IZMIR UNIVERSITY OF ECONOMICS
FACULTY OF ENGINEERING
COMPUTER ENGINEERING

CE 475 PROJECT REPORT

İZMİR EKONOMİ ÜNİVERSİTESİ

Author: Atahan Ekici

Number: 2015 060 2015

CONTENTS

METHODOLOGY

In this project we were asked to predict 20 Y values with using 100 x1 to x6 values so in order to accomplish this goal I have used various libraries and techniques such as K-Fold-CV and sklearn library functions to overcome this problem. The methods I used to achieve this goal is:

- Linear Regression
- Polynomial Regression
- Ridge Regression
- Lasso Regression
- Random Forest
- Gradient Boosting
- XGB Regression
- Light GBM Regression
- Cat Boost
- K Nearest Neighbor (KNN)
- Decision Tree

Pros

- For Linear Models such as Linear Regression, Polynomial Regression, Lasso and Ridge Regression; I would have chosen those even if we were not permitted to use libraries since Linear Models are easy to implement because of their simplicity.
- Since they are simple to implement they are easy to compute as well so this results a minimal waiting time compared to more complex models such as Gradient Boosting or Decision Tree since they are prone to iterate over the data more than 1 time.

- For Random Forest Model, it is a more complex version of the Decision Tree so it can be more accurate than most of the models and it does not require any normalization to work.

- For Gradient Boosting, it is an ensemble method that does not need any pre-processing of the data.

- For XGB Regression, this method is an ensemble method and implemented using decision trees with boosted gradient for enhanced accuracy.

- For Light GBM Regression, is a boosting model so it uses a histogram-based algorithm to “boost” its efficiency and since it replaces continuous values to discrete bins results a lower memory usage than most models.

- For Cat Boost, is another boosting model that does not require conversion of data set to any specific format.

- For K Nearest Neighbor, is a model that does not require any training period in order to function properly and it can be easily implemented since the model only requires to calculate distances between different points and as a result any data can be added instantly without any conversion.

- For Decision Tree, is a model that does not require any normalization or scaling of the data.

Cons

- For Linear Models, these models are sensitive to outliers and prone to underfitting

- For Random Forest, this model is hard to adjust variables through better fitting of the model, parameter complexity is high and overfitting is a risk for this model.

- For Gradient Boosting, since this model continue improving itself to minimize all errors this behavior causes it to be prone to overfitting so this model is almost always supposed to dependent for a cross validation technique in order to neutralize overfitting.
- This model is a highly complex model so the computation times are usually over the roof.

- For XGB Regression, this model is a highly sophisticated model and since it uses tree-based algorithms for prediction it falls short when it comes to tasks that involves extrapolation.

- For Light GBM Regression, is a boosting model so like all the boosting models this one is not different when it comes to overfitting.

- For Cat Boost, this model is combining both Decision Trees and Gradient Boosting methods together and since they are 2 complex models when combined resulting a slow computational performance like the Decision Tree and prone to overfitting like the Gradient Boost.

- For K Nearest Neighbor, this model does not work very well under large dataset since calculating distances between every node can be costly and this model can be very sensitive towards noisy or missing data.

- For Decision Tree, this model is a tree-based model and sometimes a small change in the training data can be cause this model to collapse and can lead to instability.
- Training this model is costly and its complexity grows geometrically.

IMPLEMENTATION

Coding Environment

For implementation I have used PyCharm Community Edition 2020.3 as the Integrated Development Environment with accompanied by Python 3.9.0.


```
C:\Windows\system32>python --version
Python 3.9.0
```

Libraries Used

```
# Sklearn Imports #
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import KFold

from sklearn.pipeline import make_pipeline

from sklearn.tree import DecisionTreeRegressor

from sklearn.neighbors import KNeighborsRegressor

from sklearn.linear_model import LinearRegression
from sklearn.linear_model import LogisticRegression
from sklearn.linear_model import Ridge
from sklearn.linear_model import Lasso

from sklearn.preprocessing import PolynomialFeatures
from sklearn.preprocessing import StandardScaler

from sklearn.ensemble import GradientBoostingRegressor
from sklearn.ensemble import RandomForestRegressor
# Sklearn Imports #
```

```
# Other libraries #
from xgboost import XGBRegressor

from lightgbm import LGBMRegressor

from catboost import CatBoostRegressor
# Other libraries #
```

```
import csv
import numpy as np
import matplotlib.pyplot as plt
```

RESULTS

1) Linear Regression

```
Linear Model Predictions:
 [4431.91858524 1013.30332245 -773.11440542 6367.78100717 3742.45548917
 2484.77899205 3266.30925745 2467.11761955 2461.42804775 4679.68457569
  510.09772027  960.62107084  391.5551055   608.78575876  329.54799611
 5028.17192301  160.09695284 2321.30060282 1717.89755071 3151.92954987]

Linear Model Score: 0.5445025542552582
Linear K-Fold CV Scores: [0.61568525 0.56583517 0.34791114 0.55931669 0.27119326]
Linear K-Fold MSE Scores: [-2739430.60473244 -4817824.33748552 -5016093.48952495 -3846310.19714958
 -2613471.01675035]
Linear K-Fold CV Score Average:  0.4719883005523207
Linear K-Fold CV N-MSE Score Average:  -3806625.9291285677
```

Non-Validated Score: 0.5445025542552582

K-Fold CV Scores: [0.61568525 0.56583517 0.34791114 0.55931669 0.27119326]

K-Fold CV Score Average: 0.4719883005523207

K-Fold N-MSE Scores: [-2739430.60473244 -4817824.33748552 -5016093.48952495 -3846310.19714958 -2613471.01675035]

K-Fold CV N-MSE Score Average: -3806625.9291285677

Predictions:

[4431.91858524 1013.30332245 -773.11440542 6367.78100717 3742.45548917

2484.77899205 3266.30925745 2467.11761955 2461.42804775 4679.68457569

510.09772027 960.62107084 391.5551055 608.78575876 329.54799611

5028.17192301 160.09695284 2321.30060282 1717.89755071 3151.92954987]

- Since this model is a linear based model I did not have much hope for it to be the optimal solution for this data set but it was the first model we have been introduced in this lecture so I have featured this model in my project out of respect.

- With a Non-Validated Score of **0.54** and a Validated Score of **0.47** this model is consistent but far from optimal so I'm not choosing this model for submission.

2) Polynomial Regression

```
Polynomial Model Predictions:
 [4051.13290479 -179.76179119  210.16434264 8732.9495304  2259.99363568
 1649.29766924 3189.96238066 3665.11883082 1553.68228464 6199.01660085
  392.65932787 -185.95893427 1061.56473183  746.003454   2055.6858206
 3775.86014784 1408.84312535 2052.23124113   89.66937274 4064.01431171]

Polynomial Model Score: 0.8181607985449979
Linear K-Fold CV Scores: [0.60693507 0.79249228 0.52035467 0.74397897 0.5154487 ]
Polynomial K-Fold MSE Scores: [-2801802.6698021  -2302664.05450101 -3689598.09865423 -2234566.82366566
 -1737581.04828823]
Polynomial K-Fold CV N-MSE Score Average:  -2553242.538982247
Polynomial CV Score Average:  0.6358419369840121
```

Non-Validated Score: 0.8181607985449979

K-Fold CV Scores: [0.60693507 0.79249228 0.52035467 0.74397897 0.5154487]

K-Fold CV Score Average: 0.6358419369840121

K-Fold N-MSE Scores: [-2801802.6698021 -2302664.05450101 -3689598.09865423 -2234566.82366566 -1737581.04828823]

K-Fold CV N-MSE Score Average: -2553242.538982247

Model Predictions:

[4051.13290479 -179.76179119 210.16434264 8732.9495304 2259.99363568

1649.29766924 3189.96238066 3665.11883082 1553.68228464 6199.01660085

392.65932787 -185.95893427 1061.56473183 746.003454 2055.6858206

3775.86014784 1408.84312535 2052.23124113 89.66937274 4064.01431171]

- Since this model is another linear model I knew this model was not going to be the most optimized for this data set yet again this linear model has the best scores out of all other linear based models.

- With a Non-Validated Score of **0.81** and a Validated Score of **0.63** this model is not optimal considering the upcoming models so I'm not choosing this model for submission.

3) Ridge Regression

```
Ridge Model K-Fold CV Scores: [0.61567449 0.56581627 0.34793139 0.55932319 0.27121366]
Ridge Model K-Fold MSE Scores: [-2739507.27799502 -4818034.06049002 -5015937.67535353 -3846253.45898889
 -2613397.86465838]
Ridge Model Model CV Score Average:  0.4719918005570004
Ridge Model N-MSE Score Average:  -3806626.067497169
Ridge Model Predictions :
 [4431.72238338 1013.36795997 -772.98591744 6367.62653838 3742.3277131
 2484.92579777 3266.3520396  2467.19167996 2461.56287502 4679.58265485
  510.22589723  960.83028593  391.64896503  608.89229759  329.67226803
 5028.11059103  160.3116025  2321.21889168 1717.78004085 3151.90672049]

Ridge Model Score: 0.5445025522391778
```

Non-Validated Score: 0.5445025522391778

K-Fold CV Scores: [0.61567449 0.56581627 0.34793139 0.55932319 0.27121366]

K-Fold CV Score Average: 0.4719918005570004

K-Fold N-MSE Scores: [-2739507.27799502 -4818034.06049002 -5015937.67535353 -3846253.45898889 -2613397.86465838]

K-Fold CV N-MSE Score Average: -3806626.067497169

Model Predictions:

[4431.72238338 1013.36795997 -772.98591744 6367.62653838 3742.3277131

2484.92579777 3266.3520396 2467.19167996 2461.56287502 4679.58265485

510.22589723 960.83028593 391.64896503 608.89229759 329.67226803

5028.11059103 160.3116025 2321.21889168 1717.78004085 3151.90672049]

- Since this model is another linear model I knew this model was not going to be the most optimized for this data set but I wanted to try this model anyway.
- With a Non-Validated Score of **0.54** and a Validated Score of **0.47** this model is not optimal considering the upcoming models so I'm not choosing this model for submission.

4) Lasso Regression

```
Lasso Model K-Fold CV Scores: [0.61568129 0.56580364 0.34792764 0.55933251 0.27118679]
Lasso Model K-Fold MSE Scores: [-2739458.8372076  -4818174.20441747 -5015966.51600241 -3846172.07828906
 -2613494.20992934]
Lasso Model Model CV Score Average:  0.4719863753362571
Lasso Model N-MSE Score Average:  -3806653.169169175
Lasso Model Predictions :
 [4431.62374187 1013.35548427 -773.21791395 6367.4524136  3742.0303696
 2484.97483783 3266.44047806 2467.02501431 2461.48241998 4679.75308167
  510.13213527  960.8884896   391.82804278  608.75111787  329.5301485
 5028.05188104  160.11889167 2321.05709449 1717.73291299 3151.85126962]

Lasso Model Score: 0.5445025479634276
```

Non-Validated Score: 0.5445025479634276

K-Fold CV Scores: [0.61568129 0.56580364 0.34792764 0.55933251 0.27118679]

K-Fold CV Score Average: 0.4719863753362571

K-Fold N-MSE Scores: [-2739458.8372076 -4818174.20441747 -5015966.51600241 -3846172.07828906 -2613494.20992934]

K-Fold CV N-MSE Score Average: -3806653.169169175

Model Predictions:

[4431.62374187 1013.35548427 -773.21791395 6367.4524136 3742.0303696

2484.97483783 3266.44047806 2467.02501431 2461.48241998 4679.75308167

510.13213527 960.8884896 391.82804278 608.75111787 329.5301485

5028.05188104 160.11889167 2321.05709449 1717.73291299 3151.85126962]

- This was the last linear model I have included in my project and admittedly it was not any different than the other Linear-based Models except the Polynomial Model.
- With a Non-Validated Score of **0.54** and a Validated Score of **0.47** this model is not optimal considering the upcoming models so I'm not choosing this model for submission.

5) Random Forest

```
Random Forest K-Fold CV Scores: [0.90739399 0.88297571 0.76615926 0.90358313 0.84170965]
Random Forest K-Fold N-MSE Scores: [ -660104.11851807 -1298590.83574107 -1798784.02946267  -841532.18794638
  -567622.70165517]
Random Forest K-Fold CV Score Average:  0.8603643456705411
Random Forest K-Fold CV N-MSE Score Average:  -1033326.7746646717
Random Forest Model Predictions:
 [4296.91811905 1785.31902778  129.3542619  7628.48242857 3821.14784921
 1345.00099603 4011.26290079 1453.40089286 1473.85280556 4299.15249206
  139.26564286   64.63767063 3179.77960714 1044.12649206   80.59548016
 3606.20418254 3006.14305952 1043.96505952  721.8814881  4135.42329365]

Random Forest Model Score: 0.9742025292551322
```

Non-Validated Score: 0.9742025292551322

K-Fold CV Scores: [0.90739399 0.88297571 0.76615926 0.90358313 0.84170965]

K-Fold CV Score Average: 0.8603643456705411

K-Fold N-MSE Scores: [-660104.11851807 -1298590.83574107 -1798784.02946267 -841532.18794638 -567622.70165517]

K-Fold CV N-MSE Score Average: -1033326.7746646717

Model Predictions:

[4296.91811905 1785.31902778 129.3542619 7628.48242857 3821.14784921

1345.00099603 4011.26290079 1453.40089286 1473.85280556 4299.15249206

139.26564286 64.63767063 3179.77960714 1044.12649206 80.59548016

3606.20418254 3006.14305952 1043.96505952 721.8814881 4135.42329365]

- This model was the first complex model we used in this course. In fact, this model was so complex to implement that, our lab assistant decided to let us use libraries in order to implement this model in our lab.
- With a Non-Validated Score of **0.97** and a Validated Score of **0.86** this model was the one of the best results I have acquired in this project.

6) Gradient Boosting

```
Gradient Boosting K-Fold CV Scores: [0.9461908  0.93507671 0.80969587 0.82509721 0.83654812]
Gradient Boosting K-Fold MSE Scores: [ -383556.90455445  -720438.35508109 -1463885.28010878 -1526561.9742312
  -586131.70703329]
Gradient Boosting Model CV Score Average:  0.8705217435921355
Gradient Boosting Model N-MSE Score Average:  -936114.8442017615
Learning rate:  0.1
Gradient Boosting Model Predictions :
 [6180.58063538 1305.46705582  295.58960909 8378.20026105 3211.8906464
  904.77480618 4656.92391147 1575.03011538  881.69717218 4471.83330717
  203.58793691 -413.20563842 2258.29926492  313.31959641  280.12546774
 3801.54306556 3158.91250103 1062.85778283  933.61367892 3864.04522627]

Gradient Boosting Model Score: 0.9976332094480423
```

```
Gradient Boosting K-Fold CV Scores: [0.94180419 0.94370772 0.74014301 0.84886379 0.73048478]
Gradient Boosting K-Fold MSE Scores: [ -414825.06645892  -624662.10591247 -1998910.02393188 -1319125.82730701
  -966470.48935808]
Gradient Boosting Model CV Score Average:  0.8410006980010134
Gradient Boosting Model N-MSE Score Average:  -1064798.7025936742
Learning rate:  0.25
Gradient Boosting Model Predictions :
 [5606.7392419  1166.49347105  189.26286819 8778.10125495 3310.50246466
  771.09913109 4771.4481983  2140.80124736  876.94553493 4119.93789463
  -63.1221326  -100.57552446 2423.34194113 -231.9189543   328.61756479
 2935.5545578  3178.84419278 1025.9284633  1103.23497691 3792.55237316]

Gradient Boosting Model Score: 0.9999335002176734
```

```
Gradient Boosting K-Fold CV Scores: [0.94255388 0.93180627 0.74713913 0.88678935 0.83548756]
Gradient Boosting K-Fold MSE Scores: [ -409481.21335729  -756729.68658107 -1945093.43555002  -988109.29700949
  -589934.83944875]
Gradient Boosting Model CV Score Average:  0.8687552373779498
Gradient Boosting Model N-MSE Score Average:  -937869.6943893215
Learning rate:  0.5
Gradient Boosting Model Predictions :
 [6785.54149411 1449.11807526  338.47057493 8991.70312516 3667.51620766
  289.10538294 4947.07684657 1054.64565789  603.38878817 4585.46918958
  -46.4425255  -182.06599558 2672.85534523  502.45386496  110.08470937
 3920.8605416  3159.37359101 1072.73976054  500.65355159 3506.32311109]

Gradient Boosting Model Score: 0.9999999513345907
```

```
Gradient Boosting K-Fold CV Scores: [0.92160056 0.8373886  0.76231133 0.69646997 0.57454787]
Gradient Boosting K-Fold MSE Scores: [ -558838.36910503 -1804460.18391237 -1828383.58915565 -2649228.16620225
 -1525653.85799989]
Gradient Boosting Model CV Score Average:  0.7584636667826032
Gradient Boosting Model N-MSE Score Average:  -1673312.8332750374
Learning rate:  0.75
Gradient Boosting Model Predictions :
 [6665.45560846  804.73718653 1009.38905609 9998.10986908 3571.11560972
  608.71441664 3409.79627274 1663.30077054  626.51248596 3410.59930589
  316.67973659 -935.27514461  309.48884178 -302.99195815  270.98434665
 3305.75080653 3678.43908332 1183.57253611  977.26593604 3283.35309184]

Gradient Boosting Model Score: 0.9999999997354306
```

- For this model I have used various learning rates to find the sweet-spot for optimization and I have concluded that the learning rate 0.1 is the most optimized one since it has the highest Cross Validation Score (0.8705217435921355) out of them all.

7) XGB Regression

```
XGB Regression K-Fold CV Scores: [0.94969867 0.91523466 0.83737861 0.82224742 0.94945466]
XGB Regression K-Fold MSE Scores: [ -358552.49790819  -940620.93811242 -1250940.08289631 -1551435.09403798
  -181253.49742611]
XGB Regression Model CV Score Average:  0.8948028038543867
XGB Regression Model N-MSE Score Average:  -856560.4220762026
XGB Regression Model Predictions :
 [3222.177     1827.363      176.88905   8101.096     3963.4197
  524.53613   4296.197     1392.9767     467.53726   4463.836
   10.815881   -96.293686  3727.2617    1032.4156      -6.5601473
 4334.6807    3541.7214    1281.1564     672.39575   3955.7278   ]

XGB Regression Model Score: 0.9999999999941673
```

Non-Validated Score: 0.9999999999941673

K-Fold CV Scores: [0.94969867 0.91523466 0.83737861 0.82224742 0.94945466]

K-Fold CV Score Average: 0.8948028038543867

K-Fold N-MSE Scores: [-358552.49790819 -940620.93811242 -1250940.08289631 -1551435.09403798 -181253.49742611]

K-Fold CV N-MSE Score Average: -856560.4220762026

Model Predictions:

[3222.177 1827.363 176.88905 8101.096 3963.4197

524.53613 4296.197 1392.9767 467.53726 4463.836

10.815881 -96.293686 3727.2617 1032.4156 -6.5601473

4334.6807 3541.7214 1281.1564 672.39575 3955.7278]

- This model was the merged version of the Gradient Boosting and Decision Tree Models so my expectations for this model was high and after testing I have seen that it was not misplaced since it has the highest Validated Cross Validation Score out of my models and I have picked this model for submission.

- With a Non-Validated Score of **0.99** and a Validated Score of **0.89** this model generated the highest Cross Validated Score in my project.

8) Light GBM Regression

```
Light GBM Regression K-Fold CV Scores: [0.61070347 0.84895428 0.51603868 0.71493266 0.68558089]
Light GBM Regression K-Fold MSE Scores: [-2774941.18654079 -1676118.62284566 -3722798.13862969 -2488084.7410303
 -1127493.98804158]
Light GBM Regression Model CV Score Average:  0.6752419969798411
Light GBM Regression Model N-MSE Score Average:  -2357887.335417605
Light GBM Regression Model Predictions :
 [6535.05897054 1173.37628982  746.14541436 7608.56975054 2889.24089837
 1717.01290709 5196.85938    1735.44944392 1862.28826221 5998.64988156
  357.54365709 -114.7739166  1989.1730111   643.71609921  240.75382667
 3913.04425493 2766.31015417  808.39174153 1327.80859363 4820.12554869]

Light GBM Regression Model Score: 0.9512551526580857
```

Non-Validated Score: 0.9512551526580857

K-Fold CV Scores: [0.61070347 0.84895428 0.51603868 0.71493266 0.68558089]

K-Fold CV Score Average: 0.6752419969798411

K-Fold N-MSE Scores: [-2774941.18654079 -1676118.62284566 -3722798.13862969 -2488084.7410303 -1127493.98804158]

K-Fold CV N-MSE Score Average: -2357887.335417605

Model Predictions:

[6535.05897054 1173.37628982 746.14541436 7608.56975054 2889.24089837

1717.01290709 5196.85938 1735.44944392 1862.28826221 5998.64988156

357.54365709 -114.7739166 1989.1730111 643.71609921 240.75382667

3913.04425493 2766.31015417 808.39174153 1327.80859363 4820.12554869]

- This model was a bit of a disappointment really since it could not maintain its accuracy scores when subjected to a Cross Validation method and it dropped more significantly considering other models.
- With a Non-Validated Score of **0.95** and a Validated Score of **0.67** this model proved itself to be an unreliable one so I'm not choosing this failed model as my submission.

9) Cat Boost

```
Cat Boost Model K-Fold CV Scores: [0.89333238 0.93043236 0.68599393 0.90693609 0.85290724]
Cat Boost Model K-Fold MSE Scores: [ -760336.53131579  -771975.67889972 -2415443.52253098  -812267.31862977
  -527468.58115923]
Cat Boost Model Model CV Score Average:  0.8539203996598482
Cat Boost Model N-MSE Score Average:  -1057498.3265070969
Cat Boost Regression Model Predictions :
 [3516.74483277 1921.53650535  349.48545721 8573.52169785 3033.74145398
 1547.75717532 4791.11569987 1783.93190343 1660.85236402 4668.03265687
  674.31889562  547.80057788 2809.53128232 1499.76715204  125.81857003
 3503.44287768 2749.77530347 1484.95773952  733.52761755 4798.09615819]

Cat Boost Regression Model Score: 0.9988727879711145
```

Non-Validated Score: 0.9988727879711145

K-Fold CV Scores: [0.89333238 0.93043236 0.68599393 0.90693609 0.85290724]

K-Fold CV Score Average: 0.8539203996598482

K-Fold N-MSE Scores: [-760336.53131579 -771975.67889972 -2415443.52253098 -812267.31862977 -527468.58115923]

K-Fold CV N-MSE Score Average: -1057498.3265070969

Model Predictions:

[3516.74483277 1921.53650535 349.48545721 8573.52169785 3033.74145398

1547.75717532 4791.11569987 1783.93190343 1660.85236402 4668.03265687

674.31889562 547.80057788 2809.53128232 1499.76715204 125.81857003

3503.44287768 2749.77530347 1484.95773952 733.52761755 4798.09615819]

- This model was one of the most promising models that I have experienced in this project. While its score was high it could not beat my XGB Model.
- With a Non-Validated Score of **0.99** and a Validated Score of **0.85** this model proved itself to be an unreliable one so I'm not choosing this model as my submission.

10) K- Nearest Neighbor

```
KNN Regression K-Fold CV Scores: [0.28177922 0.62487922 0.2531647  0.72250096 0.13914759]
KNN Regression K-Fold MSE Scores: [-5119543.25416667 -4162626.51527778 -5744915.94027778 -2422028.12777778
 -3086981.3625    ]
KNN Regression Model CV Score Average:  0.4042943367972936
KNN Regression Model N-MSE Score Average:  -4107219.04
K Nearest Neighbour Model Predictions :
 [2256.         1721.83333333  310.         5986.5        4109.16666667
 3617.66666667 2885.5        4225.66666667 4150.         4412.66666667
  620.66666667 2788.16666667  747.          712.83333333 2739.16666667
 5782.33333333 1087.66666667 1557.33333333  913.16666667 3207.33333333]

K Nearest Neighbour Model Score: 0.6387415084444226
```

Non-Validated Score: 0.6387415084444226

K-Fold CV Scores: [0.28177922 0.62487922 0.2531647 0.72250096 0.13914759]

K-Fold CV Score Average: 0.4042943367972936

K-Fold N-MSE Scores: [-5119543.25416667 -4162626.51527778 -5744915.94027778 -2422028.12777778 -3086981.3625]

K-Fold CV N-MSE Score Average: -4107219.04

Model Predictions:

[2256. 1721.83333333 310. 5986.5 4109.16666667

3617.66666667 2885.5 4225.66666667 4150. 4412.66666667

620.66666667 2788.16666667 747. 712.83333333 2739.16666667

5782.33333333 1087.66666667 1557.33333333 913.16666667 3207.33333333]

- This model scored was one of the worst Cross Validated Score in my project. To be honest I have expected better results from this model.
- With a Non-Validated Score of **0.63** and a Validated Score of **0.40** this model proved itself to be an unoptimized one for this data set.

11) Decision Tree Regression

```
Decision Tree  K-Fold CV Scores: [0.94063814 0.90502053 0.75381458 0.87915881 0.485259  ]
Decision Tree  K-Fold MSE Scores: [ -423136.76128472 -1053964.63360497 -1893743.54827409 -1054709.13388875
 -1845840.04772222]
Decision Tree Model CV Score Average:  0.792778213421241
Decision Tree Model N-MSE Score Average:  -1254278.8249549512
Decision Tree Model Predictions:
 [3263.         1863.          53.14285714 7970.66666667 3431.
 1025.88888889 4228.55555556 1025.88888889 1025.88888889 3431.
  -27.5         -27.5        4228.55555556 1296.          -33.4
 3431.         3602.66666667 1296.          301.2        4228.55555556]

Decision Tree Model Score: 0.9711205854131314
```

Non-Validated Score: 0.9711205854131314

K-Fold CV Scores: [0.94063814 0.90502053 0.75381458 0.87915881 0.485259]

K-Fold CV Score Average: 0.792778213421241

K-Fold N-MSE Scores: [-423136.76128472 -1053964.63360497 -1893743.54827409 -1054709.13388875 -1845840.04772222]

K-Fold CV N-MSE Score Average: -1254278.8249549512

Model Predictions:

[3263. 1863. 53.14285714 7970.66666667 3431.

1025.88888889 4228.55555556 1025.88888889 1025.88888889 3431.

-27.5 -27.5 4228.55555556 1296. -33.4

3431. 3602.66666667 1296. 301.2 4228.55555556]

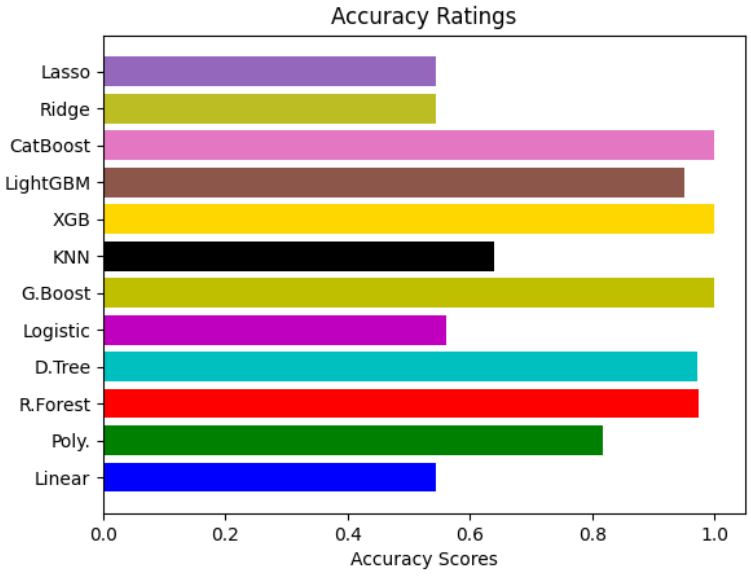
- I expected this model to be fall a little short of more complex variants of this model like Cat Boost or Gradient Boost and I was right. This model follows its more complex brothers very close.
- With a Non-Validated Score of **0.97** and a Validated Score of **0.79** this model proved itself to be one of the most optimized choice for this data set.

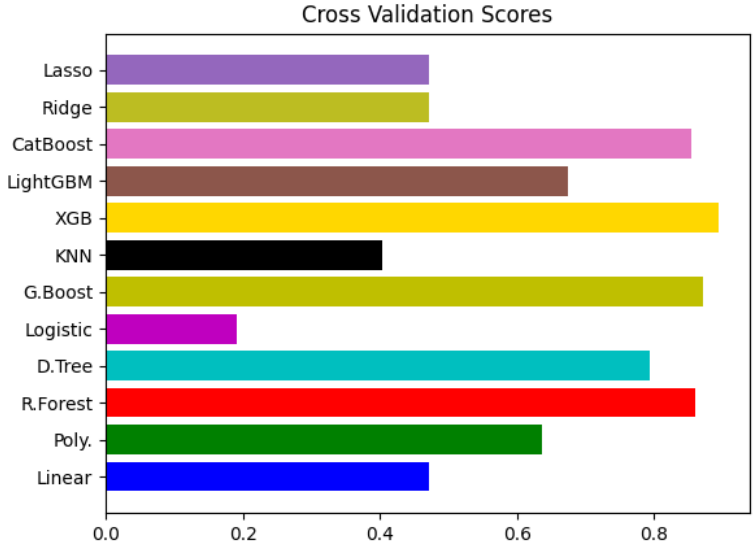
CONCLUSION

To summarize, I was going to try implementing every one of these Models by hand and as we progressed further through our semester I have realized that I lack both Python Programming experience as well as the knowledge about the Prediction Algorithms and how they work so I have decided to use libraries for this problem and I have glad that I did.

The implementation was not difficult since individually all Models can be used with only 3 lines of code thanks to sklearn and other libraries I have used. In this semester, I have learned both Python Programming as well as some Prediction Models that I have researched while trying to implement this project.

The only models that have failed in my project is Logistic Regression and I have not included that model in my report since their accuracy scores dropped more significantly with respect to their Non-Validated Accuracy Scores.





REFERENCES

- quora.com _____ Information about Gradient Boosting
- holypython.com _____ Information about Random Forest
- medium.com _____ Information about Linear Regression
- scikit-learn.org _____ Information about the sklearn library
- towardsdatascience.com _____ Information about the KNN Model
- kaggle.com _____ Information about the Light GBM
- analyticsvidhya.com _____ Information about the Cat Boost
- iq.opengenus.org _____ Information about XGB Regressor
- dhirajkumarblog.medium.com _____ information about the Decision Trees