Admission Data Prediction Using Machine Learning Methods

Problem Describe: Admission Data Prediction

- This dataset was built with the purpose of helping students in shortlisting universities with their profiles. The predicted output gives them a fair idea about their chances for a particular university.
- It contains several parameters which are considered important during the application for Masters Programs.

Mohan S Acharya, Asfia Armaan, Aneeta S Antony : A Comparison of Regression Models for Prediction of Graduate Admissions, IEEE International Conference on Computational Intelligence in Data Science 2019

Dataset

- The parameters included are :
  - GRE Scores (out of 340)
  - TOEFL Scores (out of 120)
  - University Rating (out of 5)
  - Statement of Purpose and Letter of Recommendation Strength (out of 5)
  - Undergraduate GPA (out of 10)
  - Research Experience (either 0 or 1)
- The Prediction is:
  - Chance of Admit (ranging from 0 to 1)

Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of Admit
1	337	118	4	4.5	4.5	9.65	1	0.92
2	324	107	4	4	4.5	8.87	1	0.76
3	316	104	3	3	3.5	8	1	0.72
4	322	110	3	3.5	2.5	8.67	1	0.8
5	314	103	2	2	3	8.21	0	0.65
6	330	115	5	4.5	3	9.34	1	0.9
7	321	109	3	3	4	8.2	1	0.75
8	308	101	2	3	4	7.9	0	0.68
9	302	102	1	2	1.5	8	0	0.5
10	323	108	3	3.5	3	8.6	0	0.45
11	325	106	3	3.5	4	8.4	1	0.52
12	327	111	4	4	4.5	9	1	0.84

Methodology

- Linear Regression
  - Least square
  - Ridge regression
  - Lasso regression
- KNN
- Boosting
  - Random Forest
  - AdaBoost
- SVM
- Decision tree
- Classification
  - LDA
  - Naïve Bayes
  - Logistic

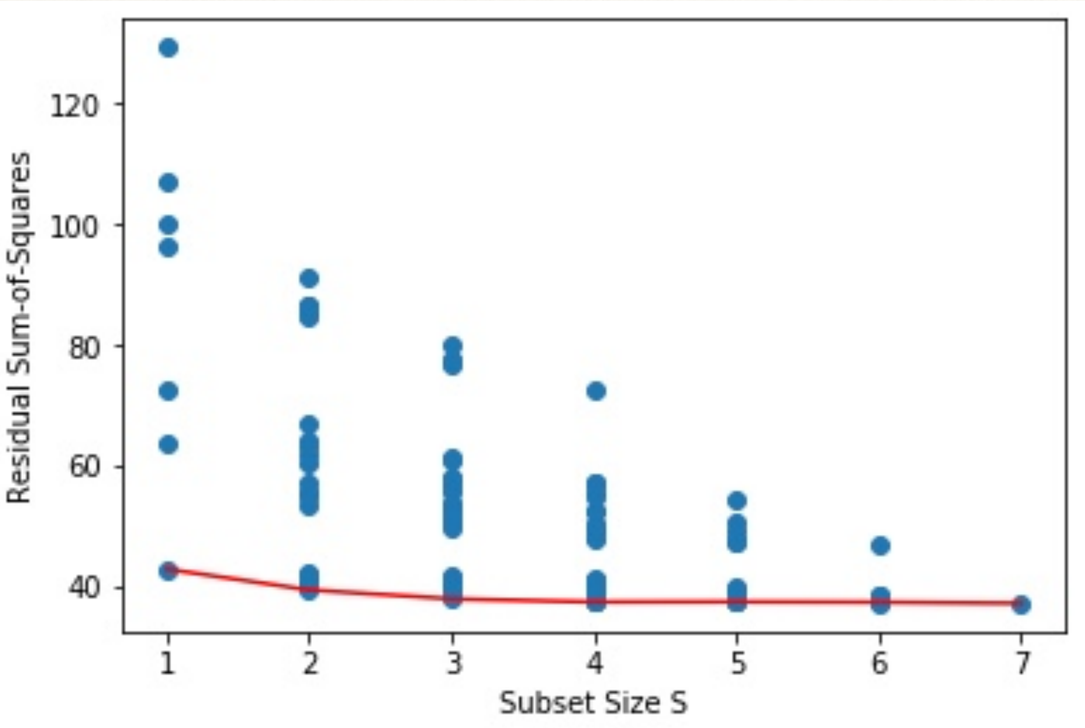
Experiments

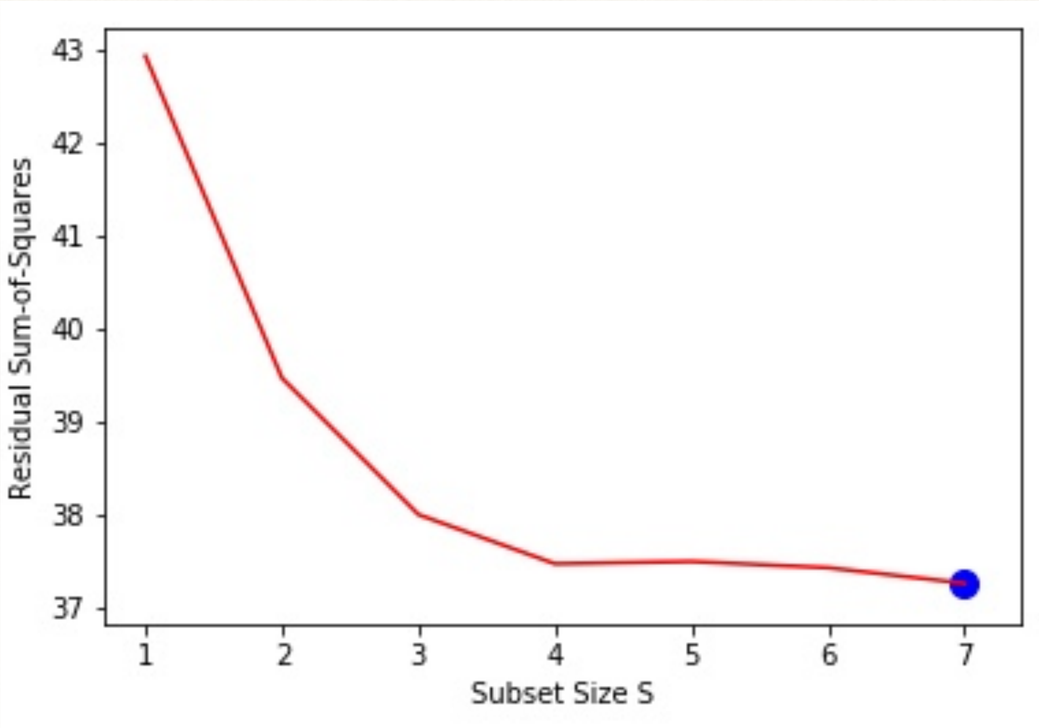
- Preprocessing
- Experiment Results
- Comparison

Preprocessing

- Split Data: 320 training, 80 validation, 100 testing
- Subset Selection
- Normalization
- Discretize label data for some certain algorithms

Preprocessing

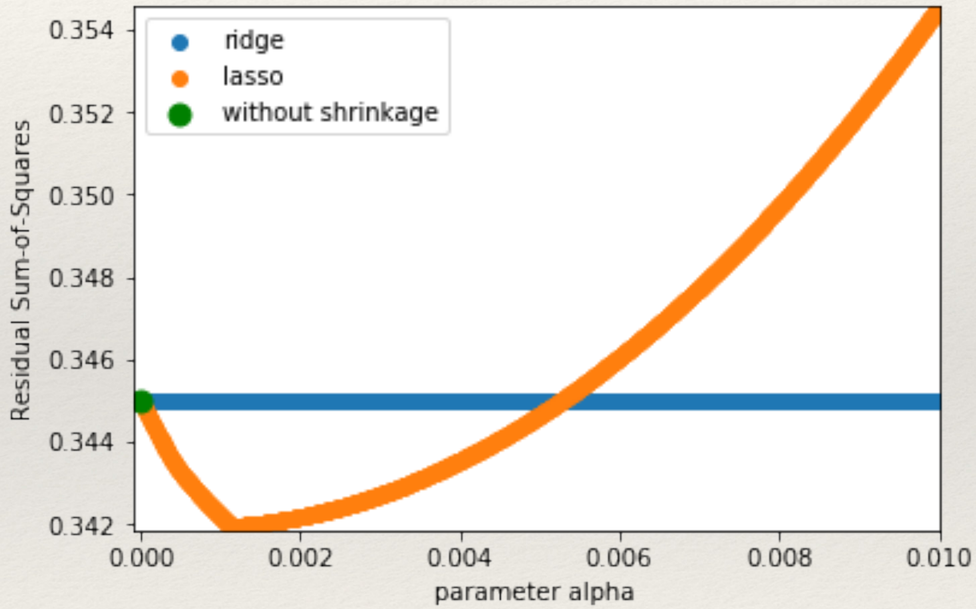




Experiment Results

1. Regression

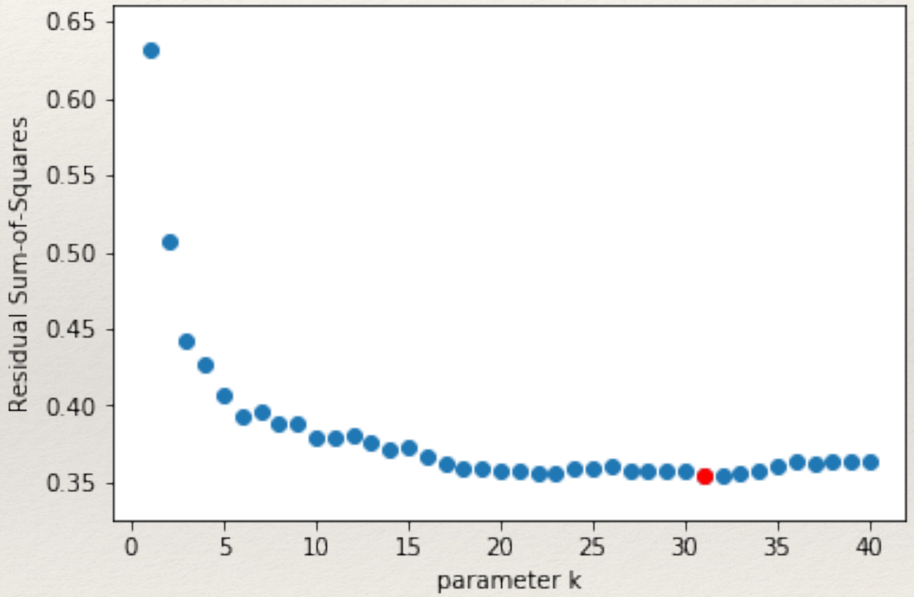
- Regression with different shrinkage method
  - RSS error:
    - Without shrinkage: 0.3450
    - Ridge: 0.3450
    - Lasso: 0.3419
  - Lasso Val Accuracy: 88.956%
  - Lasso Test Accuracy: 78.377%



Experiment Results

2. *KNN*

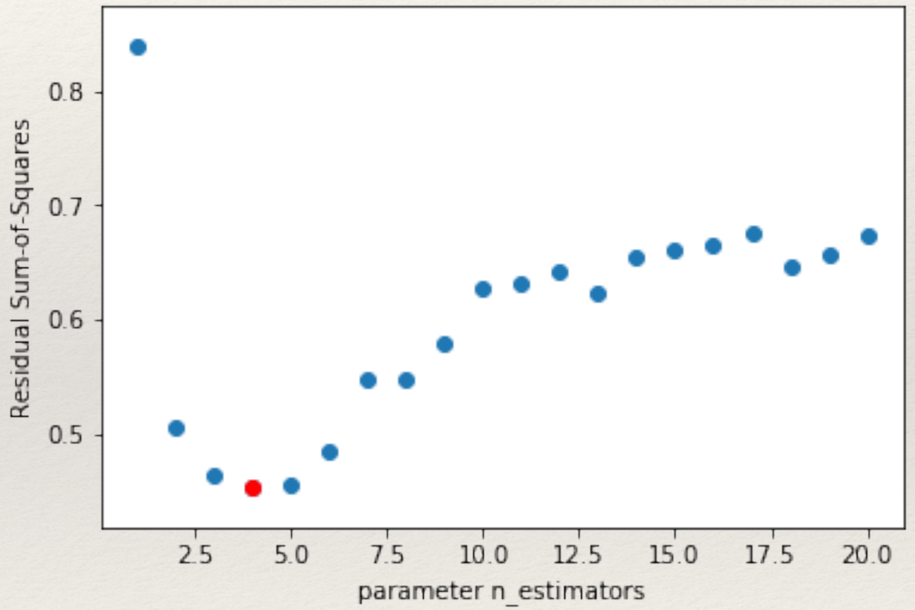
- KNN: Analyze k value
  - optimal k: 31
  - RSS: 0.3546
  - Val Accuracy: 77.199%
  - Test Accuracy: 66.657%



Experiment Results

3. Decision Tree

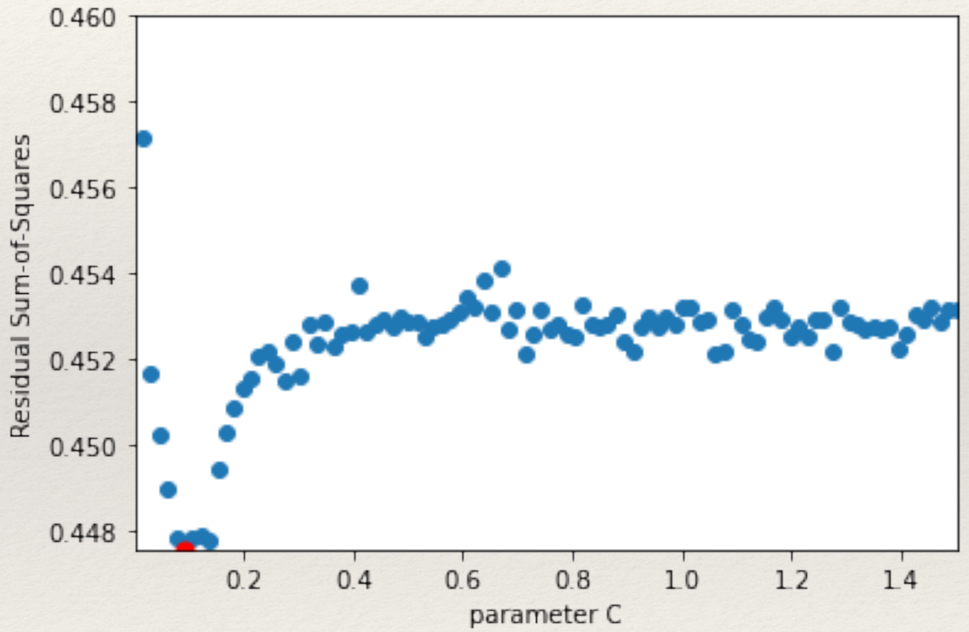
- Decision Tree: Analyze depth value
  - optimal depth: 4
  - RSS: 0.4518
  - Val Accuracy: 81.410%
  - Test Accuracy: 74.015%



Experiment Results

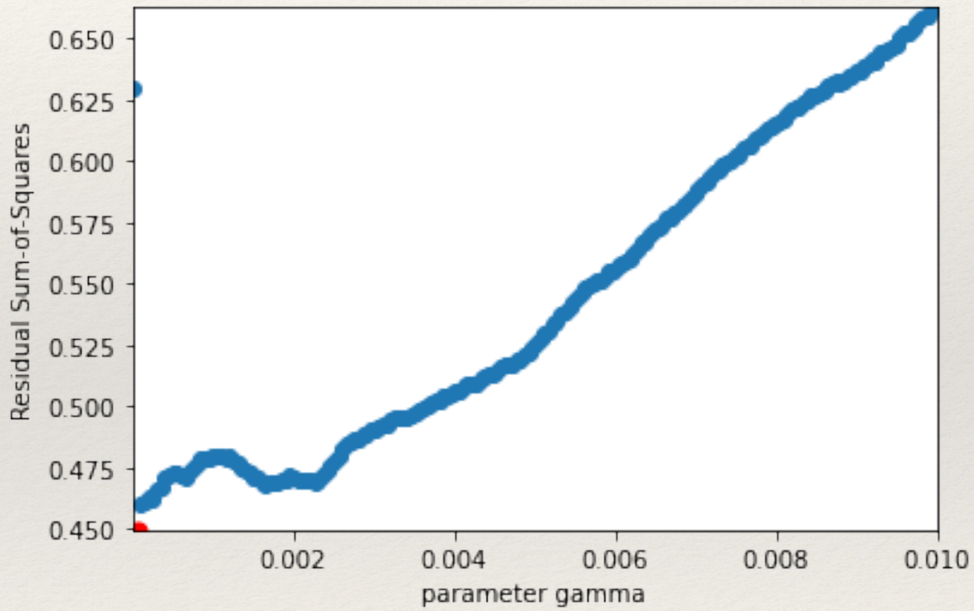
4. SVM

- SVM: With different kernel
  - linear kernel:
    - C: 0.0918
    - RSS: 0.4476
    - Val Accuracy: 76.862%
    - Test Accuracy: 54.370%



Experiment Results

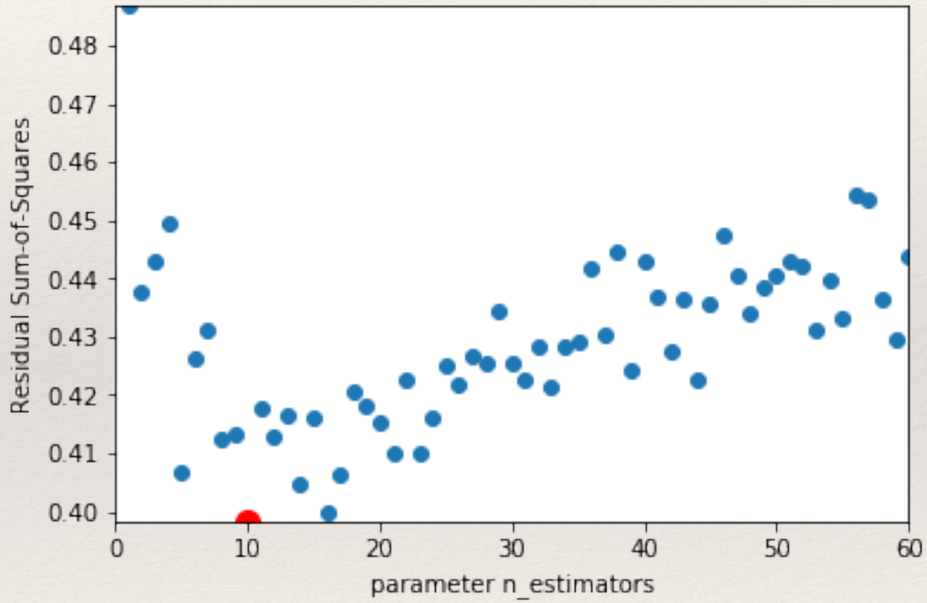
- SVM: With different kernel
  - rbf kernel:
    - optimal gamma: 5.0351e-05
    - RSS: 0.4495
    - Val Accuracy: 61.790%
  - poly kernel:
    - degree: 1
    - RSS: 0.4532
    - Val Accuracy: 62.110%



Experiment Results

5. *AdaBoost*

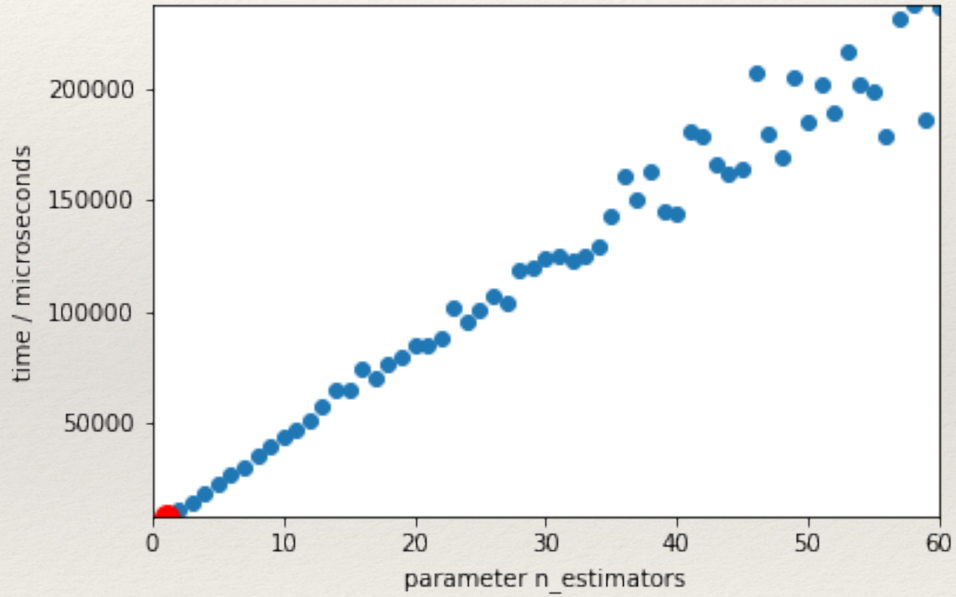
- AdaBoost: Analyze #estimators
  - the lowest residual sum-of-squares:
    - n_estimators: 10
    - RSS: 0.3951
    - Val Accuracy: 81.611%
    - Test Accuracy: 65.844%



Experiment Results

5. AdaBoost

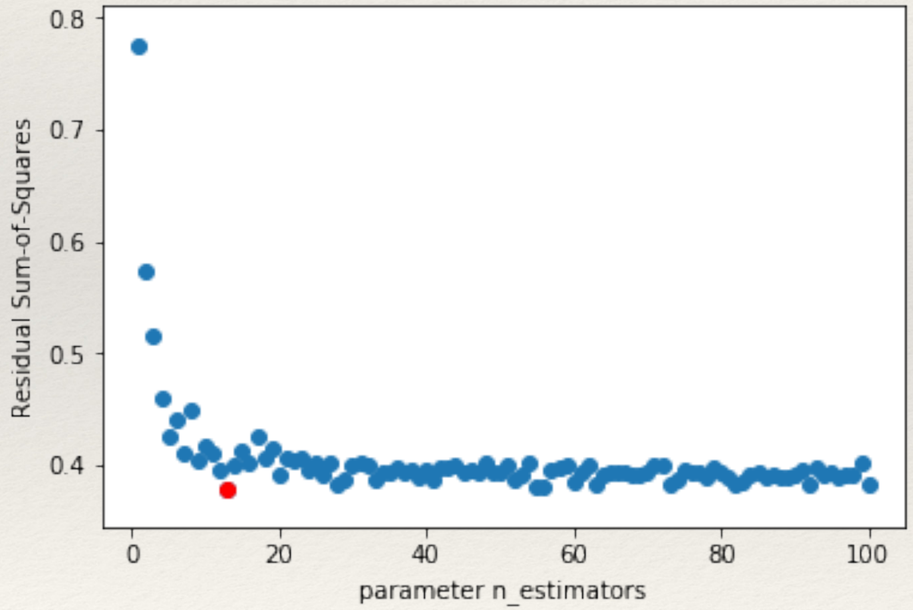
- AdaBoost: Analyze #estimators
  - the running time:
    - n_estimators: 1
    - Min time: 11035 microseconds

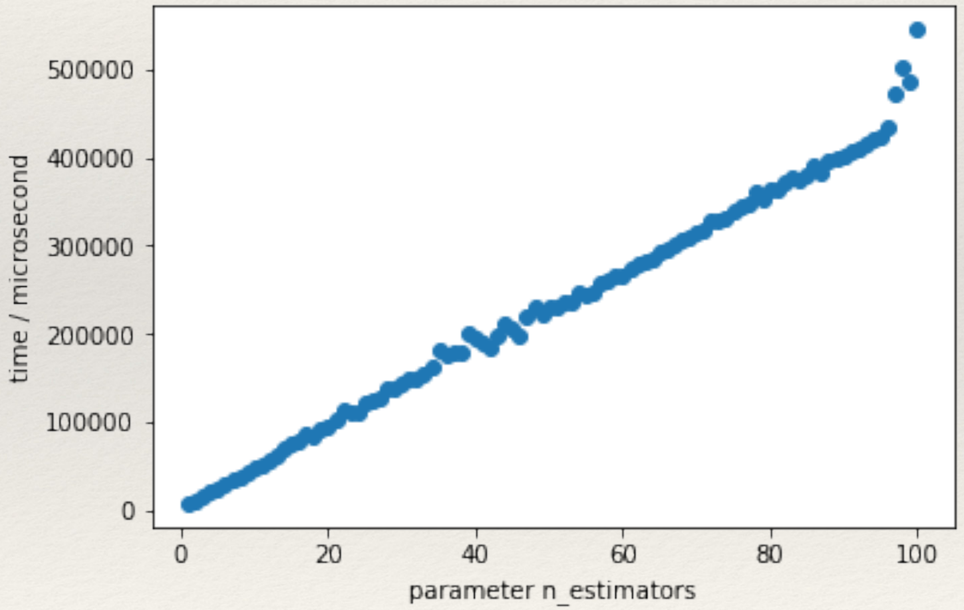


Experiment Results

6. Random Forest

- Random Forest: Analyze #estimators

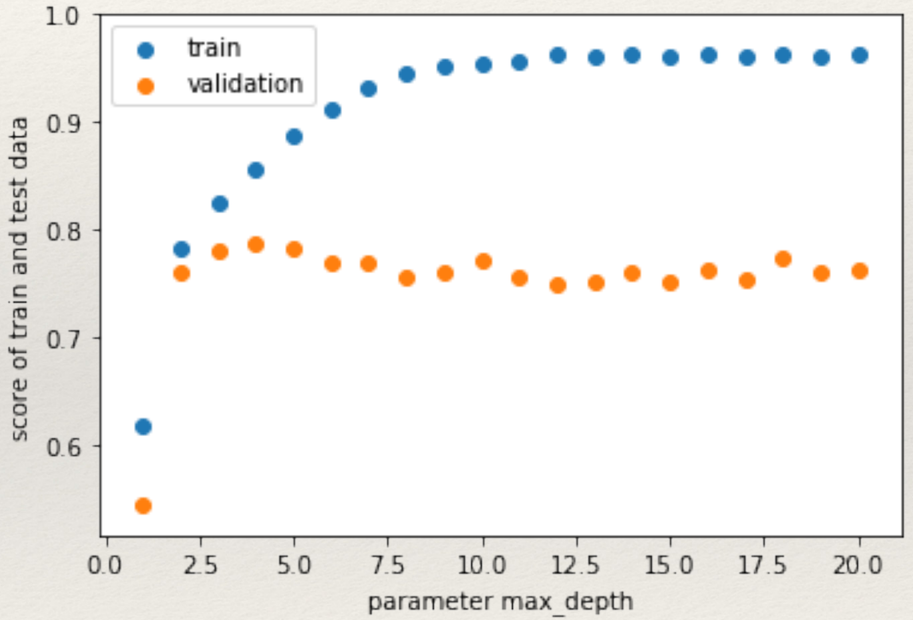




Experiment Results

6. Random Forest

- Random Forest: Analyze max depth
  - Optimal depth: 4
  - Optimal #estimators: 13
  - RSS: 0.3797
  - Val Accuracy: 84.684%
  - Test Accuracy: 74.522%



Experiment Results

7. Classification Methods

- Logistic
  - penalty function: l1
  - RSS: 0.7880
  - Val Accuracy: 58.333%
  - Test Accuracy: 52.500%
- Naïve Bayes
  - Model: Gaussian NB
  - RSS: 0.7
  - Val Accuracy: 64.286%
  - Test Accuracy: 58.750%
- LDA
  - solver: SVD
  - RSS: 0.5833
  - Val Accuracy: 66.666%
  - Test Accuracy: 51.250%

Comparison

Algorithm	RSS error	Val Accuracy	Test Accuracy	Time(Microsecond)
Regression(Lasso)	**0.3419**	**88.956%**	**78.377%**	**1293**
KNN	**0.3546**	77.199%	66.657%	3089
Decision Tree	0.4518	81.410%	**74.015%**	**867**
SVM(Linear)	0.4476	76.862%	54.370%	3122
AdaBoost	0.3982	**81.611%**	65.844%	9639
Random Forest	**0.3797**	**84.684%**	**74.522%**	**11593**
LDA	0.5440	66.667%	51.250%	1400
Naive Bayes(Gaussian)	0.7000	64.286%	58.750%	1599
Logistic(L1-penalty)	0.7880	58.333%	52.500%	4017

Thanks