```
print("h_I>3p/n, outliers are")
```

```
## [1] "h_I>3p/n, outliers are"
```

```
print(outlier3p)
```

```
##      9      49      142      143      153      155      156      163
## 0.2354513 0.2211416 0.2196870 0.2272326 0.3272566 0.2732434 0.2282326 0.1974755
##      164      254      258      284      311      354      357      365
## 0.2298756 0.2676467 0.1906601 0.1912266 0.2467671 0.2761866 0.3444005 0.4626355
##      366      368      369      371      373      375      381      411
## 0.2995252 0.1896458 0.1789528 0.2365689 0.3381414 0.6361285 0.6384477 0.2407394
##      413      415      419      439
## 0.2526555 0.4148876 0.2481005 0.1917905
```

```
plot(rownames(boston_data),lev, main = "Leverage in Boston Housing Dataset", xlab="observation", ylab =
abline(h = 2 *p/n, lty = 1)
abline(h = 3 *p/n, lty = 1)
```



Based on the above plot, it appears we still have high leverage points but none of them appear to be particularly influential (no points with a concerning cooks distance). Hence, it appears we do not have any outliers that could pose problems.

4. CONCLUSION AND DISCUSSION

We developed a predictive model using multiple linear regression to estimate the median value of owner-occupied homes in Boston based on 13 housing and neighborhood features. After loading the data, we tested for multicollinearity; where none was detected. Next, we built a baseline additive model and refined it by removing insignificant predictors. Afterwards, Interaction effects were explored, and a reduced interaction model was selected based on significance and model performance. Then we explored higher order terms to capture non-linear relationships, and we iteratively refined the model to retain only the significant predictors. Finally we tested the multiple regression assumptions: which included linearity, independence, equal variance, normality, and outliers. While the final model met most assumptions, the residuals remained non-normal despite attempts with log, Box-Cox, and weighted least squares transformations. Ultimately, the weighted least squares regression model was selected as the final model, as it effectively treated heteroscedasticity, improved model interpretability, and maintained linearity and robustness against outliers. This comprehensive modeling process provided insights into key factors influencing housing prices, our final predictive model is:

$$\begin{aligned} \hat{medv} = & 72.03 - 6.281crim + 0.003005I(crim^2) + 24.61chas - 15.37nox - 12.46rm + 1.827I(rm^2) - 7.739dis + \\ & 0.1715I(dis^2) + 0.3466rad - 0.01298tax + 2.704ptratio + 0.01990b - 4.951lstat + 0.5050I(lstat^2) - \\ & 0.02711I(lstat^3) + 0.0006835I(lstat^4) - 0.000006415I(lstat^5) + 1.485crim : chas - 0.3624crim : \\ & rad + 0.02194crim : tax - 22.75chas : nox - 1.872chas : rm + 0.6587rm : dis - 0.5220rm : ptratio + 0.09832dis : \\ & lstat - 0.0006878b : lstat \end{aligned}$$

Based off of this, the key factors that predict the median value (in thousands of dollars) are:

crim, $I(crim^2)$, chas, nox, rm, $I(rm^2)$, dis, $I(dis^2)$, rad, tax, ptratio, b, lstat, $I(lstat^2)$, $I(lstat^3)$, $I(lstat^4)$, $I(lstat^5)$, crim:chas, crim:rad, crim:tax, chas:nox, chas:rm, rm:dis, rm:ptratio, dis:lstat, b:lstat

4.1 Approach The approach we took does seem logically sound. Starting with a basic multiple linear regression model allowed us to build a strong baseline and interpret the effects of various predictors on housing prices. Looking at interaction terms and higher-order relationships enabled us to improve model complexity effectively. Additionally, testing each of the regression assumptions step-by-step ensured that our final model would be valid statistically. Some issues we had were that there was consistent violation of the normality assumption, even with various transformations. Possibilities for improvement are possibly running non-linear or machine learning models. They may be able to handle normality and allow us to create a better predictive model.

4.2 Future Work

5. REFERENCES

[1] - Dataset description