Bitcoin closing price prediction
Can key indicators be found to predict next day closing price

Laura Pelayo, Shawn Petersen, Josh Pickel, Justin Pickel *Washington State University*

This research paper provides an analyis of key stock indicators and can they be used to predict next day closing price.

Keywords: linear lasso ridge gam correlation cross validation knots

December 10, 2021

1 Introduction

Bitcoin (BTC) is a consensus network that enables a new payment system and a fully digital currency. Powered by its users, it peers into a payment network that does not require a central authority to operate. On October 31, 2008, a person or group of people working under the pseudonym "Satoshi Nakamoto" published the Bitcoin Whitepaper and described it as: "A completely peer-to-peer version of electronic cash, which allows you to send online payments directly. From party to another party.

Our team has decided to answer the question can we determine the best model to accurately predict the closing price of bitcoin? We will various technical indicators commonly used in financial trading as our predictor variables and the closing price of bitcoin as our response variable. Our team would also like to determine which predictors are statistically significant in predicting the closing price of bitcoin.

Our team is going to evaluate the accuracy of four models:

- 1. A simple linear regression model
- 2. A Lasso regression model.
- 3. A ridge regression model.
- 4. A GAM model.

We are going to split the data into test and train datasets in order to compare the test MSE across our models. The model with the lowest test MSE will be considered the best model to predict the closing price of bitcoin.

Our data set 'bitcoin_clean2.csv' was created by pulling in bitcoin price data from yahoo finance over the last 2 years. We then added several technical indicators to our data set by utilizing a technical indicator package in python. In order for our model to be useful we decided to shift the technical indicator data by one day as this allows us to determine if we can use the previous days technical indicator data to predict the current days closing price.

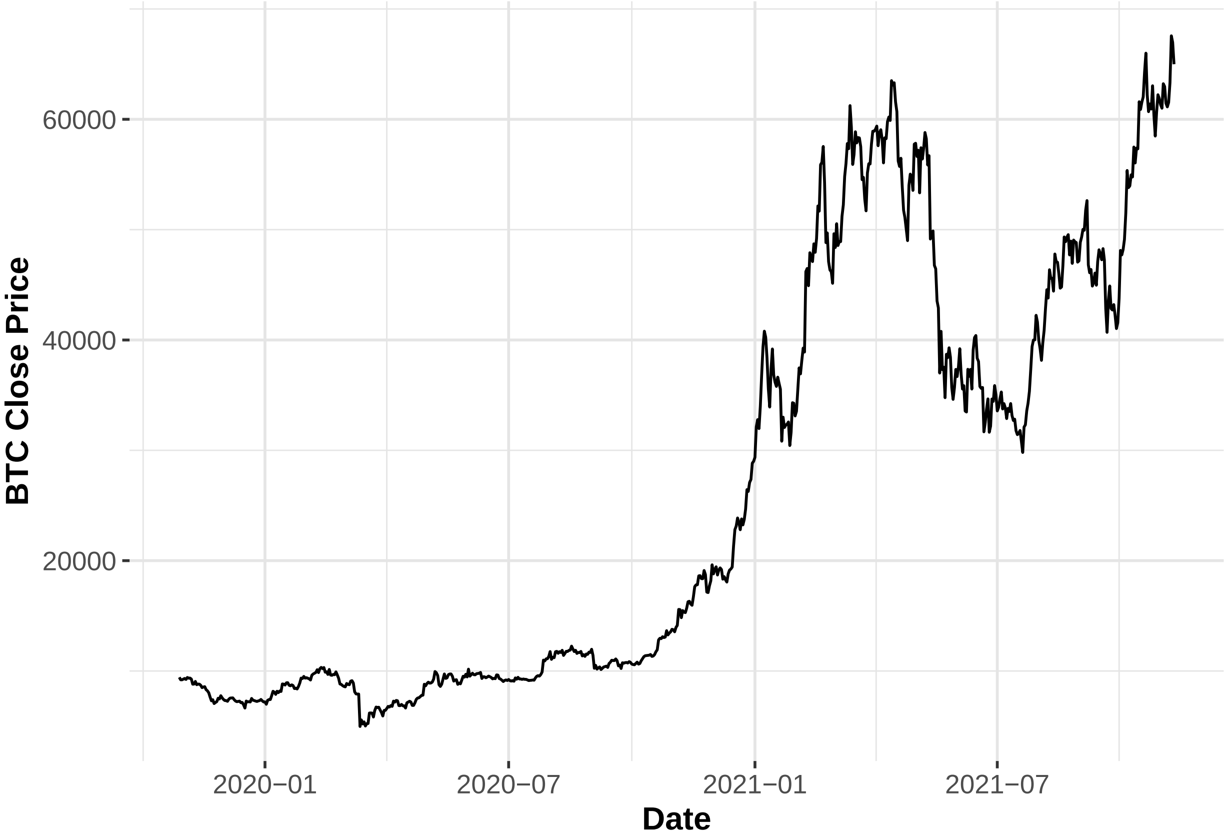
The first model our team will evaluate is the simple linear regression model.

Figure 1: Bitcoin Can we make money?

2 Dataset

Reading the bitcoin dataset. The dataset contains:

- {high,low,open,close}
- 14 indicators {volumne, macd_26, macd_9, macd_diff, adx, adx_neg, adx_pos, span_a, span_b, kijun, tenkan, rsi, cmf, sma}



2.1 BTC closing price history

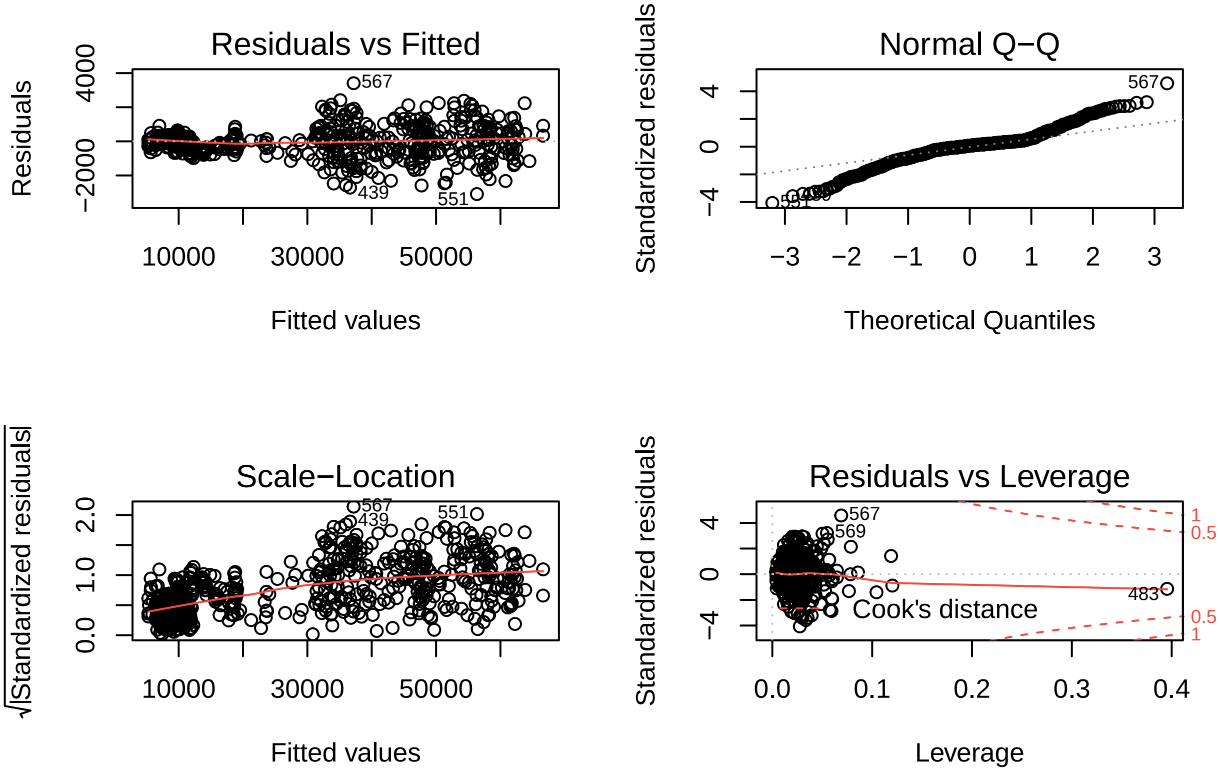
The plot is the historical closing price for BTC. The intent of the project is to determine if a model can be found and used to predict the closing price.

3 Linear Analysis Closing Price

Linear Analysis performed using linear regression (lm) on the closing BTC price and each of the 14 key indicators

```
## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :
##  extra argument ’formaula’ will be disregarded
```

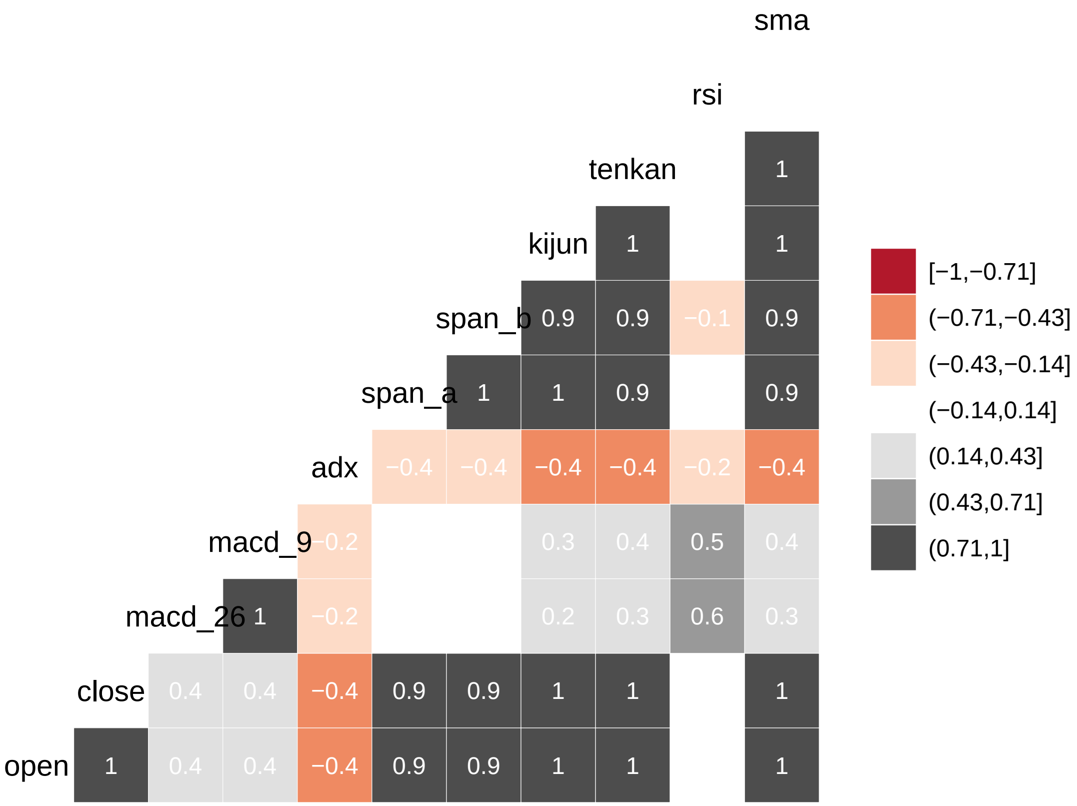
```
##
## Call:
## lm(data = Indicators, formaula = close ~ .)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3093.7  -319.3    54.0   271.4  3408.4
##
## Coefficients: (1 not defined because of singularities)
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.900e+03  6.076e+02  -3.127  0.00184 **
## close        3.200e-01  1.884e-02  16.989  < 2e-16 ***
## volume      -1.558e-09  1.647e-09  -0.946  0.34432
## macd_26      2.381e+00  1.157e-01  20.572  < 2e-16 ***
## macd_9      -2.844e+00  2.337e-01 -12.172  < 2e-16 ***
## macd_diff           NA         NA      NA       NA
## adx          1.150e+01  5.120e+00   2.247  0.02495 *
## adx_neg      1.707e+01  1.173e+01   1.456  0.14592
## adx_pos     -2.242e+01  1.443e+01  -1.553  0.12080
## span_a      -1.509e-01  3.250e-02  -4.644 4.06e-06 ***
## span_b      -4.351e-02  1.387e-02  -3.136  0.00178 **
## kijun        2.960e-01  3.227e-02   9.171  < 2e-16 ***
## tenkan       4.248e-01  5.726e-02   7.418 3.32e-13 ***
## rsi          3.291e+01  4.843e+00   6.796 2.25e-11 ***
## cmf         -4.358e+02  3.652e+02  -1.193  0.23317
## sma          1.648e-01  6.676e-02   2.468  0.01381 *
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 772.2 on 725 degrees of freedom
## Multiple R-squared:  0.9984, Adjusted R-squared:  0.9984
## F-statistic: 3.227e+04 on 14 and 725 DF,  p-value: < 2.2e-16
```

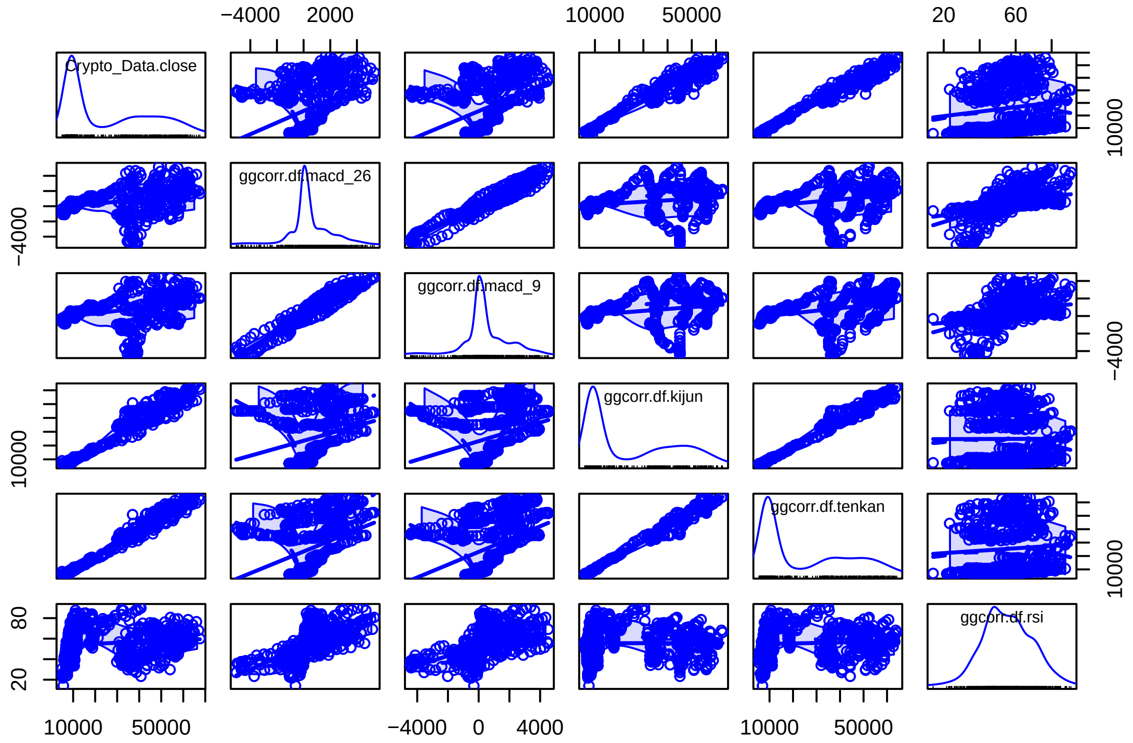


4 Linear Analysis

Given the p-values of the closing price of BTC, we may reject the null hypothesis for the following indicators.

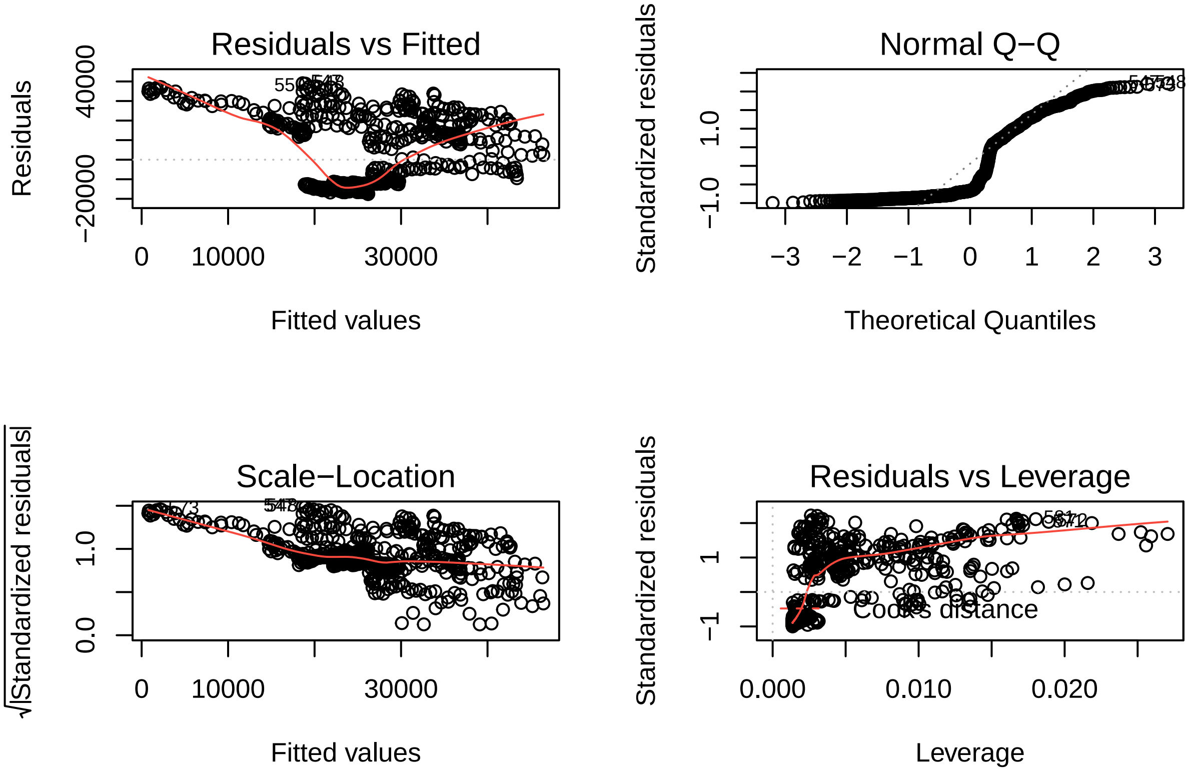
- volume (previous days trading volume)
- macd_26 (moving average convergence divergence, 26 day trend)
- macd_9 (moving average convergence divergence, 9 day trend)
- span_a (measures momentum)
- Kijun (midpoint price the last 26 periods)
- Tankan (japanese indicator)
- RSI (relative strength index)

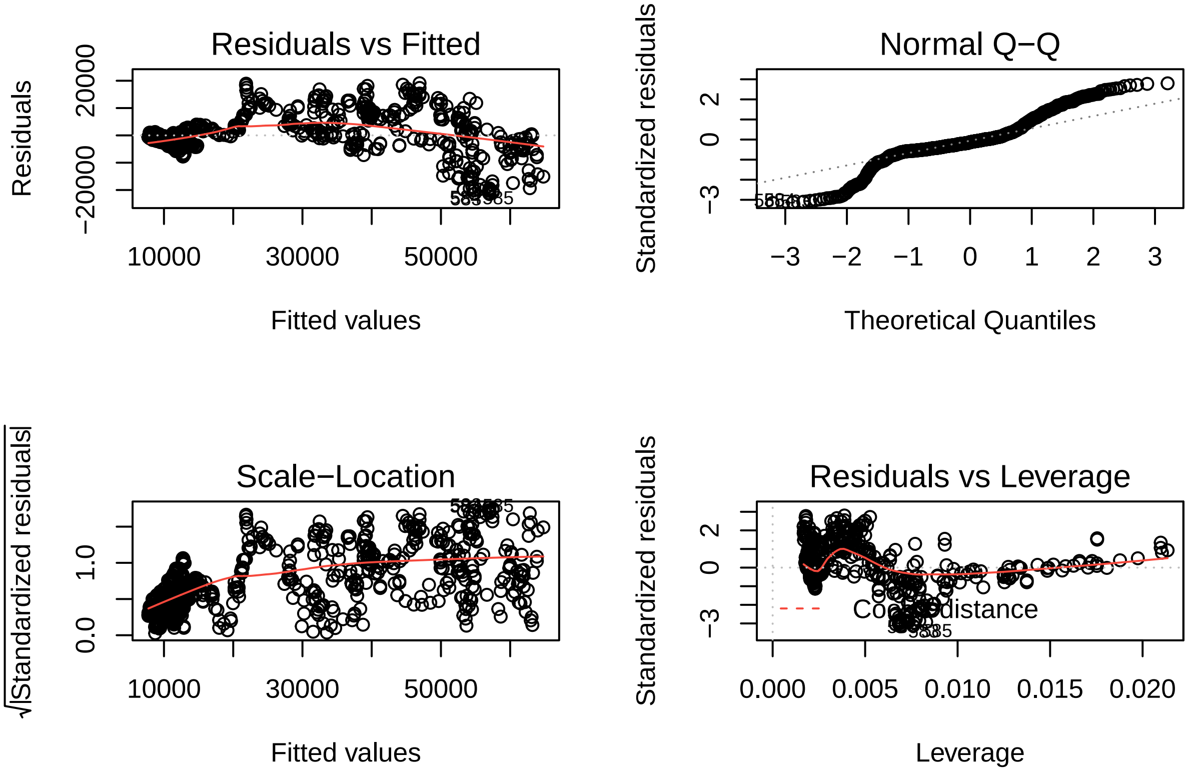

sma
rsi
tenkan 1
kijun 1 1
span_b 0.9 0.9 −0.1 0.9
span_a 1 1 0.9 0.9
adx −0.4 −0.4 −0.4 −0.4 −0.2 −0.4
macd_9 −0.2 0.3 0.4 0.5 0.4
macd_26 1 −0.2 0.2 0.3 0.6 0.3
close 0.4 0.4 −0.4 0.9 0.9 1 1 1
open 1 0.4 0.4 −0.4 0.9 0.9 1 1 1
[−1,−0.71]
(−0.71,−0.43]
(−0.43,−0.14]
(−0.14,0.14]
(0.14,0.43]
(0.43,0.71]
(0.71,1]

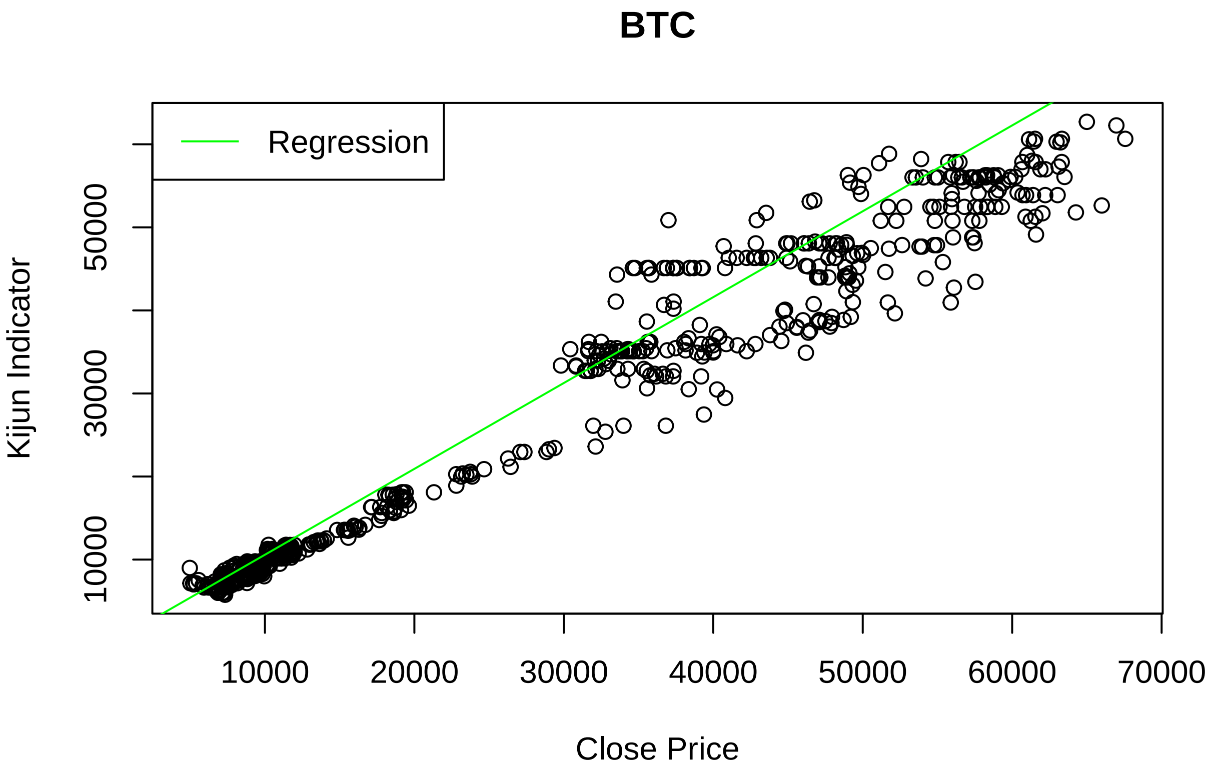


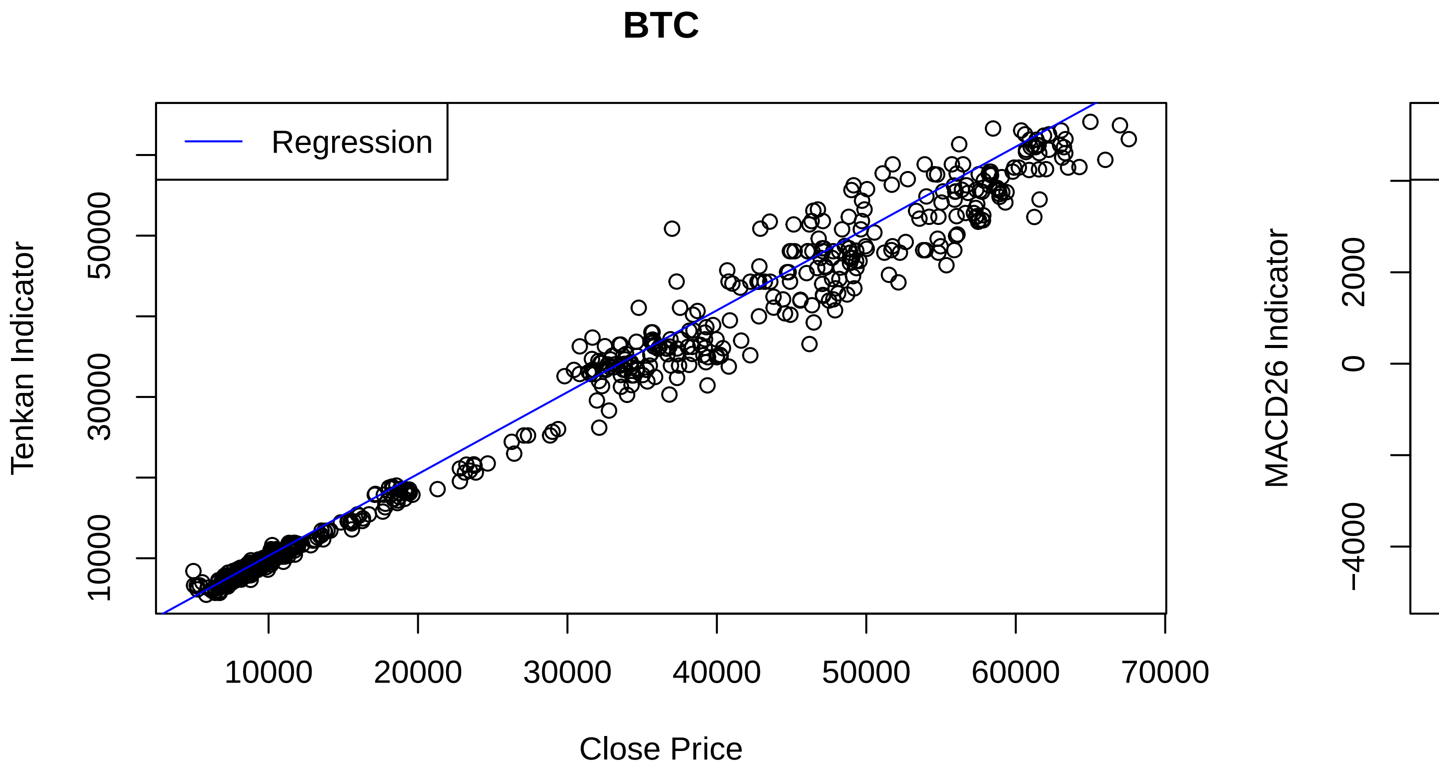
4.1 Correlation Analysis

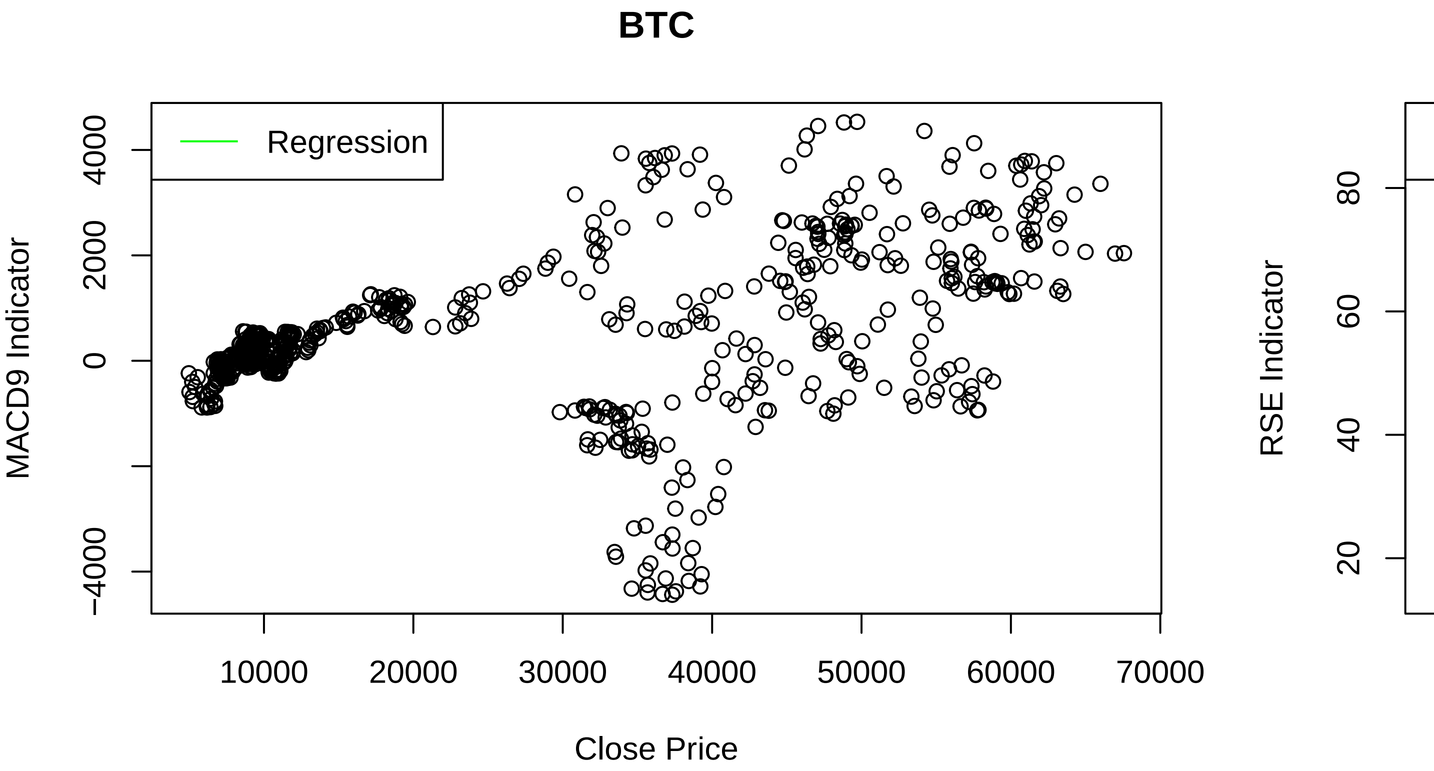
We are able to see the close price of BTC share a strong linear correlation with Kijun, Tankan and SMA indicators. Also with span_a and span_b but less so with macd_26 and macd_9. We can also see signs of high colinearity between diffrent predictor variables.

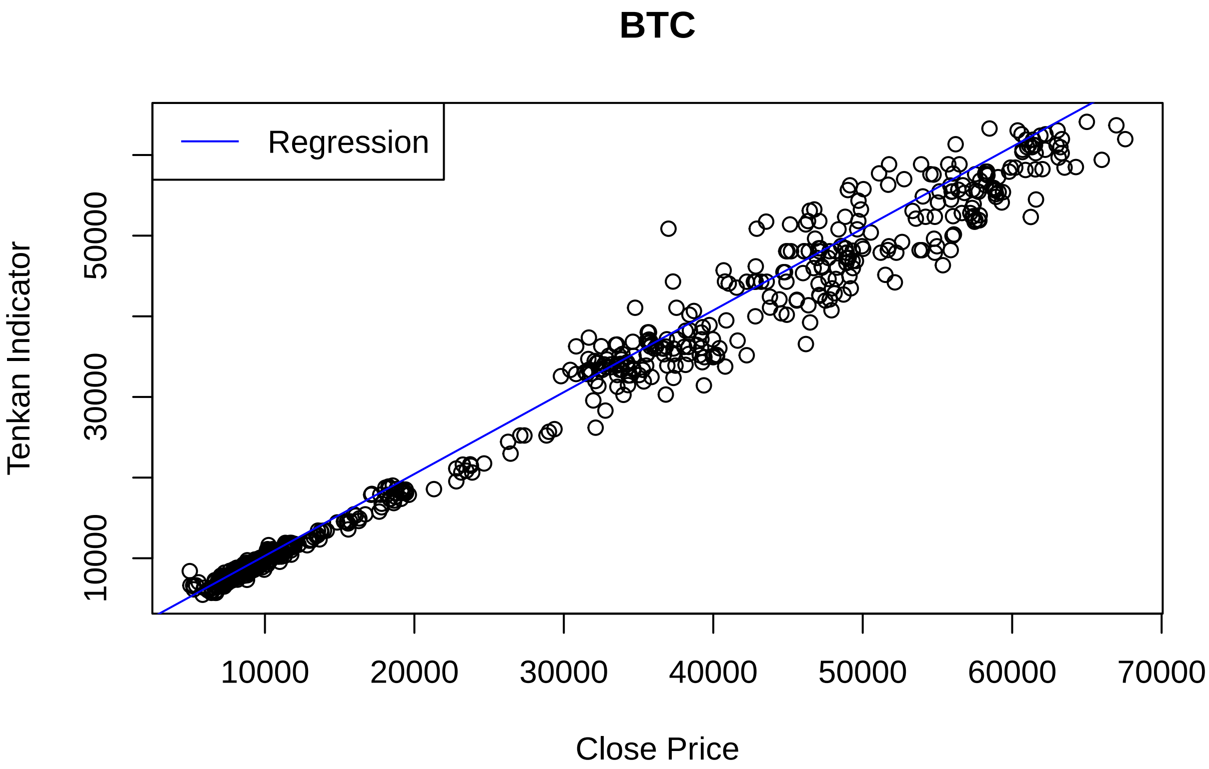

Residuals vs Fitted
Residuals
−20000
0
20000
40000
0
10000
30000
Fitted values
Normal Q−Q
Standardized residuals
−1.0
1.0
−3
−2
−1
0
1
2
3
Theoretical Quantiles
Scale−Location
√|Standardized residuals|
0.0
1.0
0
10000
30000
Fitted values
Residuals vs Leverage
Standardized residuals
−1
1
0.000
0.010
0.020
Leverage
Cook's distance

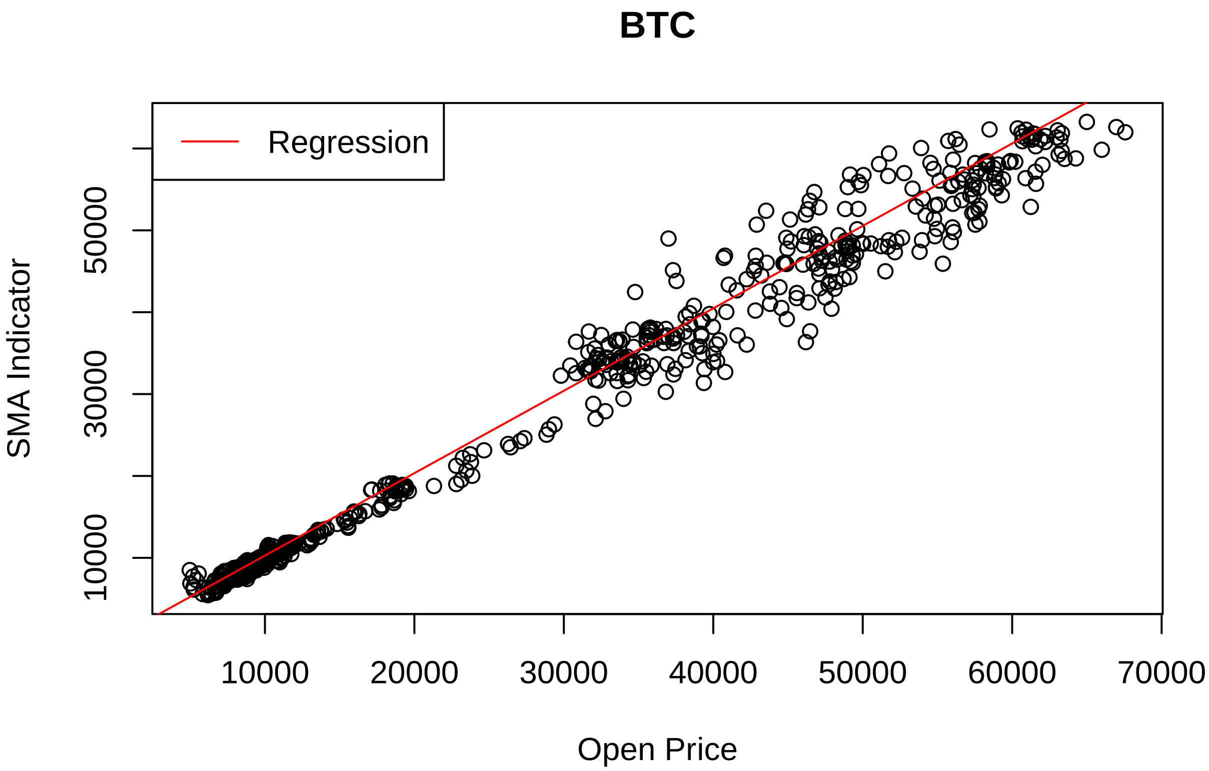

Residuals vs Fitted
Residuals
−20000
20000
10000
30000
50000
Fitted values
583
585
Normal Q−Q
Standardized residuals
−3
0
2
−3
−2
−1
0
1
2
3
Theoretical Quantiles
Scale−Location
√|Standardized residuals|
0.0
1.0
10000
30000
50000
Fitted values
Residuals vs Leverage
Standardized residuals
−3
0
2
Cook's distance
0.000
0.005
0.010
0.015
0.020
Leverage


BTC
Regression
Kijun Indicator
10000
30000
50000
Close Price
10000
20000
30000
40000
50000
60000
70000


BTC
Regression
Tenkan Indicator
10000
30000
50000
10000
20000
30000
40000
50000
60000
70000
Close Price
MACD26 Indicator
−4000
0
2000


BTC
Regression
MACD9 Indicator
4000
2000
0
−4000
Close Price
10000
20000
30000
40000
50000
60000
70000
RSE Indicator
80
60
40
20


BTC
Regression
Tenkan Indicator
10000
30000
50000
Close Price
10000
20000
30000
40000
50000
60000
70000



4.2 Analysis

Even though we can reject the null hypothesis for all the indicators above, the traditional linear regression model is not the best fit for most. It may be because the BTC prices are volatile, and we need a model that provides more flexibility.

```
## Warning: All formats failed to parse. No formats found.
```

5 Analysis

Using a simple linear model we were able to determine the following indicators from our linear regression analsysis and they will be used for a GAM model

- macd_26 (moving average convergence divergence, 26 day trend)
- macd_9 (moving average convergence divergence, 9 day trend)
- kijun (midpoint price the last 26 periods)
- tenkan (japans indicator)
- rsi (relative strength index)

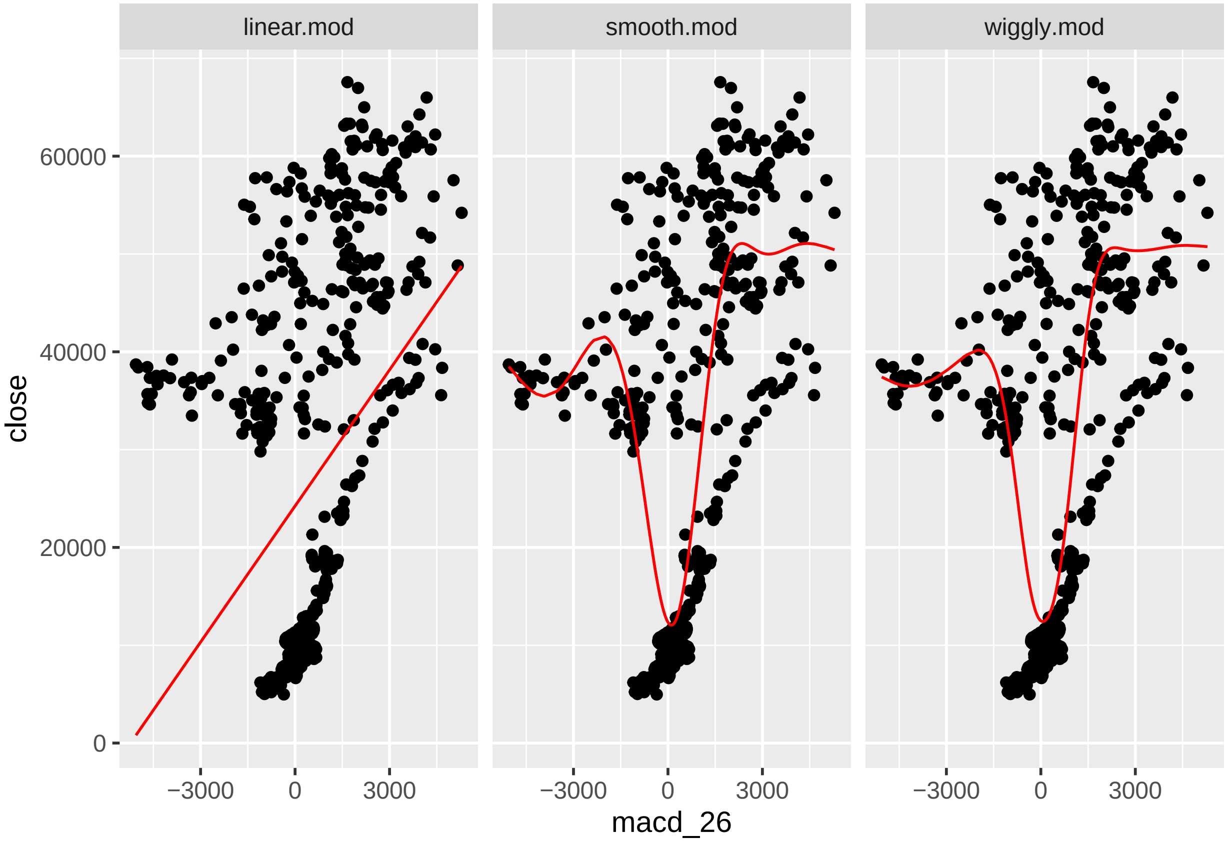
A dataframe was created the closing prices and the five listed key indicators

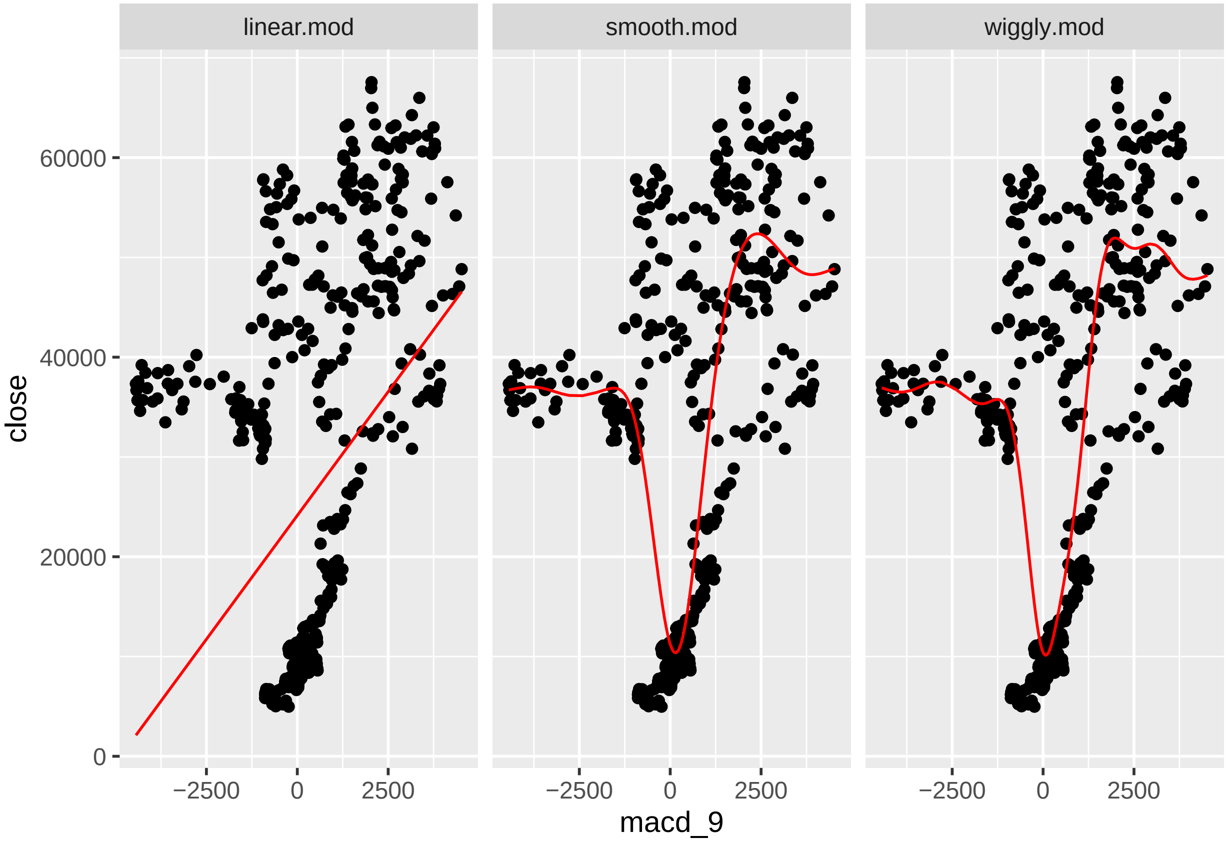
```
##
## Call:
## lm(formula = close ~ ., data = df3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8026.2  -547.0    23.4   497.9  7148.1
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.723e+03  3.463e+02  -4.976 8.08e-07 ***
## macd_26      3.681e+00  1.724e-01  21.350  < 2e-16 ***
## macd_9      -2.727e+00  1.547e-01 -17.629  < 2e-16 ***
## kijun        6.356e-01  5.340e-02  11.904  < 2e-16 ***
## tenkan       3.778e-01  5.374e-02   7.030 4.72e-12 ***
## rsi          3.186e+01  6.049e+00   5.267 1.82e-07 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 1558 on 734 degrees of freedom
## Multiple R-squared:  0.9934, Adjusted R-squared:  0.9934
## F-statistic: 2.218e+04 on 5 and 734 DF,  p-value: < 2.2e-16
```

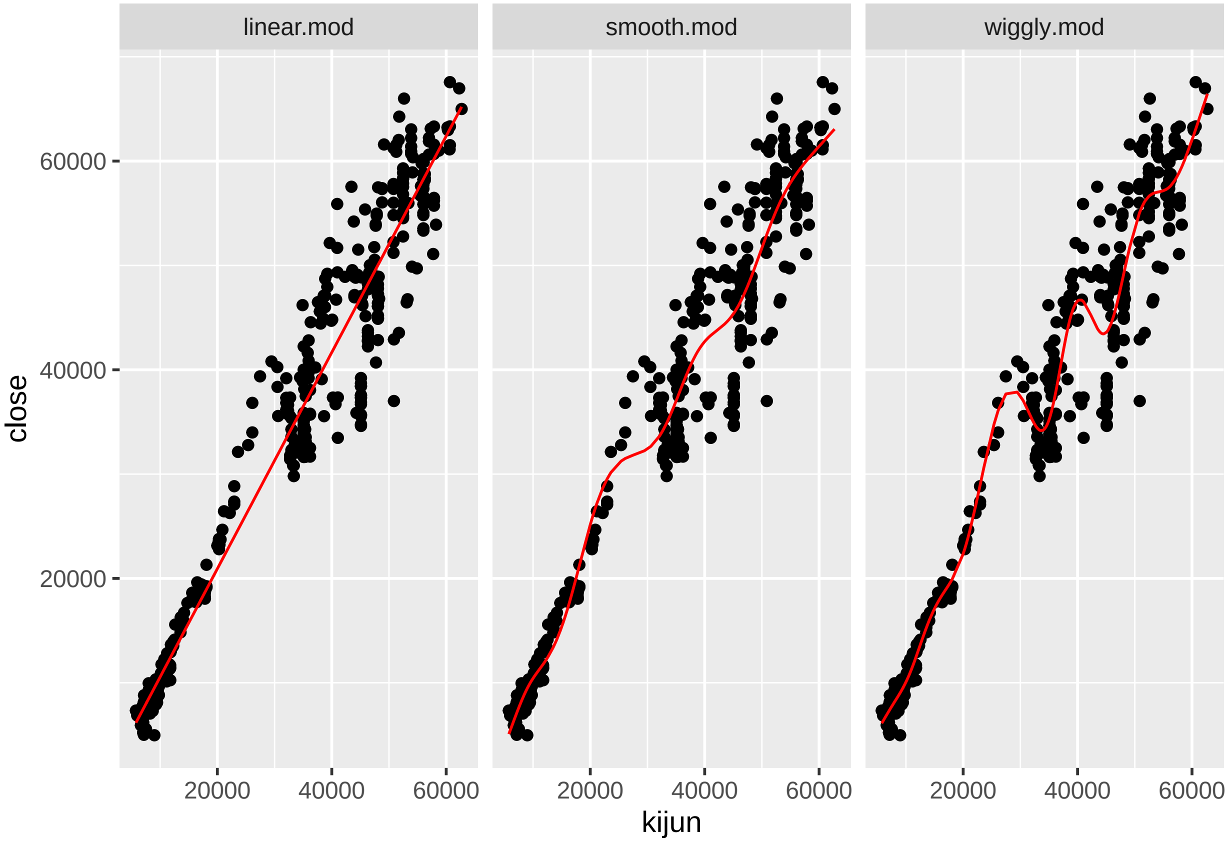
```
## Warning: package ’modelr’ was built under R version 4.1.1
```

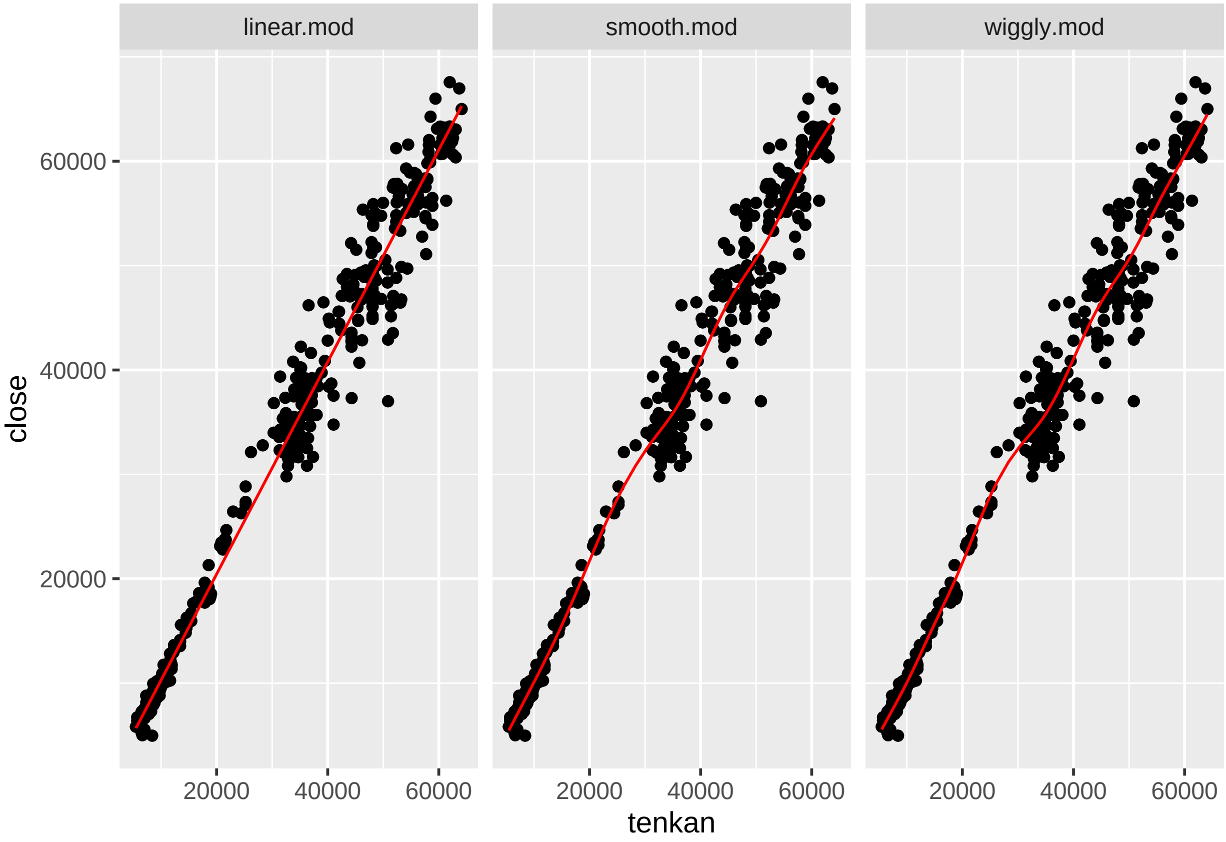
```
##
## Attaching package: ’modelr’
```

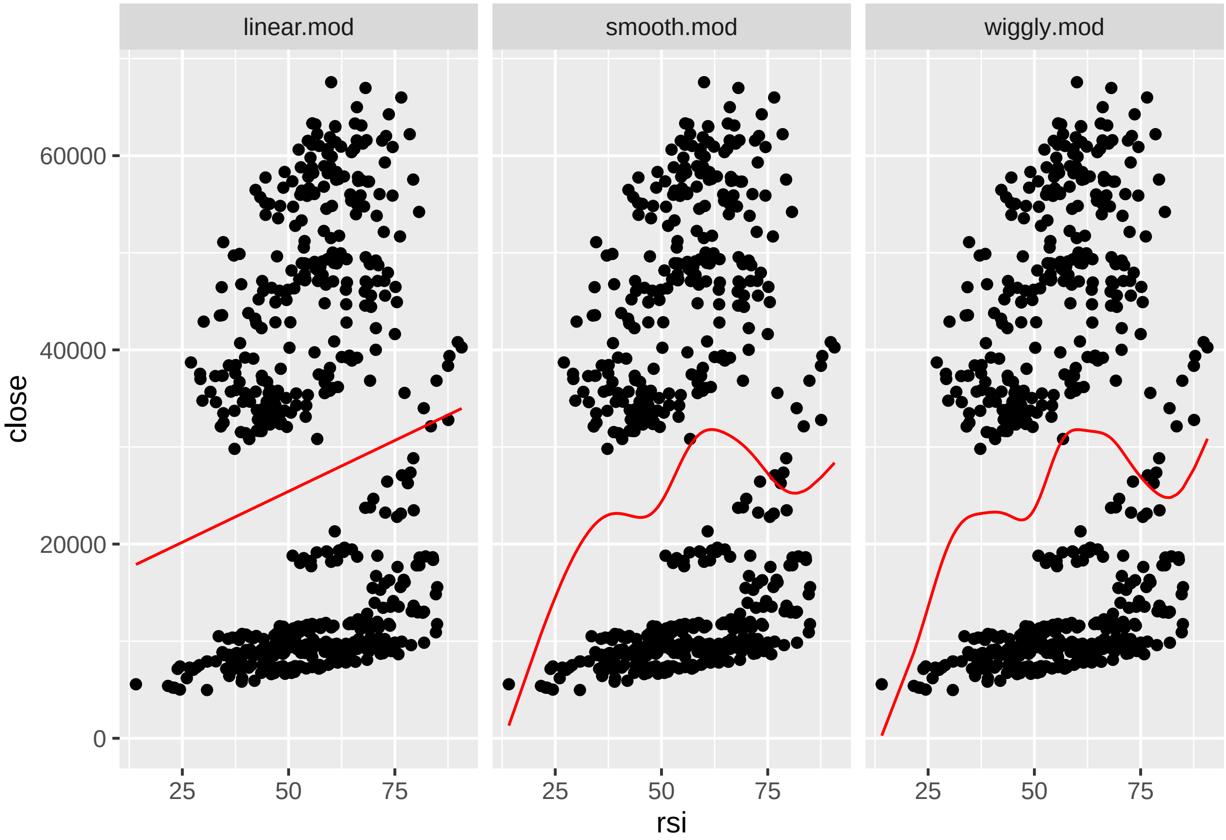
```
## The following object is masked from ’package:wrapr’:
##
##     qae
```


linear.mod
smooth.mod
wiggly.mod
close
60000
40000
20000
0
−3000
0
3000
−3000
0
3000
−3000
0
3000
macd_26


linear.mod
smooth.mod
wiggly.mod
close
0
20000
40000
60000
−2500
0
2500
macd_9


linear.mod
smooth.mod
wiggly.mod
close
60000
40000
20000
20000
40000
60000
kijun


linear.mod
smooth.mod
wiggly.mod
close
60000
40000
20000
20000
40000
60000
tenkan



5.1 GAM individual model fit

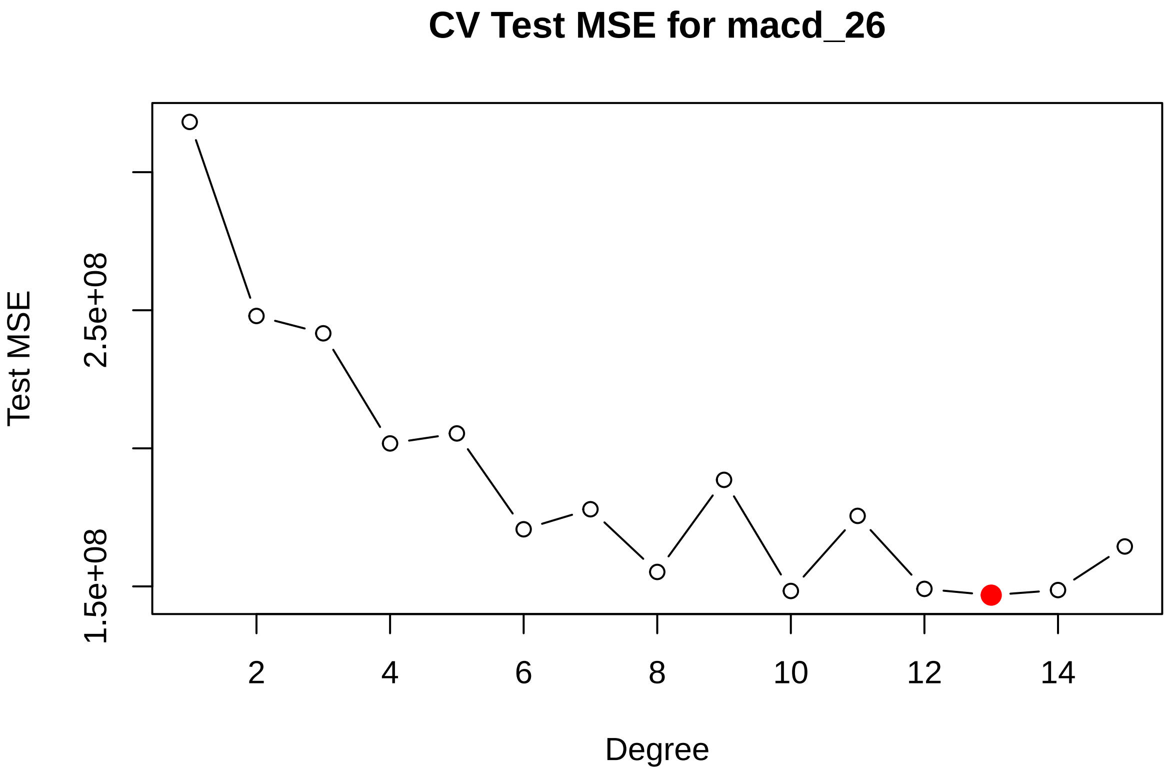
The GAM model was used to fit each of the 5 individual key indicators. As seen, we can not use them individually to get a great model of closing prices. They all need to be used in conjunction to create the best fitting model.

6 Cross Validation

A cross-validation (CV) will be performed on the five indicators to determine the best K value to use in our GAM model.

```
##
## Call: gam(formula = close ~ s(macd_26), data = train)
## Deviance Residuals:
##    Min     1Q Median     3Q    Max
## -18829  -8647  -6362   7755  42696
##
## (Dispersion Parameter for gaussian family taken to be 170237579)
##
##     Null Deviance: 219385883243 on 591 degrees of freedom
## Residual Deviance: 99929501232 on 587.0002 degrees of freedom
## AIC: 12907
##
## Number of Local Scoring Iterations: NA
##
## Anova for Parametric Effects
##             Df    Sum Sq    Mean Sq F value    Pr(>F)
## s(macd_26)   1 3.213e+10 3.2130e+10  188.73 < 2.2e-16 ***
## Residuals  587 9.993e+10 1.7024e+08
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Anova for Nonparametric Effects
##             Npar Df Npar F     Pr(F)
## (Intercept)
## s(macd_26)        3    171 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

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

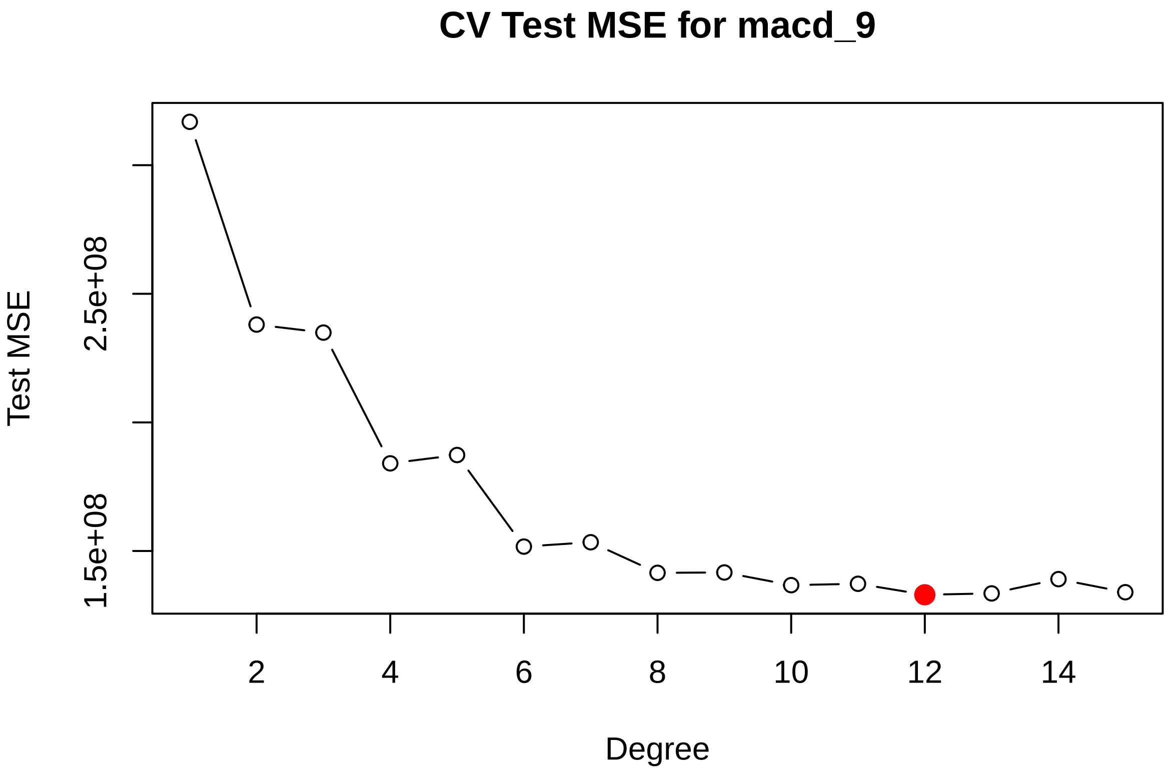


6.1 Cross Validation

CV best K = 12 for the macd_26 indicator

```
##
## Call: gam(formula = close ~ s(macd_9), data = train)
## Deviance Residuals:
##    Min     1Q Median     3Q    Max
## -21381  -7862  -5702   6651  41576
##
## (Dispersion Parameter for gaussian family taken to be 157839826)
##
##     Null Deviance: 219385883243 on 591 degrees of freedom
## Residual Deviance: 92652025824 on 587.0003 degrees of freedom
## AIC: 12862.24
##
## Number of Local Scoring Iterations: NA
##
## Anova for Parametric Effects
##            Df     Sum Sq    Mean Sq F value    Pr(>F)
## s(macd_9)   1 3.3159e+10 3.3159e+10  210.08 < 2.2e-16 ***
## Residuals 587 9.2652e+10 1.5784e+08
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Anova for Nonparametric Effects
##             Npar Df Npar F     Pr(F)
## (Intercept)
## s(macd_9)         3 197.64 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## [1] 12
```

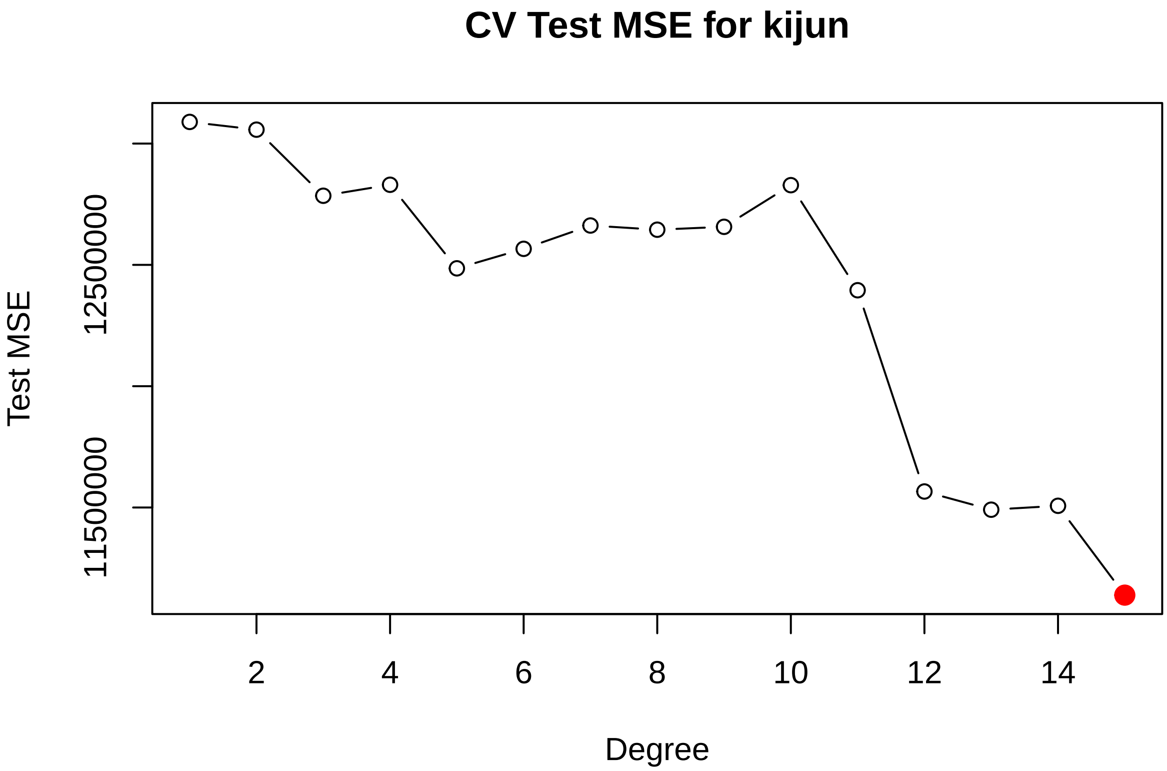


6.2 Cross Validation

CV best K = 12 for the macd_9 indicator

```
##
## Call: gam(formula = close ~ s(kijun), data = train)
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -15568.9  -1086.8     59.9    988.8  13354.5
##
## (Dispersion Parameter for gaussian family taken to be 12406008)
##
##     Null Deviance: 219385883243 on 591 degrees of freedom
## Residual Deviance: 7282325868 on 586.9999 degrees of freedom
## AIC: 11356.55
##
## Number of Local Scoring Iterations: NA
##
## Anova for Parametric Effects
##            Df     Sum Sq    Mean Sq F value    Pr(>F)
## s(kijun)    1 2.1167e+11 2.1167e+11   17062 < 2.2e-16 ***
## Residuals 587 7.2823e+09 1.2406e+07
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Anova for Nonparametric Effects
##             Npar Df Npar F     Pr(F)
## (Intercept)
## s(kijun)          3 11.578 2.212e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## [1] 15
```

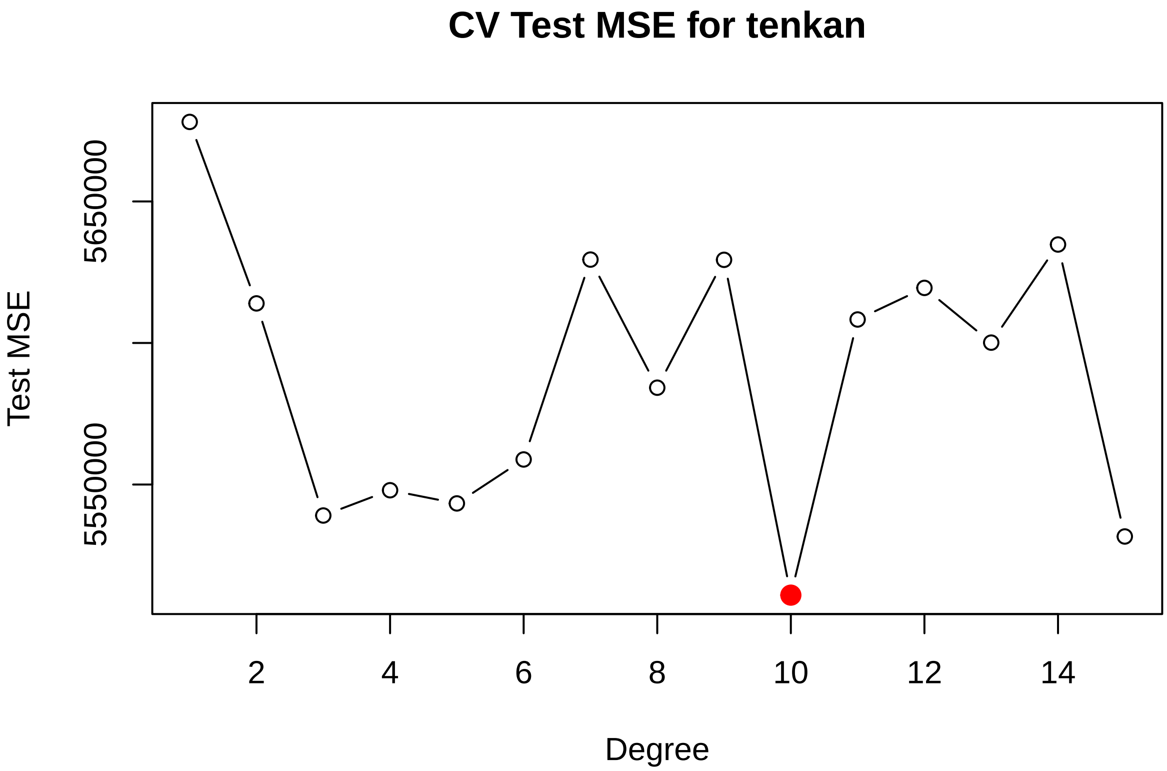


6.3 Cross Validation

CV best K = 15 for the kijun indicator

```
##
## Call: gam(formula = close ~ s(tenkan), data = train)
## Deviance Residuals:
##       Min        1Q    Median        3Q       Max
## -14678.05   -593.45     22.39    622.87   8520.26
##
## (Dispersion Parameter for gaussian family taken to be 5475475)
##
##     Null Deviance: 219385883243 on 591 degrees of freedom
## Residual Deviance: 3214103478 on 586.9999 degrees of freedom
## AIC: 10872.35
##
## Number of Local Scoring Iterations: NA
##
## Anova for Parametric Effects
##            Df     Sum Sq    Mean Sq F value    Pr(>F)
## s(tenkan)   1 2.1607e+11 2.1607e+11   39461 < 2.2e-16 ***
## Residuals 587 3.2141e+09 5.4755e+06
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Anova for Nonparametric Effects
##             Npar Df Npar F     Pr(F)
## (Intercept)
## s(tenkan)         3 6.2735 0.0003401 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## [1] 10
```

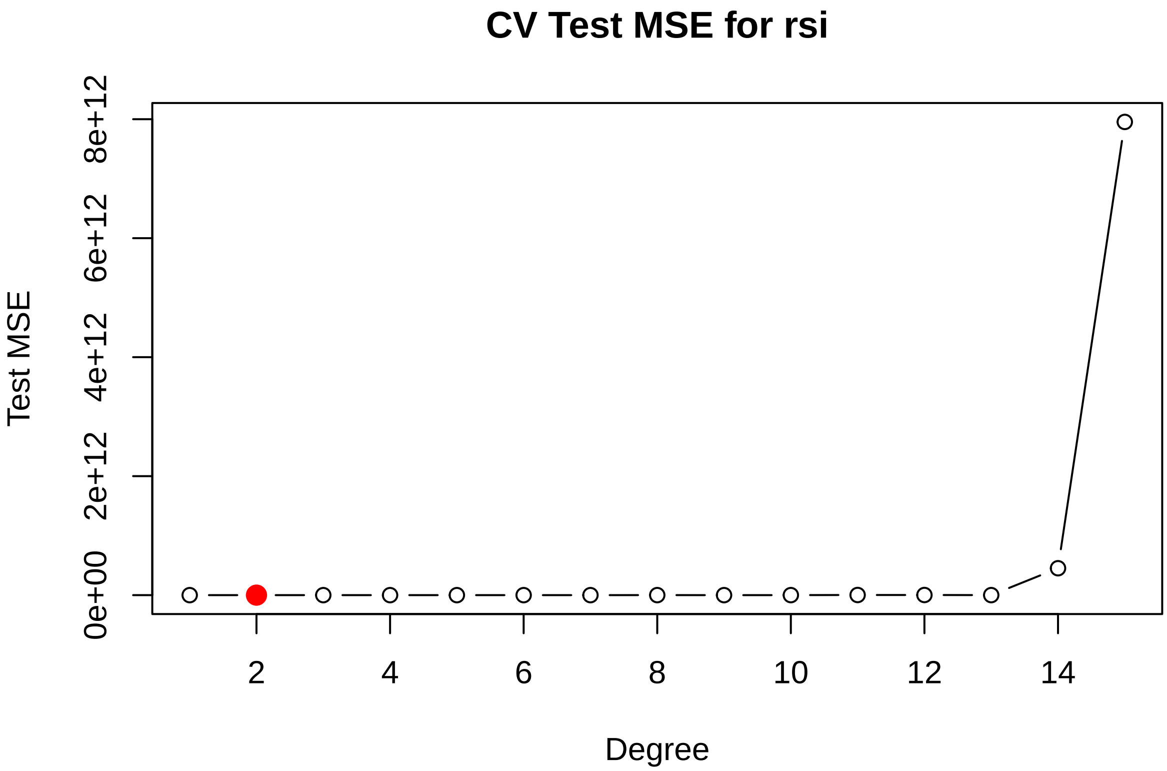


6.4 Cross Validation

CV best K = 10 for the tenkan indicator

```
##
## Call: gam(formula = close ~ s(rsi), data = train)
## Deviance Residuals:
##    Min     1Q Median     3Q    Max
## -23174 -16425 -10042  17634  38387
##
## (Dispersion Parameter for gaussian family taken to be 355869701)
##
##     Null Deviance: 219385883243 on 591 degrees of freedom
## Residual Deviance: 208895483694 on 586.9999 degrees of freedom
## AIC: 13343.53
##
## Number of Local Scoring Iterations: NA
##
## Anova for Parametric Effects
##            Df     Sum Sq    Mean Sq F value    Pr(>F)
## s(rsi)      1 4.8622e+09 4862236091  13.663 0.0002392 ***
## Residuals 587 2.0890e+11  355869701
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Anova for Nonparametric Effects
##             Npar Df Npar F    Pr(F)
## (Intercept)
## s(rsi)            3 5.2716 0.001357 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

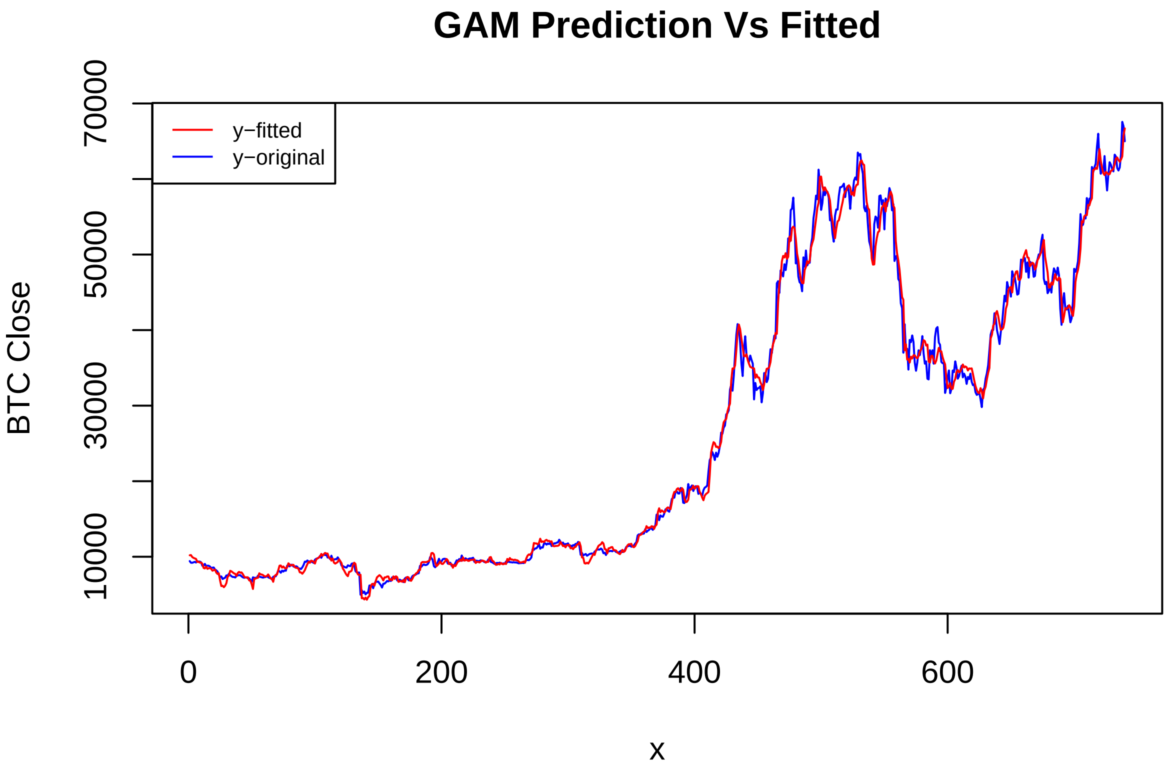
```
## [1] 2
```



6.5 Cross Validation

CV best K=2 for the rsi indicator

```
##
## Call: gam(formula = close ~ ns(macd_26, 13) + ns(macd_9, 12) + ns(kijun,
##     15) + ns(tenkan, 10) + ns(rsi, 2), family = gaussian, data = train)
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -7023.02  -555.73   -21.33   449.85  6696.05
##
## (Dispersion Parameter for gaussian family taken to be 2042279)
##
##     Null Deviance: 219385883243 on 591 degrees of freedom
## Residual Deviance: 1100788643 on 539 degrees of freedom
## AIC: 10334.01
##
## Number of Local Scoring Iterations: 2
##
## Anova for Parametric Effects
##                  Df     Sum Sq    Mean Sq   F value    Pr(>F)
## ns(macd_26, 13)  13 1.3622e+11 1.0479e+10 5130.8275 < 2.2e-16 ***
## ns(macd_9, 12)   12 1.3860e+10 1.1550e+09  565.5277 < 2.2e-16 ***
## ns(kijun, 15)    15 6.8123e+10 4.5415e+09 2223.7446 < 2.2e-16 ***
## ns(tenkan, 10)   10 4.8135e+07 4.8135e+06    2.3569 0.0098991 **
## ns(rsi, 2)        2 3.3163e+07 1.6582e+07    8.1192 0.0003357 ***
## Residuals       539 1.1008e+09 2.0423e+06
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

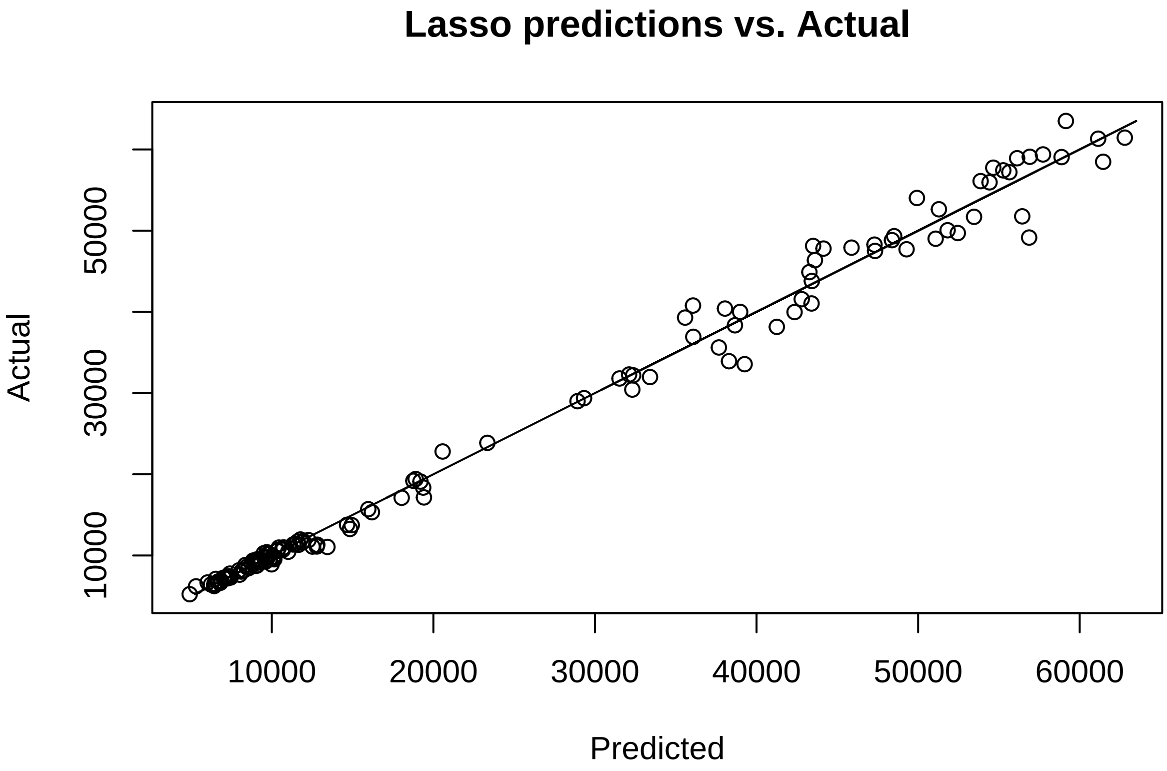


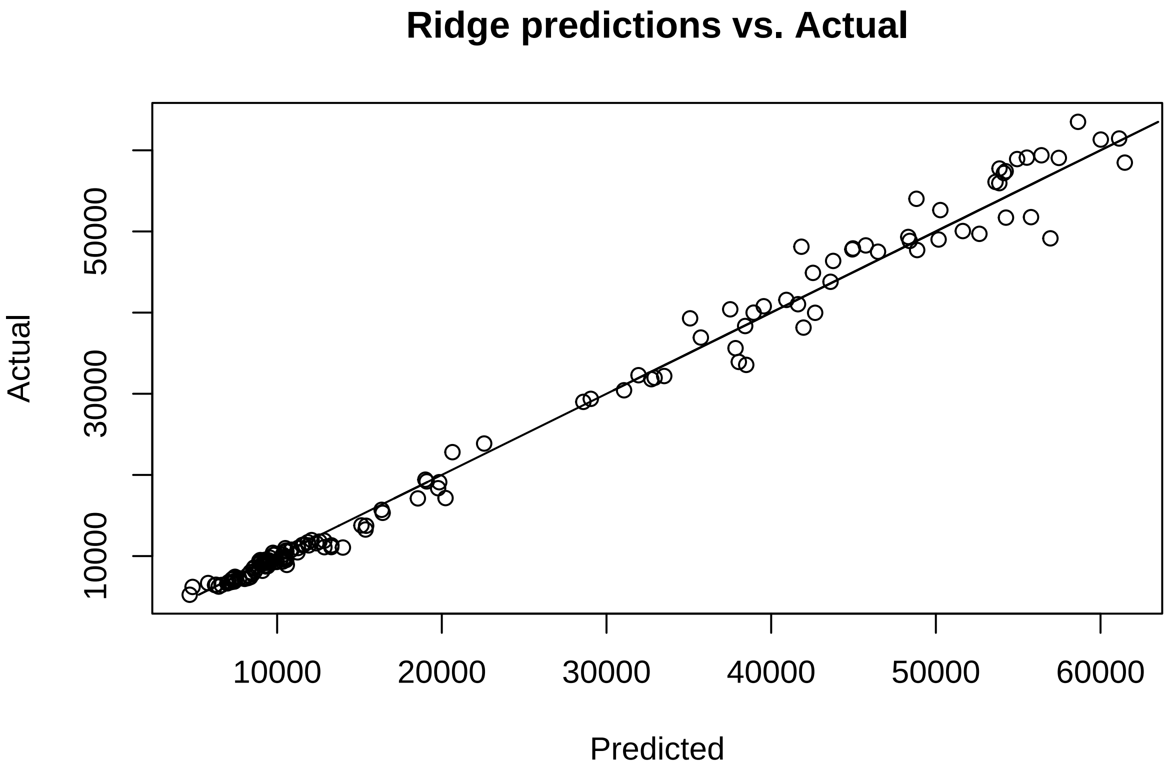
7 GAM actual vs. predicted Closing Price

This plot overlays the fitted GAM model vs. Actual BTC closing price. This is a very good fit and a suprisingly accurate model to predict BTC closing price from the previous day five indicators.

8 Lasso and Ridge Models

We are going to build a lasso and ridge regression model using all predictor variables in order to predict the close price. We will also determine which predictors went to zero using the lasso model and we will compare the test MSE for each model to determine which model performs the best. Both models will be tested using cross validation with ten folds.





9 Model Evaluation

The lasso model determined the predictor variables macd_9, span_a, span_b, and the sma all had a coeffienct of zero which indicates these variables are not important when trying to predict the close price of bitcoin. The test MSE for the lasso model is 2.7553602×10^6 which is quite a bit better than the test MSE of the ridge model of 3.4175114×10^6. Overall a lasso model is preferred to a ridge model when predicting the close price of bitcoin.

10 Calculating metrics

Our team will now compare test MSE's across all four models.

11 Model Evaluation

After comparing test MSE across the four models we have the following test MSE scores:

- Linear Regression: 2.8149251×10^6
- GAM: 2.4589802×10^6
- Lasso: 2.7553602×10^6
- Ridge: 3.4175114×10^6

So we can conclude that the best model to use for predicting bitcoin close price is the GAM model which used natural splines at the optimal degree which was found by using cross validation techniques.

We also were able to discover the coefficients for most important predictor variables that can be used to successfully predict the bitcoin close price which are the following:

- GAM:
  - macd_26 with a natural spline of 13 degrees
  - macd_9 with a natural spline of 12 degrees
  - kijun with a natural spline of 15 degrees
  - tenkan with a natural spline of 10 degrees
  - rsi with a natural spline of 2 degrees

12 Conclusion

Our team has evaluated four different models that can be used to predict the closing price of bitcoin. After looking at the test mse scores of the four models we have concluded that the GAM model which used natural splines was able to report a MSE of 2.4589802×10^6 a MAE of 969.4837132 and an r-squared of 0.9916912. Overall all our team was excited about our discovery and really enjoyed the process of building a statistical model to help predict the close price of bitcoin.

12.1 Group Participation:

Laura: Laura worked on the linear regression portion of this report, including the plots and description of results. Shawn: Shawn worked on the GAM model for this report, including the plots and analysis of the model. Justin: Justin worked on the ridge regression model for this report, plots and interpretation. Josh: Josh worked on the lasso regression model for this report, including the interpretation of results and plots.

The entire team worked collaboratively on the data collection/cleansing portion of the project, as well as the Cross Validation for the GAM model, and the overall formatting, and comparison of all models considered in this report.

12.2 Disclaimer

We are aware that we violated the assumptions that our observations are not independent. Be aware these models need to be taken with a grain of salt and to not be used for day trading.

ENDNOTES

TABLE OF CONTENTS