**Internship 2019:Privacy Issues in AI**

**Main objectives of the project:**

* The idea is to create multiclass classifiers using the machine learning (logistic regression,support vector machine, decision tree classifier using Scikit Learn Library) and deep learning (Neural networks using Keras Library).
* Then, develop the hierarchical and tournament(ensemble) classifiers using the same methods.
* To generate the noisy data in both a targeted and non-targeted manner to get a good picture of the trends of the varying accuracy with the increasing distortion

**Pre-requisites:**

* The original MNIST data set was downloaded from a Kaggle competition on distinguishing handwritten digits.
* I used Google Colaboratory to compensate for the less GPU on the current system
* Installation of libraries like Matplotlib,Scikit Learn, Pandas, Numpy, Keras on Tensorflow backend
* Coding Language used is Python

**Experiment 1**

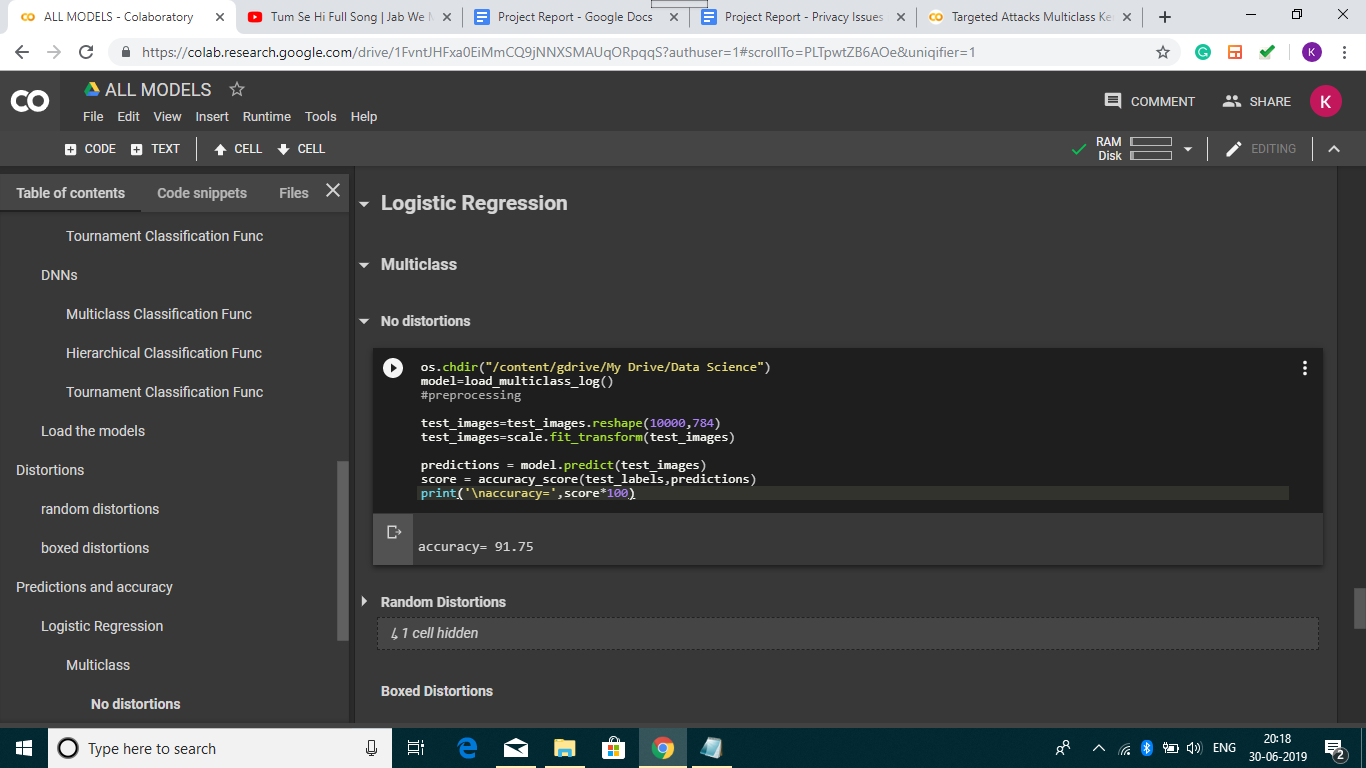
Objective:

To make multiclass, hierarchical and tournament models using logistic regression, SVM, decision trees and to compare their accuracies using MNIST dataset.

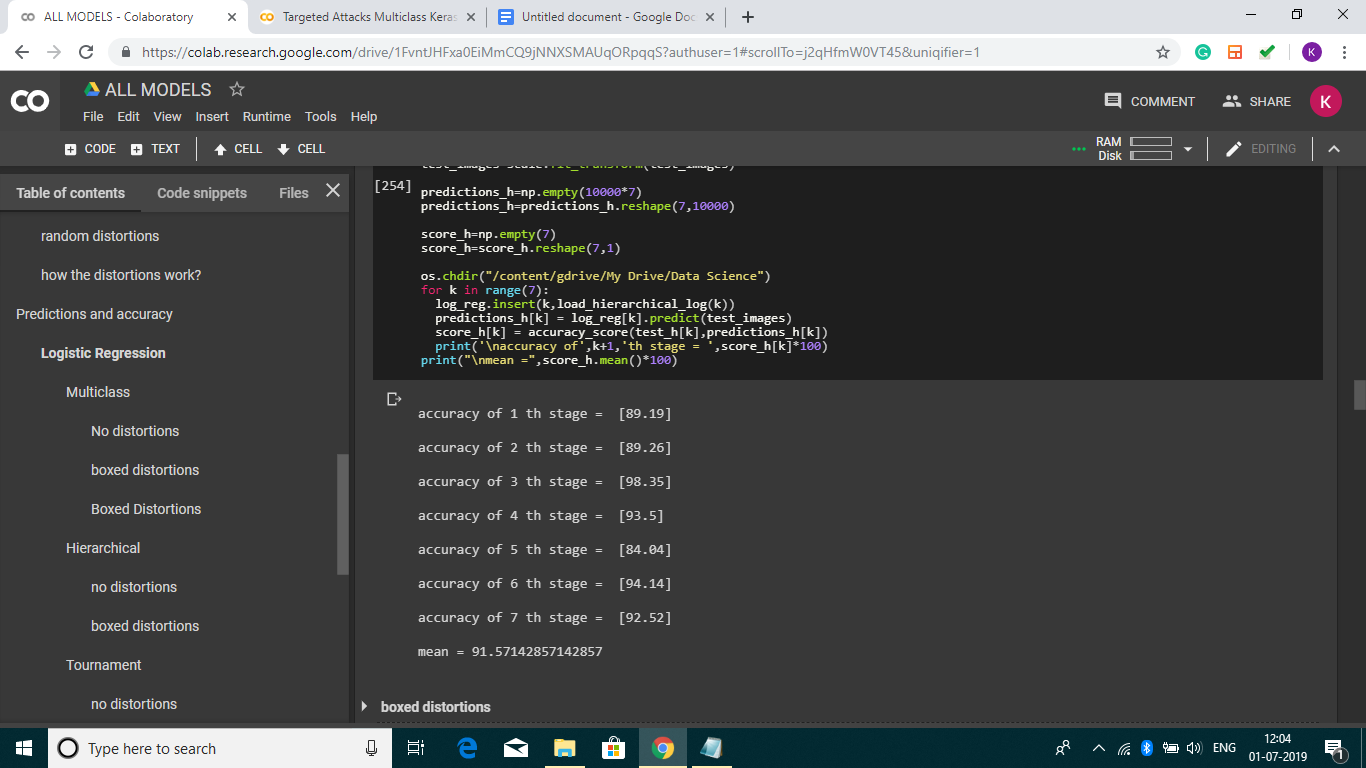
Observations:

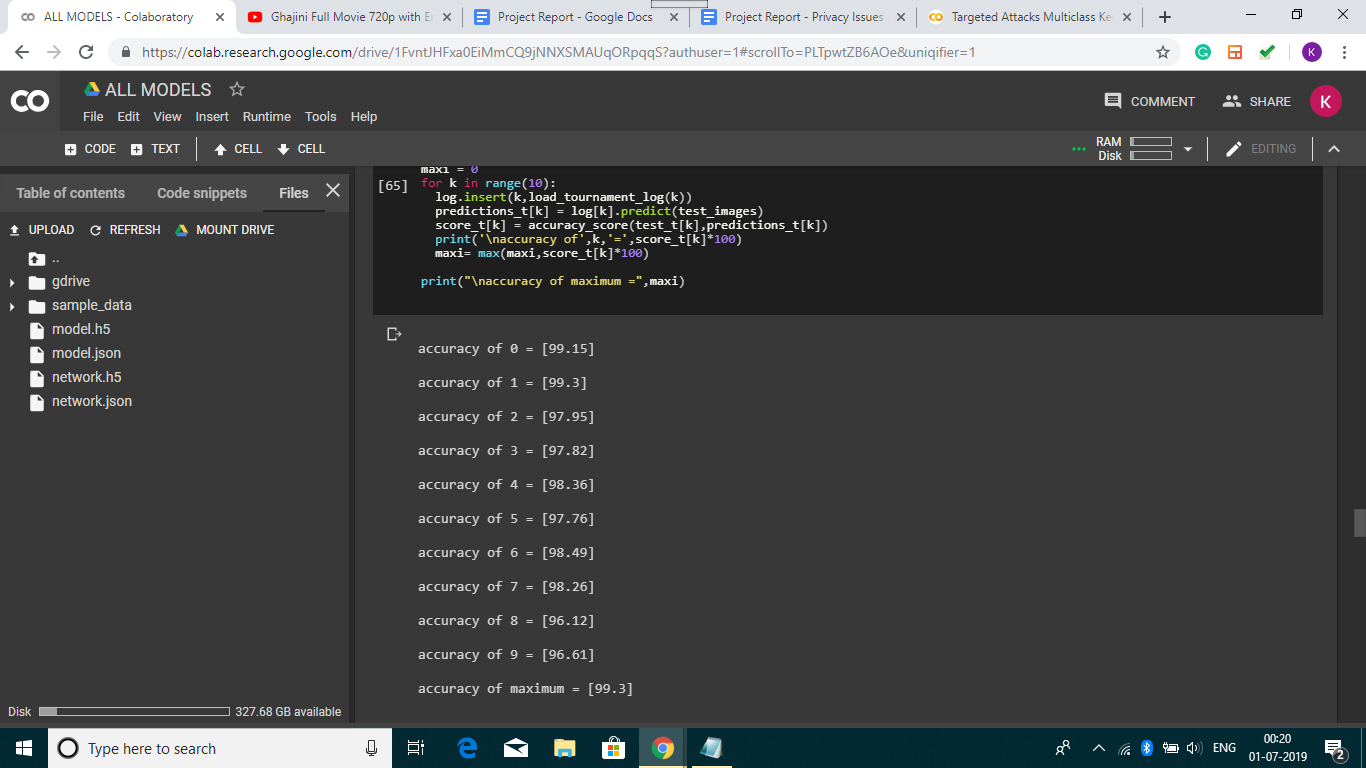
**Logistic Regression:**

*Multiclass*



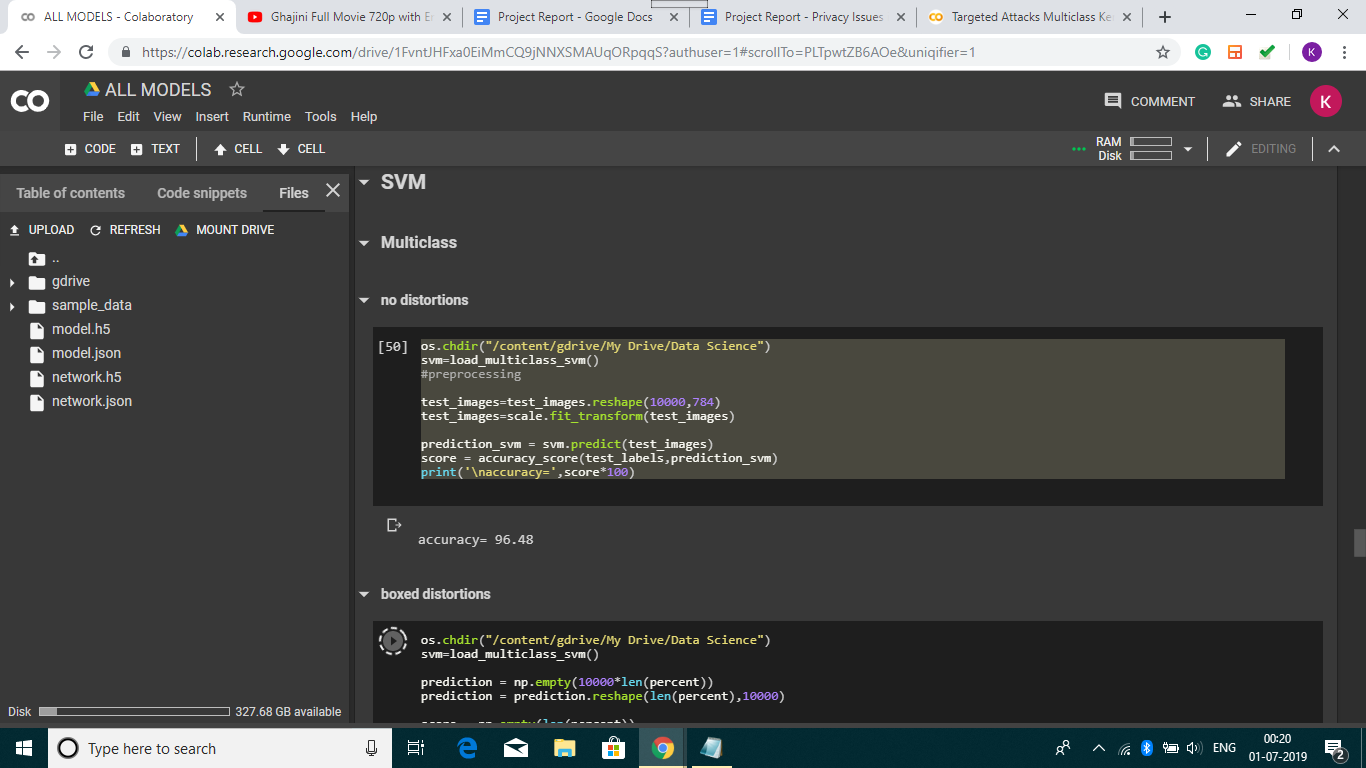
*Hierarchical*

**

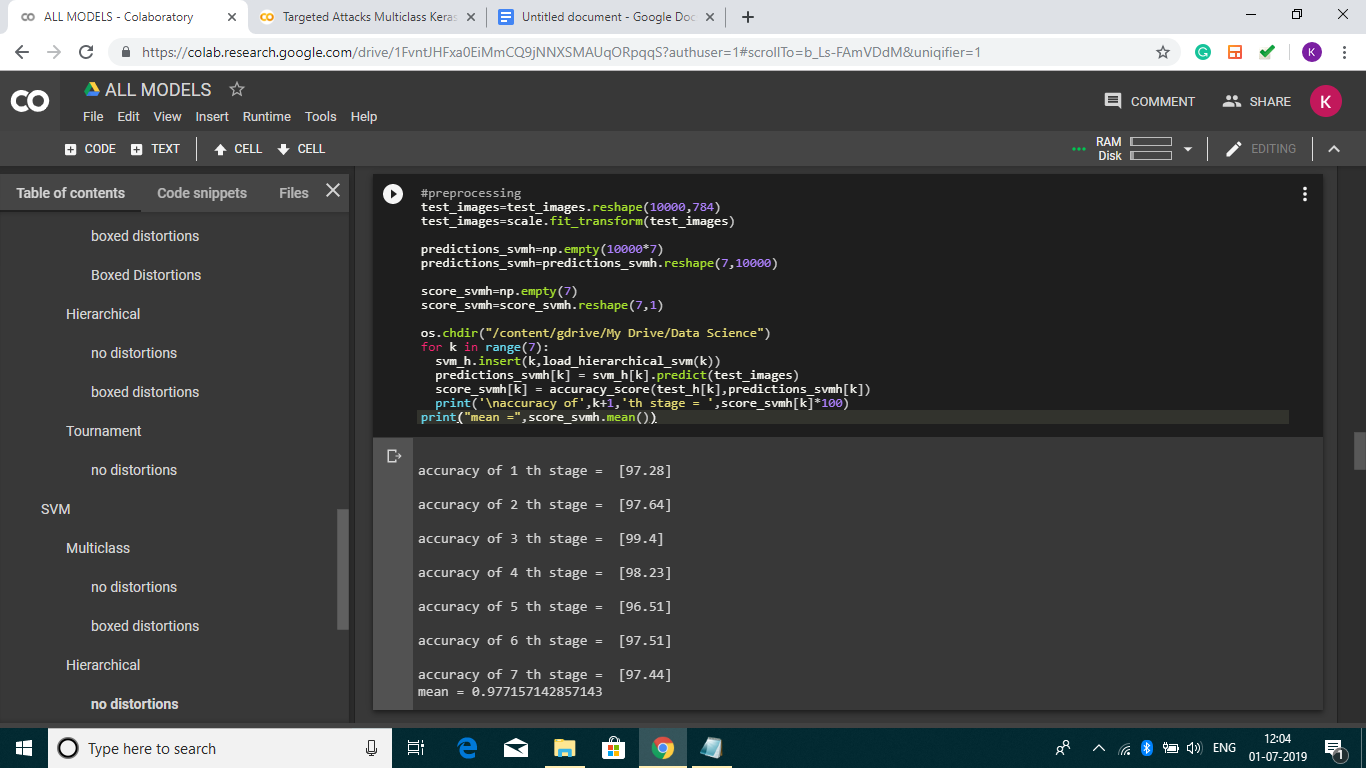
*Tournament*

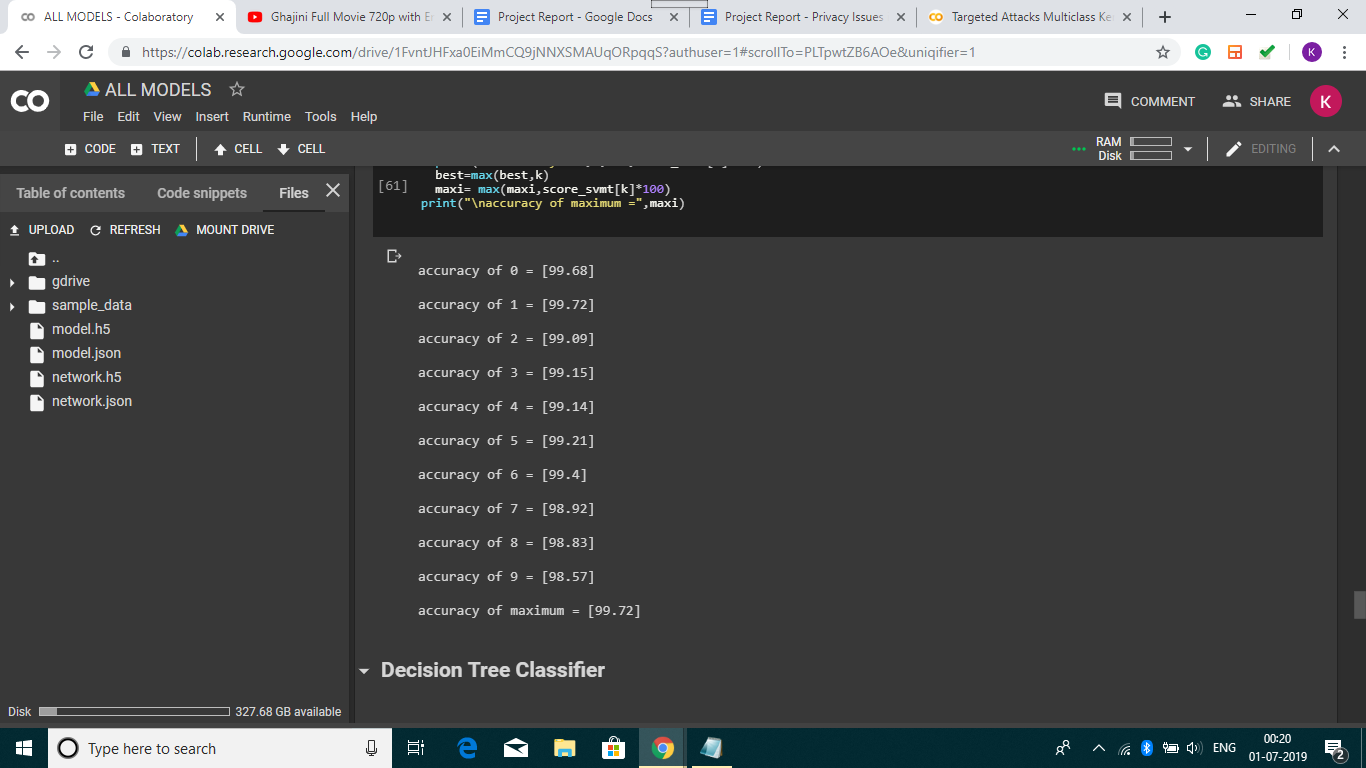
**SVM:**

*Multiclass*



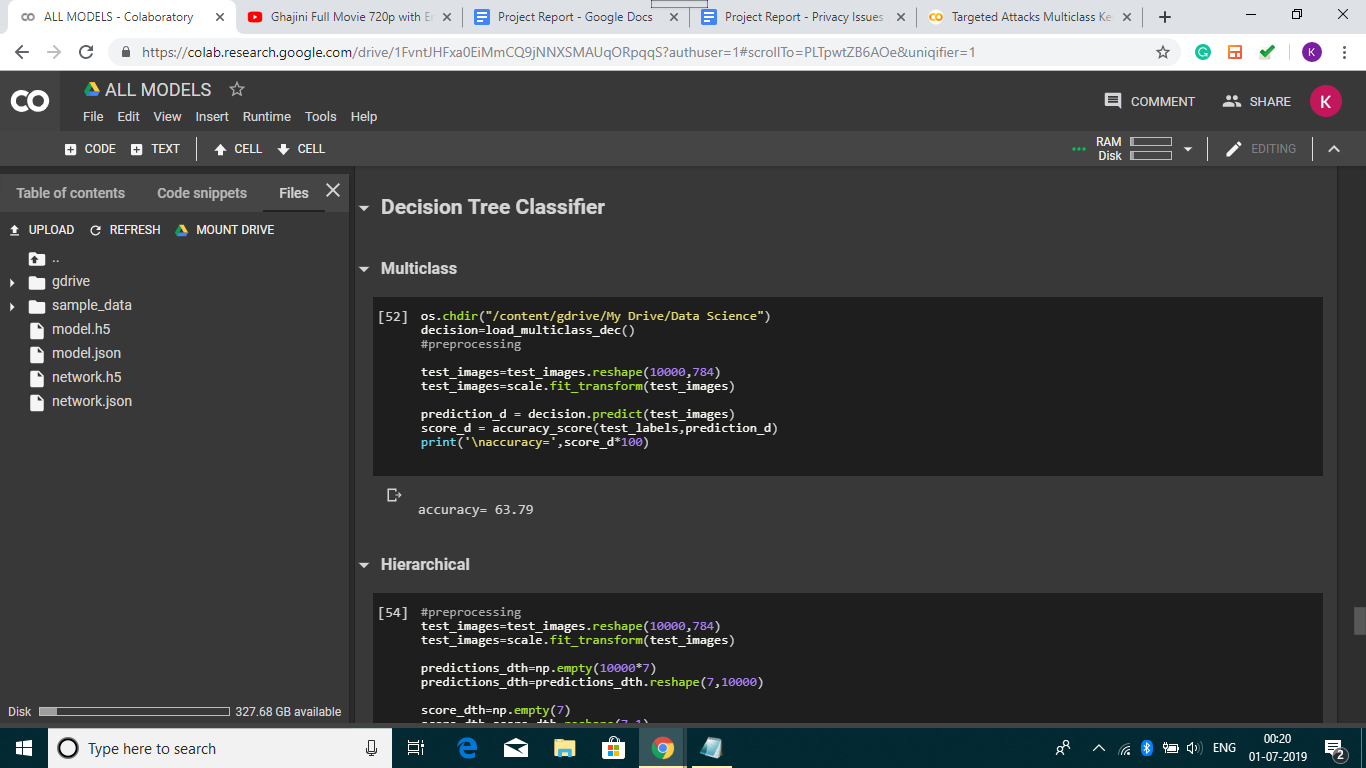
*Hierarchical*



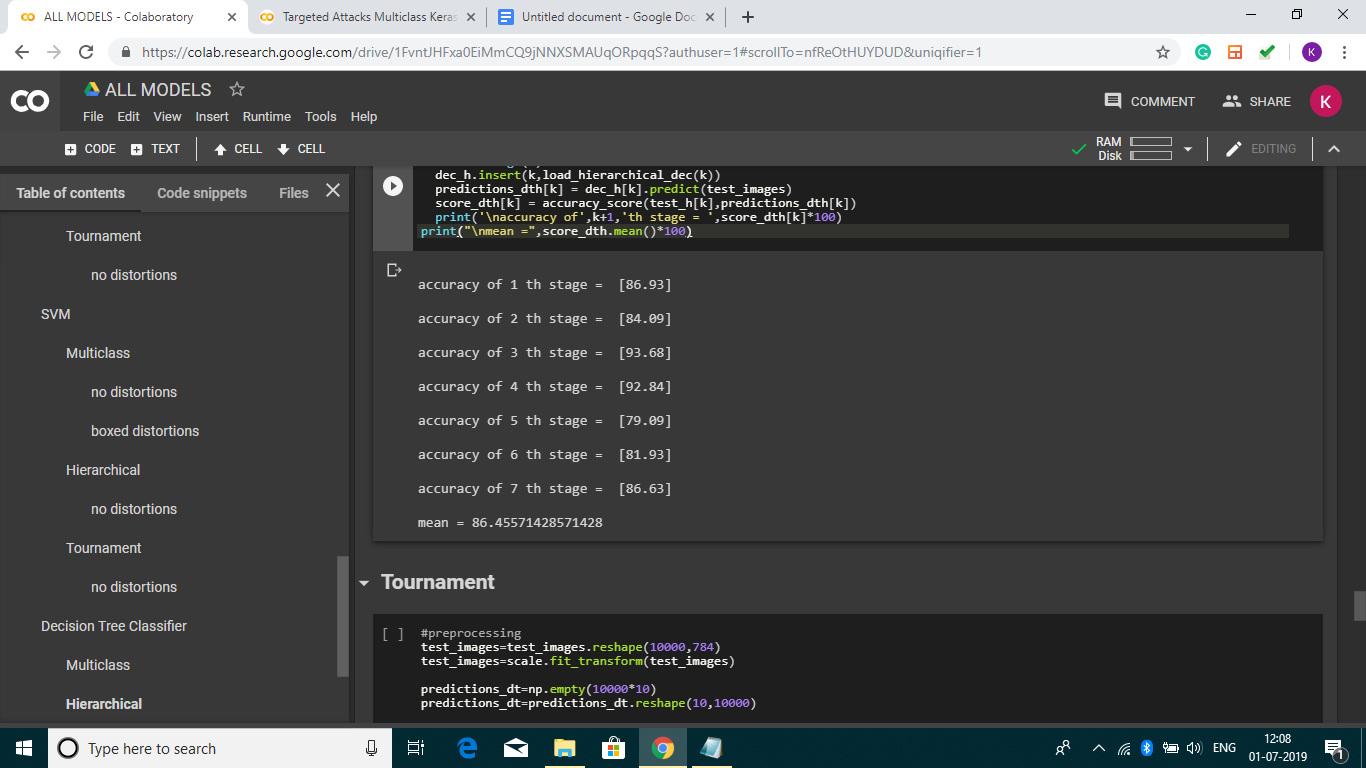
*Tournament*

**Decision Tree Classifier:**

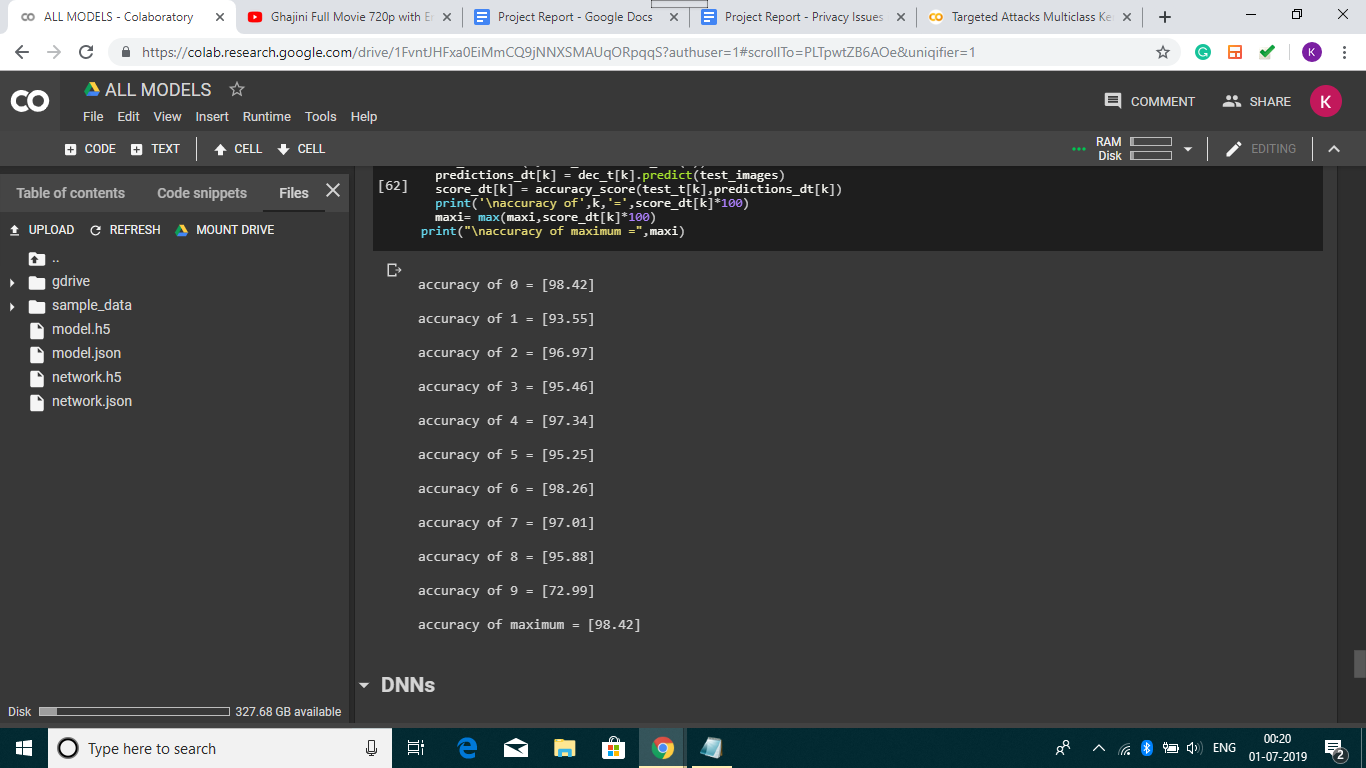
*Multiclass*



*Hierarchical*

**

*Tournament*



Tabulations:

|  |  |  |
| --- | --- | --- |
| Algorithm | Model type | Accuracy obtained |
| Logistic Regression | Multiclass | 91.75% |
|  | Hierarchical | 91.57% |
|  | Tournament | 99.3% |
| SVM | Multiclass | 96.48% |
|  | Hierarchical | 97.72% |
|  | Tournament | 99.72% |
| Decision Tree Classifier | Multiclass | 63.79% |
|  | Hierarchical | 86.46% |
|  | Tournament | 98.42% |

Inferences:

* The tournament models of every algorithm has the highest accuracy, followed by the hierarchical with an intermediate accuracy and the multiclass with the least.
* Of all the algorithms, SVM has shown the best performance.
* The decision tree seems to be the least performed but the accuracy can be enhanced with the help of XGBoost Library.

**Experiment 2**

Objective:

To create targeted and non-targeted distortions to the MNIST dataset and record the accuracies as the test images get distorted.

Explanation:

Here, we have introduced distortions to all the MNIST digits in the dataset.

In targeted, we have distorted only a square of 18x18 pixels in the middle and in the non-targeted, we have distorted on an entire range of 784 pixels.

Below, we have tabulated the percentage of distortions vs the accuracy obtained on each model that was designed.

Tabulations:

Non-targeted distortions:

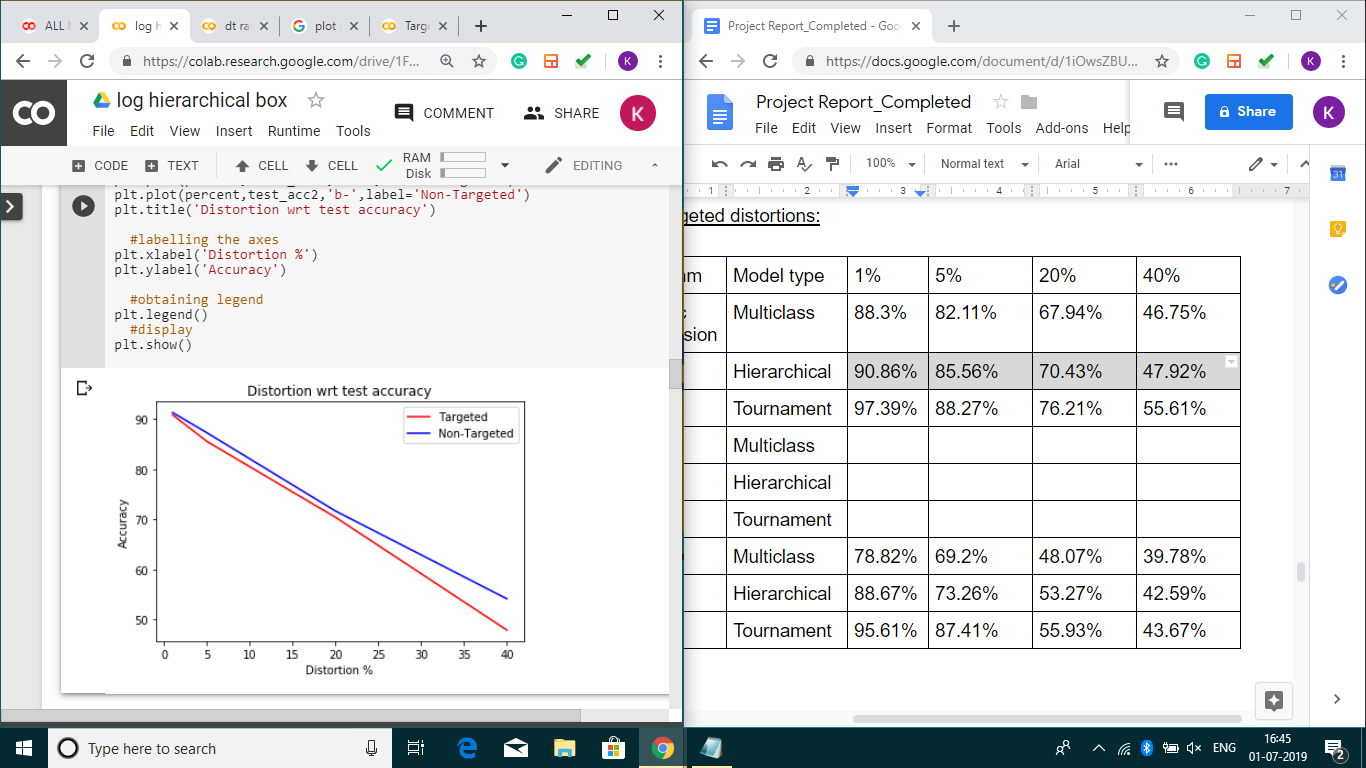
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Algorithm | Model type | 1% | 5% | 20% | 40% |
| Logistic Regression | Multiclass | 88.3% | 82.11% | 67.94% | 46.75% |
|  | Hierarchical | 90.86% | 85.56% | 70.43% | 47.92% |
|  | Tournament | 97.39% | 88.27% | 76.21% | 55.61% |
| DT | Multiclass | 78.82% | 69.2% | 48.07% | 39.78% |
|  | Hierarchical | 88.67% | 73.26% | 53.27% | 42.59% |
|  | Tournament | 95.61% | 87.41% | 55.93% | 43.67% |

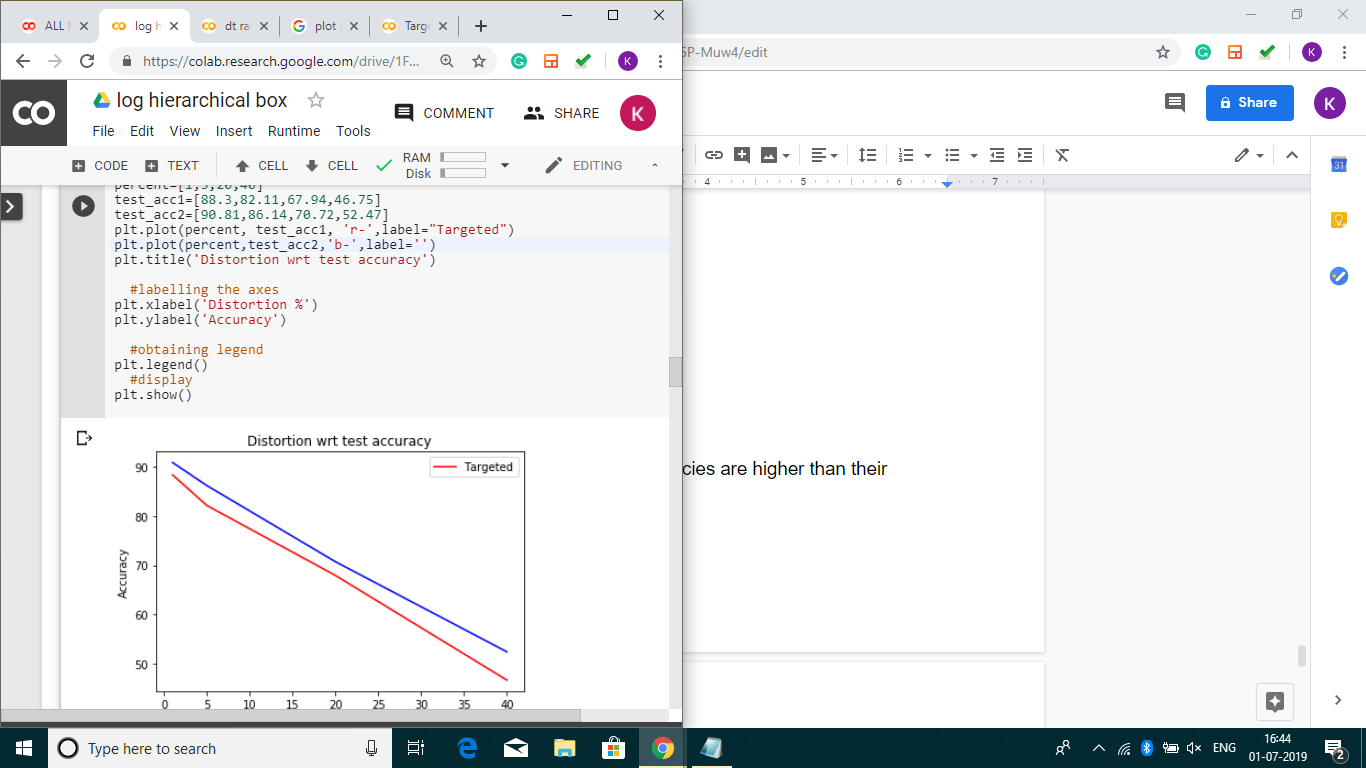
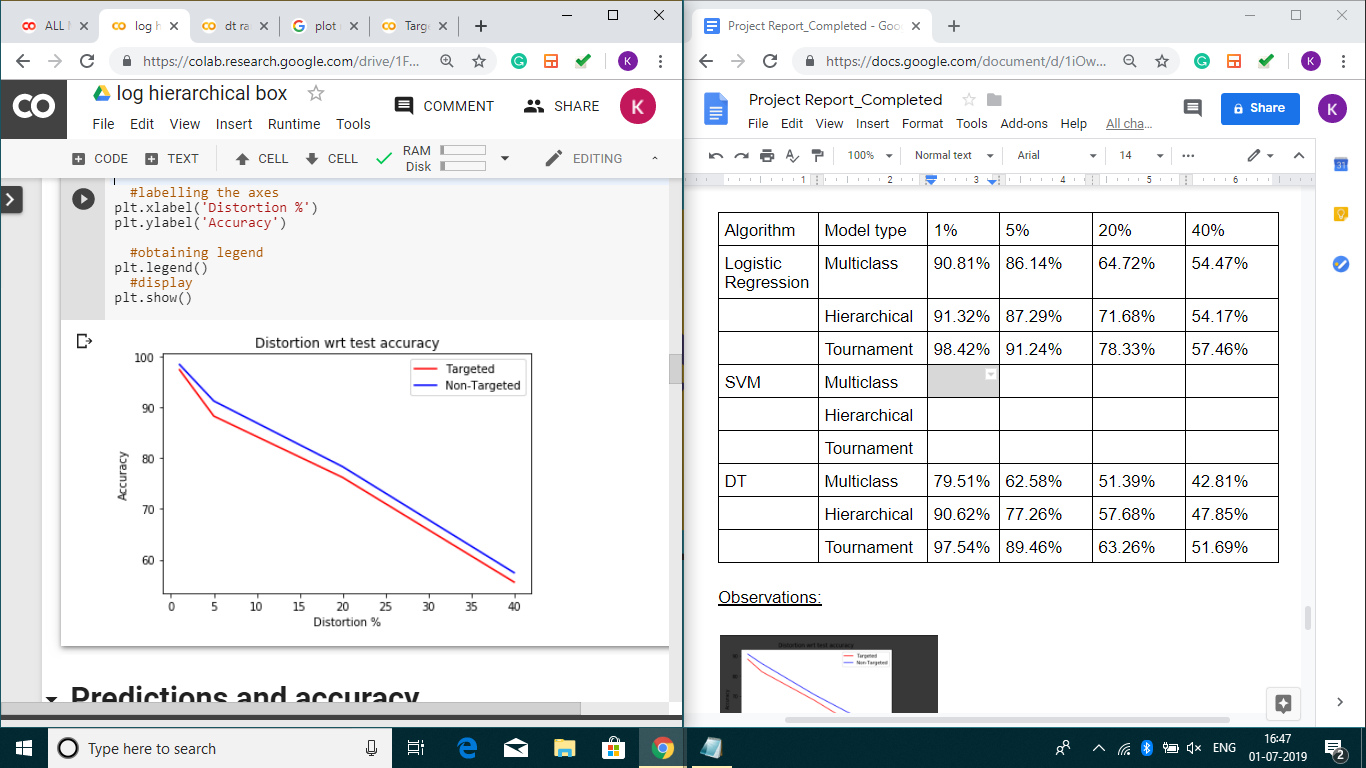
Targeted distortions:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Algorithm | Model type | 1% | 5% | 20% | 40% |
| Logistic Regression | Multiclass | 90.81% | 86.14% | 64.72% | 54.47% |
|  | Hierarchical | 91.32% | 87.29% | 71.68% | 54.17% |
|  | Tournament | 98.42% | 91.24% | 78.33% | 57.46% |
| DT | Multiclass | 79.51% | 62.58% | 51.39% | 42.81% |
|  | Hierarchical | 90.62% | 77.26% | 57.68% | 47.85% |
|  | Tournament | 97.54% | 89.46% | 63.26% | 51.69% |

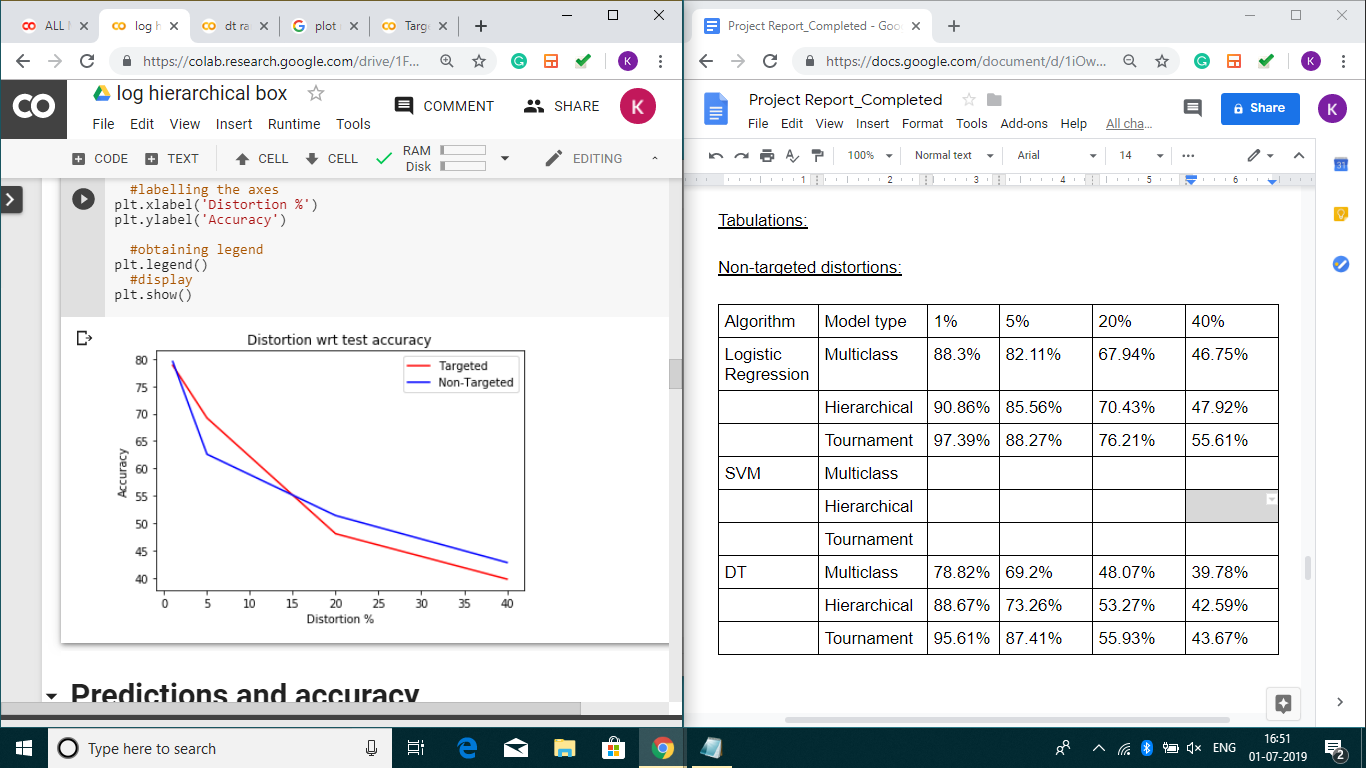
Observations:

Logistic Regression(Multiclass)

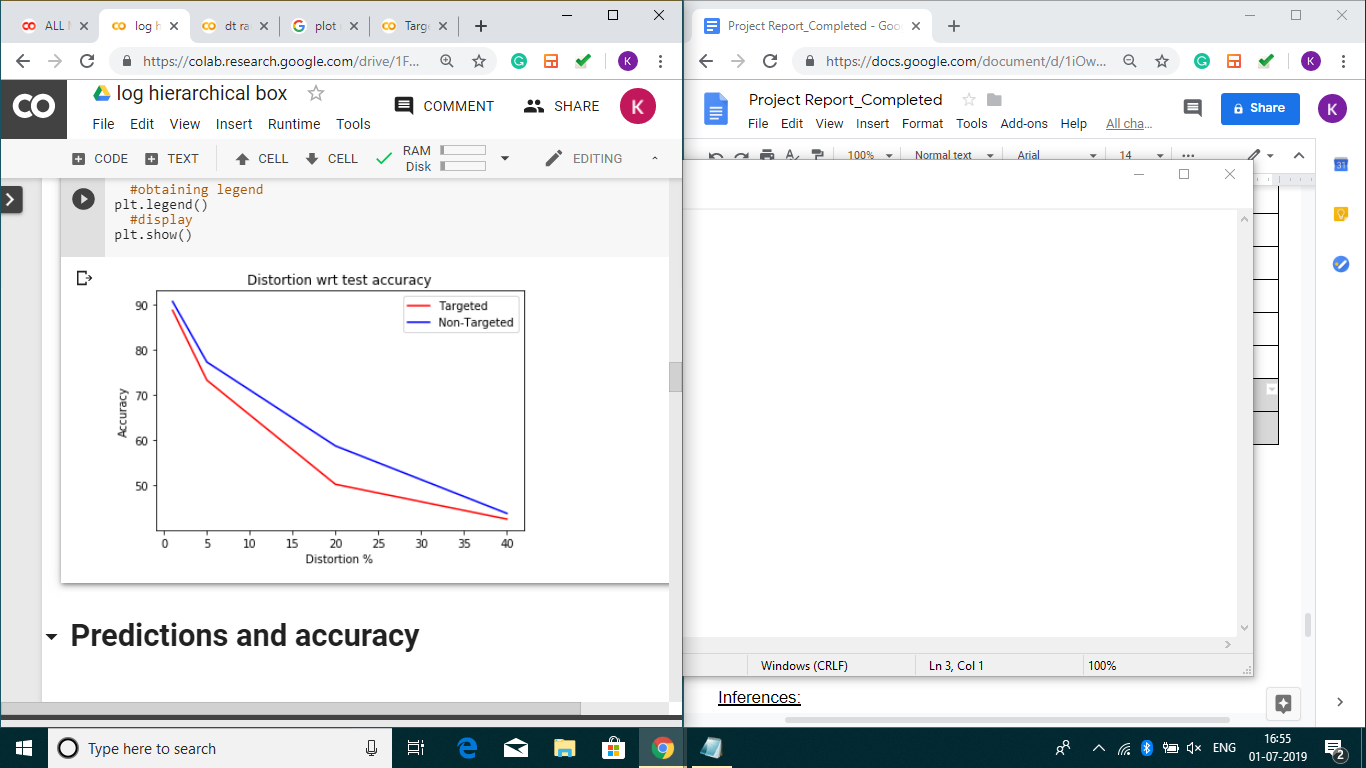
Logistic Regression(Hierarchical)

Logistic Regression(Tournament)

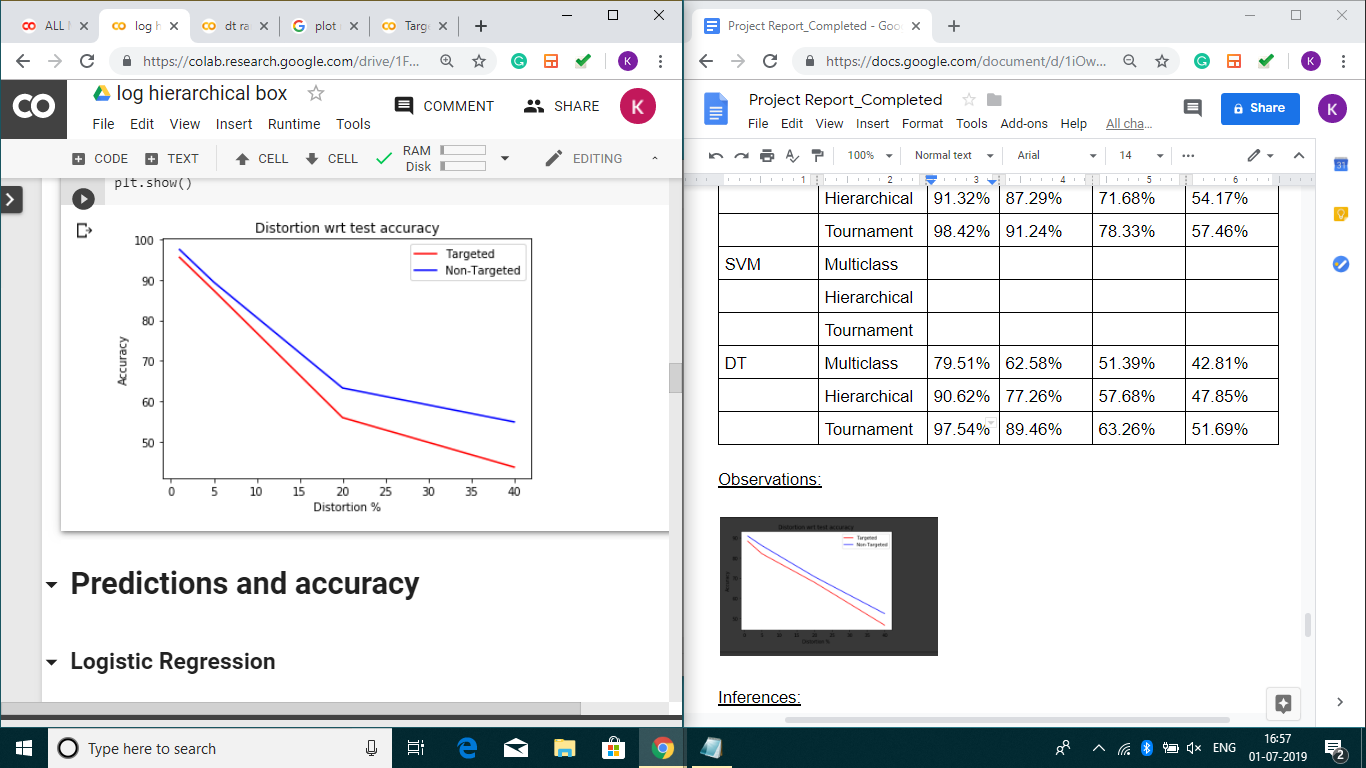
Decision Trees(Multiclass)



Decision Trees(Hierarchical)



Decision Trees(Tournament)



Inferences:

* We conclude that the targeted accuracies are higher than their non-targeted counterparts.
* The accuracy decreases as the percentage of distortions are increased.

Link to code: <https://colab.research.google.com/drive/1FvntJHFxa0EiMmCQ9jNNXSMAUqORpqqS>

The saved models are available at: <https://drive.google.com/drive/folders/1LVyaRWzCn6YHPsMhoMrDwKgY_Vlpk8NH?usp=sharing>