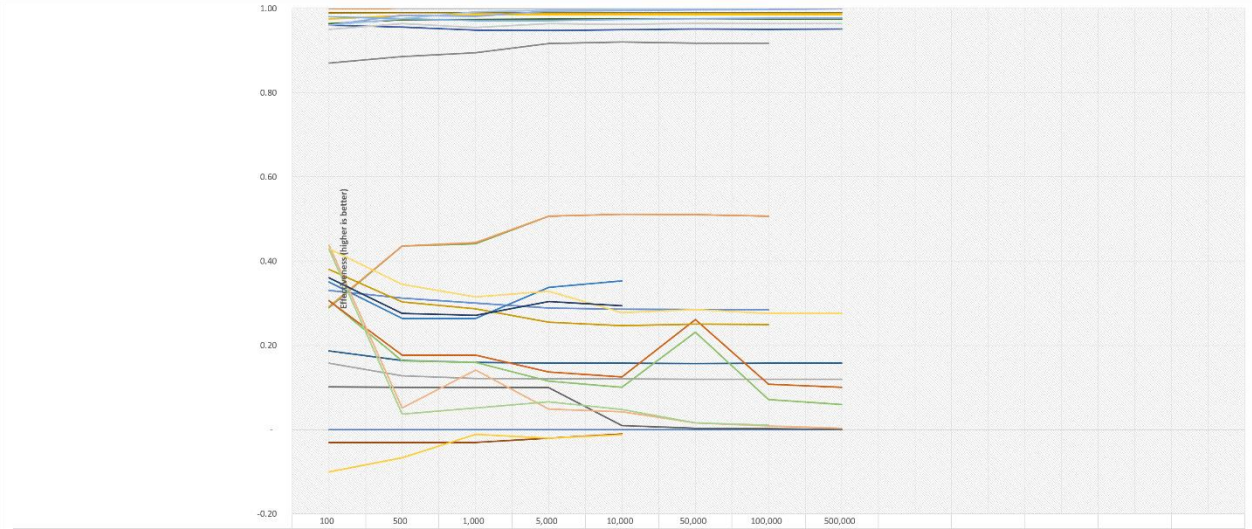
The (Un?) Reasonable Effectiveness of Data: Report

Team: team_31
Student IDs: (Nanbita Roy) 17305618, (Abhimanyu Hazarika)17314158, (Bhavik Mer)17304936
Total Time Required (in hours): 36 hours (approx.)



	100	500	1,000	5,000	10,000	50,000	100,000	500,000
Linear Regression -- The SUM Dataset (without noise) -- RMSE	-	-	-	-	-	-	-	-
Linear Regression -- The SUM Dataset (without noise) -- R2	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Linear Regression -- The SUM Dataset (with noise) -- RMSE	0.16	0.13	0.12	0.12	0.12	0.12	0.12	0.12
Linear Regression -- The SUM Dataset (with noise) -- R2	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99
Linear Regression -- Skin_NonSkin -- RMSE	0.33	0.31	0.30	0.29	0.29	0.29	0.28	
Linear Regression -- Skin_NonSkin -- R2	0.29	0.44	0.44	0.51	0.51	0.51	0.51	
Linear Regression -- OnlineNewsPopularity -- RMSE	381,010,975,407.22	5,436,238,160.24	25,436,238,160.24	1.00	1.00			
Linear Regression -- OnlineNewsPopularity -- R2	-0.03	-0.03	-0.03	-0.02	-0.01			
Lasso Regression -- The SUM Dataset (without noise) -- RMSE	0.10	0.10	0.10	0.10	0.01	0.00	0.00	0.00
Lasso Regression -- The SUM Dataset (without noise) -- R2	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
Lasso Regression -- The SUM Dataset (with noise) -- RMSE	0.19	0.16	0.16	0.16	0.16	0.16	0.16	0.16
Lasso Regression -- The SUM Dataset (with noise) -- R2	0.96	0.97	0.97	0.98	0.97	0.98	0.98	0.98
Lasso Regression -- Skin_NonSkin -- RMSE	0.33	0.31	0.30	0.29	0.29	0.29	0.29	
Lasso Regression -- Skin_NonSkin -- R2	0.29	0.44	0.44	0.51	0.51	0.51	0.51	
Lasso Regression -- OnlineNewsPopularity -- RMSE	1.01	1.00	1.00	1.00	1.00			
Lasso Regression -- OnlineNewsPopularity -- R2	-0.10	-0.07	-0.01	-0.02	-0.01			
Logistic Regression -- The SUM Dataset (without noise) -- Accuracy	0.98	0.98	0.97	0.97	0.97	0.98	0.98	0.98
Logistic Regression -- The SUM Dataset (without noise) -- log loss	0.31	0.16	0.16	0.12	0.10	0.23	0.07	0.06
Logistic Regression -- The SUM Dataset (with noise) -- Accuracy	0.96	0.96	0.95	0.95	0.95	0.95	0.95	0.95
Logistic Regression -- The SUM Dataset (with noise) -- log loss	0.31	0.18	0.18	0.14	0.13	0.26	0.11	0.10
Logistic Regression -- Skin_NonSkin -- Accuracy	0.87	0.89	0.90	0.92	0.92	0.92	0.92	
Logistic Regression -- Skin_NonSkin -- log loss	0.38	0.30	0.29	0.25	0.25	0.25	0.25	
Logistic Regression -- OnlineNewsPopularity -- Accuracy	0.35	0.26	0.26	0.34	0.35			
Logistic Regression -- OnlineNewsPopularity -- log loss	1.97	1.83	1.55	1.35	1.34			
KNN Classification -- The SUM Dataset (without noise) -- Accuracy	0.96	0.98	0.98	0.99	0.99	1.00	1.00	1.00
KNN Classification -- The SUM Dataset (without noise) -- log loss	0.44	0.05	0.14	0.05	0.04	0.02	0.01	0.00
KNN Classification -- The SUM Dataset (with noise) -- Accuracy	0.95	0.96	0.96	0.96	0.96	0.96	0.96	0.96
KNN Classification -- The SUM Dataset (with noise) -- log loss	0.43	0.35	0.31	0.33	0.28	0.29	0.28	0.28
KNN Classification -- Skin_NonSkin -- Accuracy	0.96	0.98	0.99	1.00	1.00	1.00	1.00	
KNN Classification -- Skin_NonSkin -- log loss	0.43	0.04	0.05	0.07	0.05	0.02	0.01	
KNN Classification -- OnlineNewsPopularity -- Accuracy	0.36	0.28	0.27	0.30	0.29			
KNN Classification -- OnlineNewsPopularity -- log loss	7.12	9.43	9.90	8.47	9.31			

Findings/Answer

Question 1: To what extent does the effectiveness of machine-learning algorithms depend on the size and complexity of the data? [200-300 words]

More the algorithm is trained the better it becomes; we saw some improvement when Logistic Regression, 5-Neighbors Classifier, and the Line Regression were fed more data. Log-loss metric gets closer to 0 when the size of the data grows, thus achieving better prediction. From the result, it is quite clear that after one point more SUM dataset with noise and without noise metrics doesn't change significantly. The metrics remained constant above 50,000 observations for Sum dataset without noise.

Talking about most straightforward dataset SkinNonSkin show that only three features (B, G, R) describing the target class accurately (Skin/Not skin) can be enough which gives the highest accuracy for 100,000 observation. Talking about k-neighbor classifier, we observed that at particularly two values, i.e., when n_neighbors = 3 and n_neighbors = 23 performed best for SUM dataset . We plotted the target against the predicted values dataset with 70/30 and k-fold cross-validation. Both validations gave the best accuracy output with an accuracy of "0.96".

On the other hand, Online News Popularity dataset is very complicated with more than 61 features. Here instead of selecting Target feature. We created Target feature based on the ["share"] which is quantitative in nature; we used percentile to divide the values, event after proper distribution of data the DataSet didn't give a proper prediction. This dataset served as the perfect example where the model was trying to do overfitting of test data; by attempting to fit more features thus hindering machine learning process.

Therefore, we conclude that increasing complexity does not necessarily entail better effectiveness, it depends on the classification we make and metrics valuation understanding. The important take away here was over-fitting and understanding of how more data help learn faster.

Question 2: Looking only at the performance of your best performing machine-learning algorithm on "The SUM dataset (without noise)": how well was machine-learning suitable to solve the task of predicting a) the target value and b) the target class? Consider in your assessment, how well a simple rule-based algorithm would have performed. [100 words max]

i. Machine learning algorithm was able to correctly predict the target value of the SUM dataset (without noise) for both RMSE and R2 metrics using the Linear Regression algorithm. These expected optimal results predicted variable that y is a linear combination of the features X.

ii. Best classifier k-neighbors is also anticipating the target class very accurately. However, the results aren't perfect as clustering the training data into four categories can induce erroneous predictions at the cluster's borders. A simple rule-based algorithm would have performed well here (e.g., if y> 20000 were we encode the categorical data into binary values etc.).

Data, Algorithms, etc.

Algorithm 1	Linear Regression
Algorithm 2	Lasso Regression
Algorithm 3	Logistic Regression
Algorithm 4	KNN Classification
Dataset 1	The SUM Dataset (without noise)
Dataset 2	The SUM Dataset (with noise)
Dataset 3	Skin_NonSkin
Dataset 4	OnlineNewsPopularity
Metric 1	RMSE
Metric 2	R2
Metric 3	Accuracy
Metric 4	Log loss

Contributions (max. 200 words)

17305618 implemented Logistic Regression and in major worked on generating the CSV files, 17314158 implemented Decision Tree Classifier and in major worked on overall integration of the code, 17304936 implemented KNN Regression and in major worked on the final Report. Linear Regression was the starting point and all of us worked on it together in order to be familiar with coding Python with respect to Machine Learning and sklearn framework. Also, we individually worked on calculating the metrics associated with the algorithms and the code was merged later. We took up 'The SUM Dataset' initially to implement the algorithms and worked individually to implement the algorithms on it. Finally, once we had successfully completed analyzing 'The SUM Dataset' without noise for regression and classification algorithms, we arrived at a conclusion of choosing 'Dataset Skin Non-Skin' and 'OnlineNewsPopularity'. 17305618 worked on assessing the 'Dataset Skin Non-Skin', 17314158 worked on 'The SUM Dataset with noise' and 17304936 worked on '3d Road Network'. We worked on them for both – regression and classification algorithms. To summarize, we worked as a team, delegated tasks equally and helped each other to accomplish the task.