Benchmarks: A Citizens Scorecard on Judicial Accountability

CS 506 Final Project Report

By,

Hexuan Zhang

Tiam Moradi

Rahul Suresh

E Chengyuan

Under the guidance of,

CS 506 Lecturer :	Spark! Advisor :	Spark! Project Manager:
Professor Lance Galletti	Professor Maggie Mulvihill	John Merfeld

Fall 2019

The link to our repository on github

Index

1. Introduction

We are working with Benchmarks: A Citizens Scorecard on Judicial Accountability in Massachusetts, to comprehend the circumstances in which cases are reversed. Utilizing data analysis and various tools, our aim is to raise awareness of patterns to be discovered from judicial data. Once a judge is appointed on the bench, they are appointed for life; there is no form of accountability in place for certain actions. This tool will hopefully allow for more transparency of these actions and advocate for people to voice their criticisms of the state court system. This semester, our analysis revolves around civil cases that have been appealed to the appellate courts from 2009-2019; identifying potential patterns that can explain the logic of why a case may or may not be reversed.

2. Questions that aim to be answered

- Are we able to identify any patterns between cases via their opinions text?
- What proportion of cases are reversed in Massachusetts?
- What is the distribution among civil vs criminal for the cases that are reversed?
- Which judges have the highest reversed case rate and related analysis?
- What is the distribution among judges for the cases that are reversed?
- What is the distribution among courts for the cases that are reversed?
- Is it possible to predict when a case is going to be reversed?

3. Data Collection

We have two main sources for our data. The first source is masscases, an unofficial collection of reports, provided on the official Massachusetts government website. From this source, where we were able to collect public opinions text of the Supreme Judicial Court and the Appeals Court cases. Unlike the previous groups data source, this website was trivial to scrape, and we did not have problems gathering this information. Upon getting the data is when we ran into various issues. First and foremost, we weren't able to immediately identify important information via an Xpath: docket number, the lower court judge, and the verdict. This means that the only way to find this information was to manually look through these documents. To try to have consistent formatting as the previous group's data, we tried to utilize the previous group's notebook (can be found in our repo in the jupyter subdirectory) and attempted to scrape from the previous source. From there we ran into our second problem, which was extracting information from this source. The previous group's notebook would not work because the HTML styling has been updated since Spring 2018, and would otherwise extract the wrong information. After running the code for some time, we would eventually have a connection error,

meaning they would kick us out of the website. From the other group's meeting with John, they seemed to have increased their security measures as John was also facing issues extracting information. After not being able to collect data in the format of the previous group, the other group and ours decided to manually collect the lower court judge's name and the verdict of all new cases we scraped. This information was saved as a CSV file in the opinions_data_csv subdirectory. If we wanted to extract all 140 feature columns these previous groups managed to collect, this would have taken away the opportunity for us to analyze the data we possess.

Our second source is the data that the previous group had gathered, except we will only focus on the civil cases that were extracted. We have also put in the criminal cases that the previous team collected, and they are saved in the previous_gp subdirectory.

4. Data Analysis

- **Supreme Judicial Court, Appellate Court Data Analysis**

Questions that our analysis solves by Arthur and Helena.
What is the distribution among civil vs criminal for the cases that are reversed?
The proportion of cases are reversed and partially reversed in Massachusetts and the comparison throughout different time periods, and between Supreme Judicial Court and Appeals Court.
Which judges who have the highest reversed case rate and related analysis.

After cleaning the data from we had scraped, we had some overall idea about the reversal case ratio from the history data.

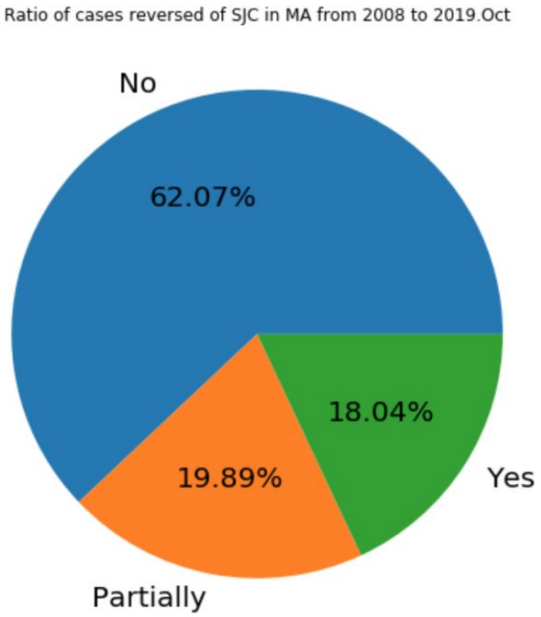


Figure 4.1

From 2008 to 2019. October in Massachusetts of Supreme Judicial Court (Figure 4.1), around 62% (468 cases) of the cases are affirmed, 38% (286 cases) are reversed (where 20% (150 cases) are partially reversed).

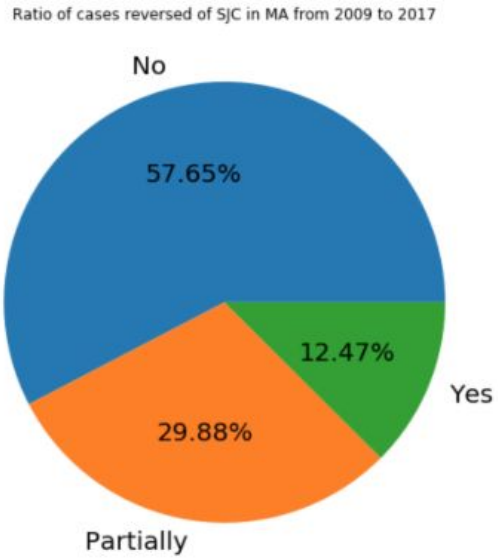


Figure 4.2

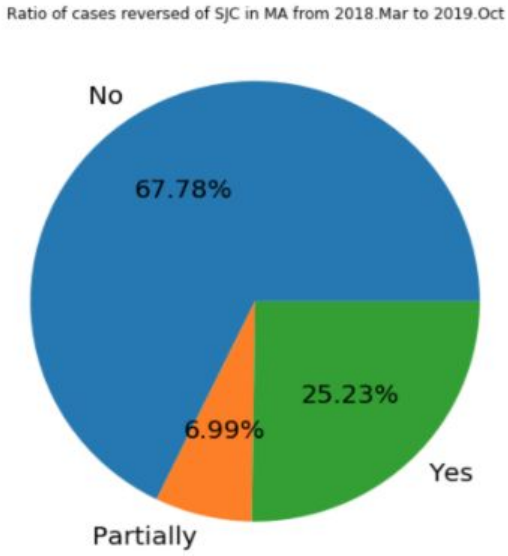


Figure 4.3

From 2008 to 2017 in Massachusetts of Supreme Judicial Court (Figure 4.2), around 58% of the cases are affirmed, 42.5% are reversed (where 30% are partially reversed). From March 2018 to October 2019 in Massachusetts of Supreme Judicial Court(Figure 4.3), around 68% of the cases are affirmed, 32% are reversed (where 7% are partially reversed).

Comparing the new cases(2018.3 - 2019.10) and the total cases, reversed cases are decreased by 6%, where partially reversed cases are decreased by 13%, fully reversed cases are increased by 7%. Comparing the old cases(2008 - 2017) and the new cases(2018.3 - 2019.10), reversed cases are decreased by 9.5%, where partially reversed cases are decreased by 23%, fully reversed cases are increased by 13%. The general trend is that cases are less to be reversed where the number of partially reversed cases drops significantly while the number of fully reversed cases increases.

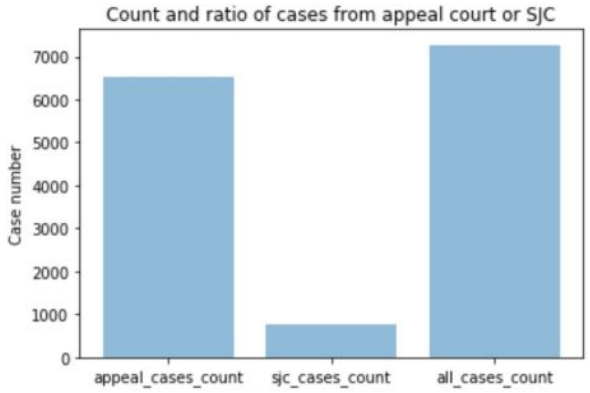


Figure 4.4

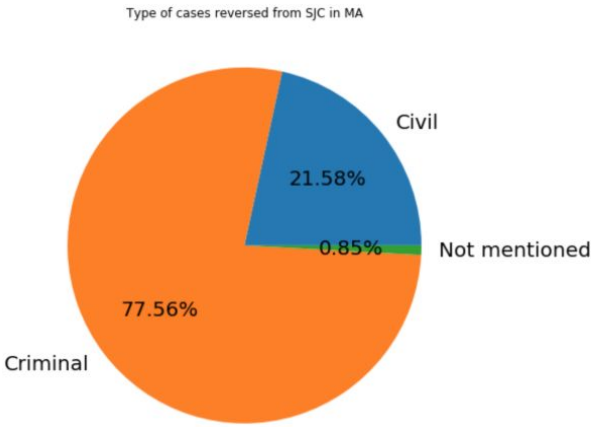


Figure 4.5

From 2008 to October 2019 in Massachusetts, among all the cases that are public (Figure 4.4), there are 7271 cases in total, where 6517 cases are Appeal Court cases, 754 cases are Supreme Judicial Court cases. The ratios are shown in the graph above. And 77.6% are criminal cases, 21.6% are civil cases (Figure 4.5).

- **Appeal Court Cases**

From March 2018 to October 2019 in Massachusetts's Appeal Court, around 57% of cases are civil type, 43% are criminal type.

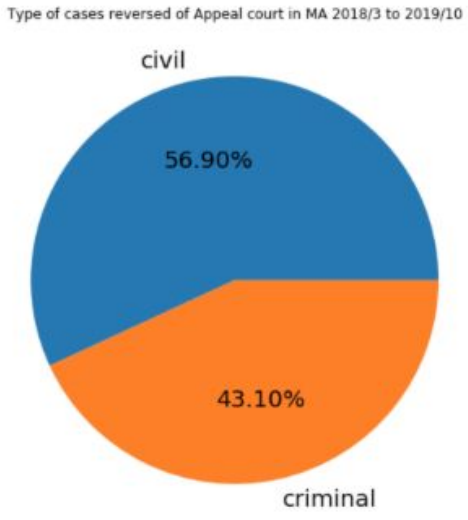


Figure 4.6

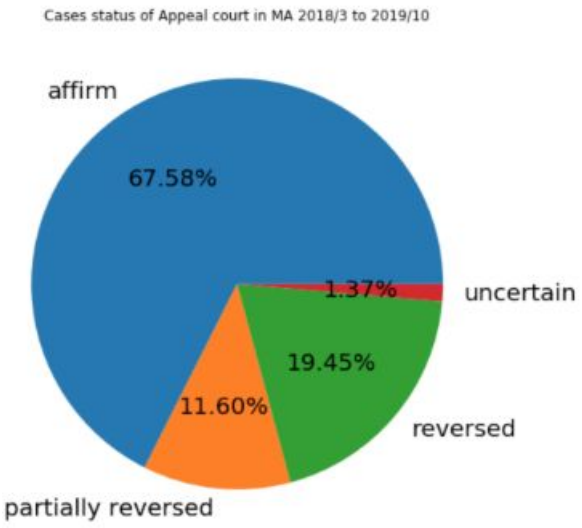


Figure 4.7

From both civil and criminal cases(Figure 4.7), around 67.6% cases are affirmed, 31% are reversed (where 11.6% are partially reversed), which is closed to the Supreme Judicial Court reversed rate. So the reversed cases rate between the Supreme Judicial Court and Appeal Court is no much difference. Take a closer look at Civil Cases, from March 2018 to October 2019 in Massachusetts from Appeal Court of only civil cases (Figure 4.6), around 67% cases are affirmed, 32% are reversed (where 15% are partially reversed).

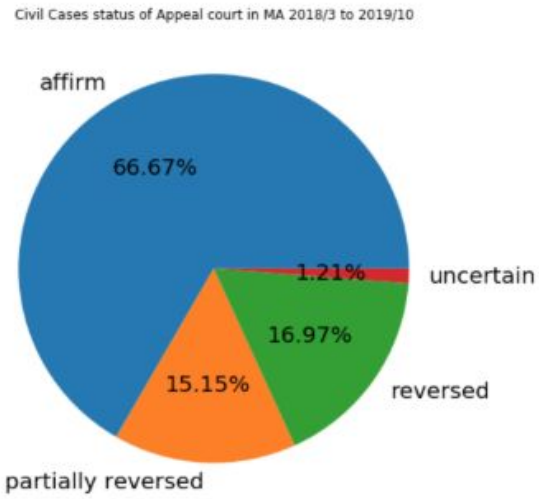


Figure 4.8

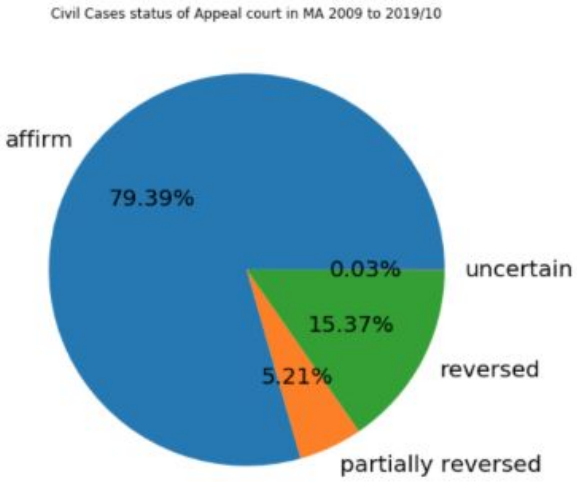


Figure 4.9

However, compared with all civil case data from 2009 to October 2019 (Figure 4.9), which had around 79.4% cases are affirmed, 20.5% are reversed (where 5.2% are partially reversed), the reversed rate increased more than 10% and this might worth to study further.

- **Judges**

Originally we believed that civil cases that had been reversed would not be common to a single judge. However, based on public data, the reversal ratio for a judge is higher than we expect.

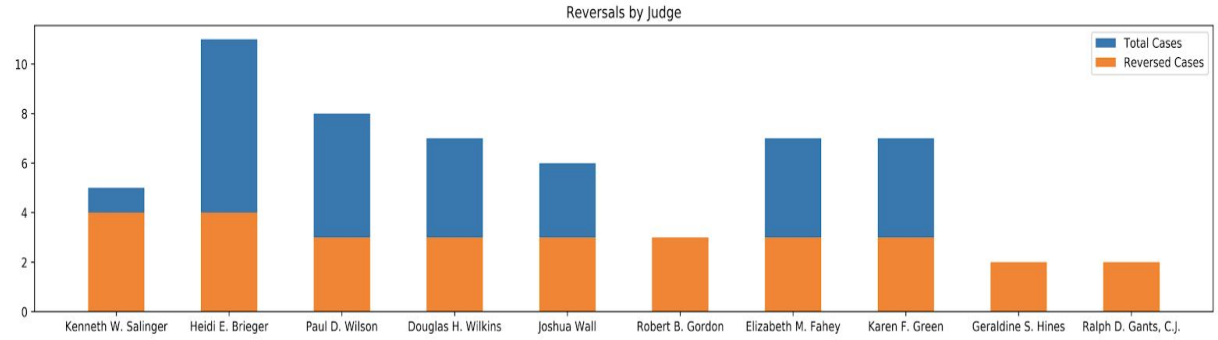


Figure 4.10

Figure 4.10 shows judges who presided over more than five cases and had more than 50% of cases they presided over reversed.

	all cases	all reversed cases	civil cases	reversed civil cases	reversed case rate	reversed civil case rate
Kenneth W. Salinger	5	4	2	1	0.800000	0.500000
Heidi E. Brieger	11	4	8	2	0.363636	0.250000
Paul D. Wilson	8	3	8	3	0.375000	0.375000
Douglas H. Wilkins	7	3	5	2	0.428571	0.400000
Joshua Wall	6	3	4	3	0.500000	0.750000
Robert B. Gordon	3	3	1	1	1.000000	1.000000
Elizabeth M. Fahey	7	3	2	1	0.428571	0.500000
Karen F. Green	7	3	6	2	0.428571	0.333333
Geraldine S. Hines	2	2	2	2	1.000000	1.000000
Ralph D. Gants, C.J.	2	2	1	1	1.000000	1.000000

Figure 4.11

The above graphs (Figure 4.11) shows the top 10 judges who have the most reversed cases from March 2018 to October 2019 of both the Supreme Judicial Court and Appeal Court in Massachusetts. Where Kenneth W. Salinger, Robert B. Gordon, Geraldine S.Hines, and Ralph D. Gants have the highest reversed case rate where the latter 3 judges reversed all cases they processed, also these cases are all civil cases. The average of reversed civil cases ratio is more than 50%

- **Opinions Text Analysis**

For this section of the analysis, we wanted to examine text data to possibly identify patterns between affirmed and reverse cases. Although we do not have enough data to utilize more advanced methods such as Deep Learning and Natural Language Processing, we valued the idea of attempting to using this data to gain further insight into these cases.

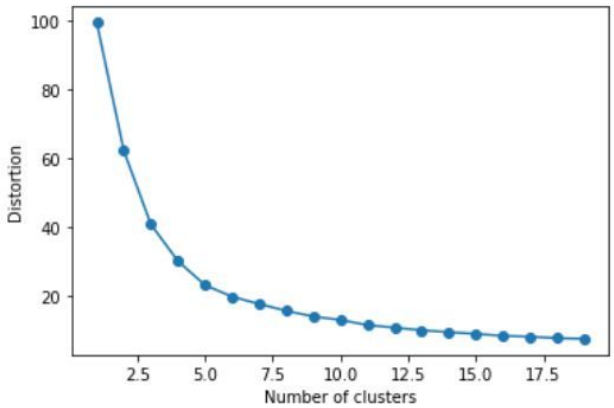
Our initial step was to apply the CountVectorizer and the TfidfVectorizer on the headnote and opinions text of our data; we wanted to see which terms would be selected as features for affirmed and reversed cases. We also separated the affirmed and reversed cases into separate data frames. This was done in order to compare the features of both types to cases without one type of case influencing another. We also wanted to examine how similar these features would be, or how many features would overlap for both data frames, if any at all. After separating the cases, we tested parameters for these feature extractors. For the analyzer, we wanted to inspect both words and ngram_ranges between the range of 2 and 3; we wanted to compare the performance of these tools to see if there would more unique features if these features where phrases instead of words in the two data frames. We also set the min_df to be 5 and max_df to be

.55; we chose these parameters because we didn't want unique words that only appeared to be apart of a few headnotes and opinions, as well as getting rid of words that appear then more than half the total documents. Finally, we also set the maximum number of features to be 50; without a restriction, vectorizers will add as many words as they could to a term-frequency that meets the criteria of our given parameters. This could include terms that are not necessary; for example within the data_reverse data frame, without the inclusion of max features, almost features were either years or numbers that were apart of the text that we extracted.

After fitting and transforming our data to these vectorizers we found that for every combination of settings (results can be found in opinions text analysis notebook) that affirmed and reversed cases shared over half of their features: the lowest being 58 percent and the highest being 68 percent of features were the same. From this, we can conclude that all cases tend to have similar terminology throughout the opinions. Thinking further this makes sense because works like "Defendant" and "Appeal" may appear in both types of cases, regardless of the outcome of the case.

To investigate this further we wanted to see if clustering to use these features to group similar cases together. We utilized KMeans++ as our clustering algorithm. We picked k equal to 5 because of our elbow plot. Originally, our data had a distortion value of over 100, which directed us to try Principal Component Analysis to see if it could capture the variance of our data. From reducing our dimensionality from 50 to 3, we replotted the elbow plot, and our distortion was 23, which was a significant improvement from using the original data. Further, reducing our dimensions would allow us to visualize our information properly.

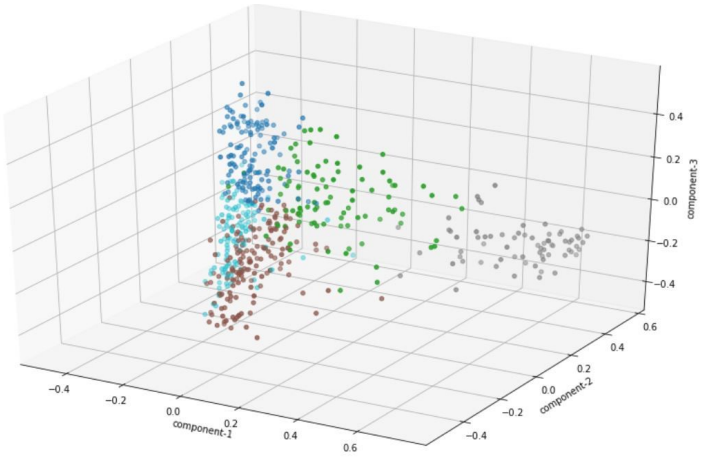
While the distortion is low, it can be a little misleading; since all of our data is sparse and the maximum value can be 1, the sum of squared distances will always be small for an individual, meaning that not all, but some clusters contain a high variance. Here is the 3-dimensional plot of our clustering.



From the plot, we can see that the Grey and Green clusters are very spread out and that it is hard to distinguish between Blue, Turquoise, Brown, and Green. Here is also the information about the clusters:

- cluster 1: 153 opinions
  - 112 criminal cases and 39 civil cases
  - 101 affirmed cases and 52 reversed cases
- cluster 2: 96 opinions
  - 85 criminal cases and 11 civil cases
  - 75 affirmed and 21 reversed
- cluster 3: 131 opinions
  - 116 criminal cases and 15 civil cases
  - 90 affirmed and 41 reversed
- cluster 4: 69 opinions
  - 68 criminal and 1 civil
  - 56 affirm and 13 reverse
- cluster 5: 112 opinions
  - 103 criminal cases and 9 civil cases
  - 100 affirmed and 12 reversed.

Besides the last two clusters being able to focus towards criminal instead of civil cases, the other clusters have slightly more criminal than civil cases, and more affirmed than reversed,



however, this doesn't indicate significant pattern recognition, as there are more criminal than civil cases, and for affirmed than reverse cases. Although we worked on a subset of opinions data, particularly the new cases we manually looked up, there is not enough present with the clustering to be confident in finding patterns between certain cases based on text features alone.

5. Predicting reversal using pattern analysis

- **Data preprocessing**

Given the nature of the data-heavy preprocessing was required to get any data in a useful form. Please refer *cleaning_prev_teams_data.ipynb* for a full picture.

- **Results**

The prediction labels were very skewed. The below represents the label distribution for *processed_cases.csv*.

Category of case	**Integer Class representation**	**Count**	**Count % (rounded to one decimal)**
Affirmed	0	10014	85.9 %
Reversed	1	1104	9.5 %
Partially Reversed	2	534	4.6 %

Table 1 : Class wise distribution of data samples

Given the skewed distribution of the classes, it was clear that accuracy was not the right metric. This was because even if the model predicted all samples as affirmed it would get an accuracy of 85.9 %. Thus we used the confusion matrix to evaluate the performance of models. As a more representative metric, we defined class-wise accuracy as,

Accuracy Class(i) = (number of correct predictions for class i) / (total number of examples for class i)

Equation 1: Class wise accuracy

An 80% train test split was used to obtain a split as follows,

Split	**Number of data points**

Train	9321
Test	2331

Table 2 : Train test split of model

First a baseline had to be set to determine progress. Two different models were run for this purpose,

- Random predictor - predict random values (uniformly) for each class
- Stratified predictor - predict classes as per class distribution of dataset

The results were as follows:

Model	**Confusion Matrix**	**Class Wise accuracy**
Random Guesser	[[680, 662, 668], [89, 73, 58], [36, 35, 30]]	Class 0: 0.338 Class 1: 0.332 Class 2: 0.297
Stratified Guesser	[[1718, 193, 99], [196, 19, 5], [81, 13, 7]]	Class 0: 0.855 Class, 1: 0.086 Class 2: 0.069

Table 3 : Baseline Metrics

The random guesser was chosen as the baseline to beat.

To mitigate the effect of the skewed distribution, oversampling and undersampling was thought of. Below is the class-wise distribution for the normal dataset, oversampled dataset and undersampled dataset.

Class	**Normal**	**ADASYN oversampling**	**Random Undersampling**

0	8004	8004	433
1	884	8004	433
2	433	8004	433

Table 4: Class wise distribution for different sampling techniques

A logistic regression model was applied to the dataset to check the effects of the different sampling techniques. The results were as follows:

Sampling technique (Model as logistic regression classifier)	**Confusion Matrix**	**Class Wise accuracy**
Normal	[[2010, 0, 0], [220, 0, 0], [101, 0, 0]]	Class 0: 1 Class 1: 0 Class 2: 0
Oversampling	[[59, 136, 1815], [3, 35, 182], [3, 22, 76]]	Class 0: 0.029 Class, 1: 0.159 Class 2: 0.752
Undersampling	[[9, 0, 2001], [4, 0, 216], [2, 0, 99]]	Class 0: 0.004 Class 1: 0.0 Class 2: 0.980

Table 5 : Sampling strategy comparison

This shows that among the different sampling techniques, oversampling turned out to be best. But on the whole even with oversampling the model was shown to have very poor performance as compared to the random guesser baseline. To eliminate the factor of a weak classifier as a source for poor performance more complex models were fit with the following results.

Model	**Confusion Matrix**	**Class Wise accuracy**

Decision Tree (DT)	[[1642, 214, 154], [161, 45, 14], [77, 13, 11]]	Class 0: 0.817 Class 1: 0.205 Class 2: 0.109
Gradient Boosted Decision Tree (GBDT)	[[1725, 196, 89], [148, 66, 6], [81, 12, 8]]	Class 0: 0.858 Class 1: 0.3 Class 2: 0.079

Table 6 :Model Comparison

While the results were better than that of the linear regression model, it still doesn't unequivocally beat the baseline random guesser. At this point the result is clear and as discussed in the next section.

To check the effects of adding more features, *text_processed_cases.csv* is used. This has all the features of the *processed_cases.csv* plus extra text features from the 'Nature' and 'Sub-Nature' columns. But the catch is that since these columns contain a lot of NULL values which can not be filled in suitably we are having to drop all rows which have NULLs in these columns as a result the size of the dataset decreases to 4369 examples only. Nonetheless, ignoring the possibility of potential overfitting, a DT was fit on it with the following results.

Model	Confusion Matrix	Class Wise accuracy
Decision Tree (DT)	[[733, 34, 22], [49, 5, 0], [30, 0, 1]]	Class 0: 0.929 Class 1: 0.0926 Class 2: 0.0323

Table 7 : Results of DT using text features

These results show that adding text features has detrimental to almost no effect on the model. This again confirms the point brought up by Table 6 and is discussed in the next section.

- **Discussion**

The discussion based on the model results all point to the same thing, which is that better data is needed. This can be tackled in 3 steps.

1. Better data needs to be scraped off the web -

Features like case text and headnotes would have been useful but were not available in the dataset we had. Scraping new data was not an option too as the websites kept blocking us off. This is a problem that is hard to fix. Combined with the fact that only 10 % of cases are published online this gives us very little data to work with. One fix at this stage is to work with the courts to get better data but this would undoubtedly be hard.

2. Transforming or scraping targeted features -

Extensive legal knowledge would be required to better use the features and come up with a better hypothesis. This better hypothesis would give rise to specific features that would have to be scrapped to better fit models. In simple terms, this would mean looking for specific features of a case in maybe the docket notes and creating new columns to represent that.

3. Getting better results from the data we have -

This would involve using all useful features that the dataset has. In my analysis, certain columns were dropped because of the extensive legal background required to process it. So maybe working tightly with legal experts could be of use. Also handling NULL values in hard. Many columns have over 90% NULL values. While these features could potentially be useful, we are not able to use it currently because of the absence of a clear direction to fill NULL values with. A good example would be the ‘Brief Status’ column. In the dataset we have the correlation matrix for the features we were able to extract is as given below -

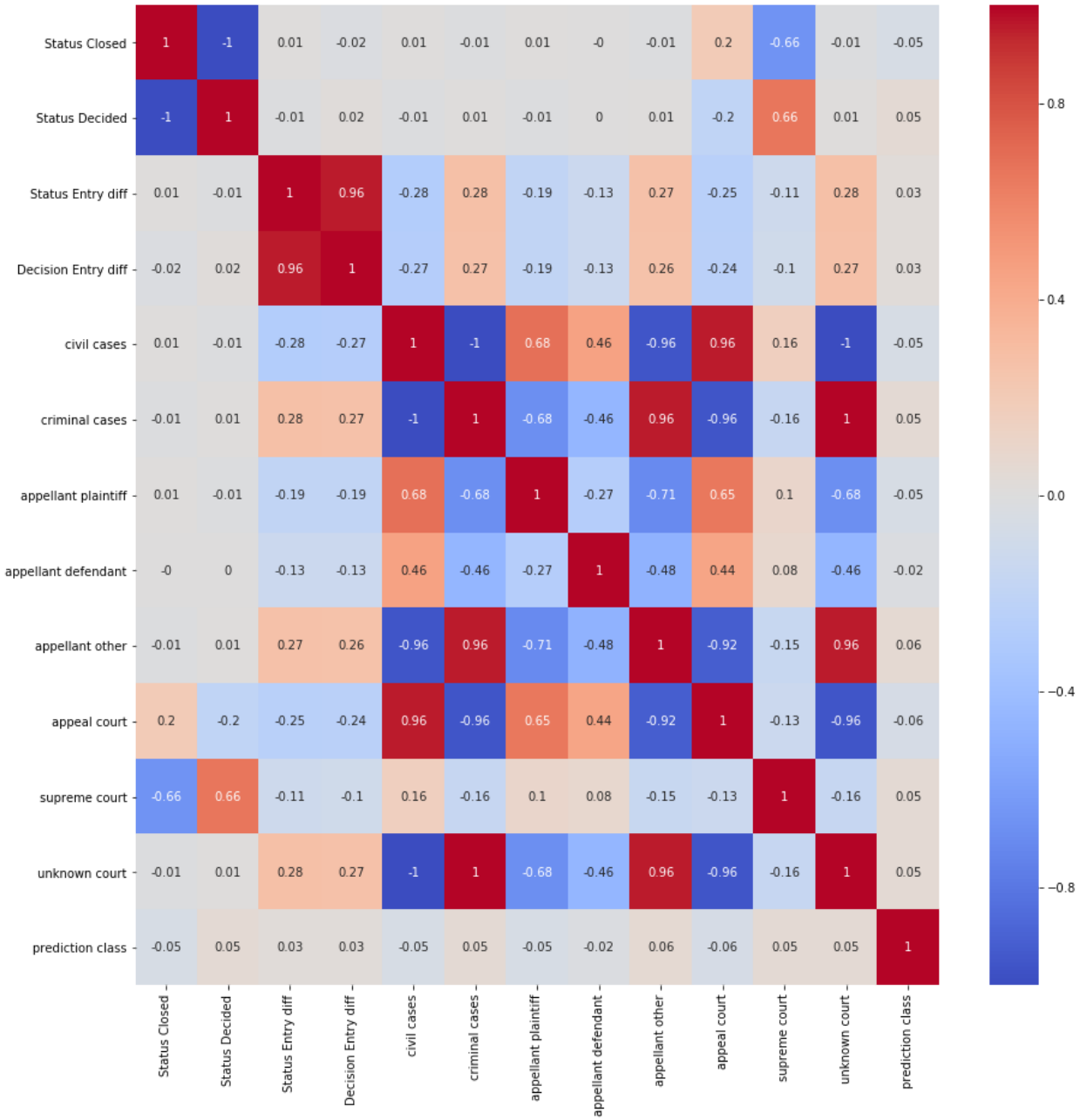


Figure 1: Feature Correlation matrix

This shows the low correlation of the assumed dependent variable on the independent variables which is bad. Also, a few of the independent variables (assumed) seem to be correlated. This would have to be fixed too. The feature importance as given by the GBDT is as shown below -

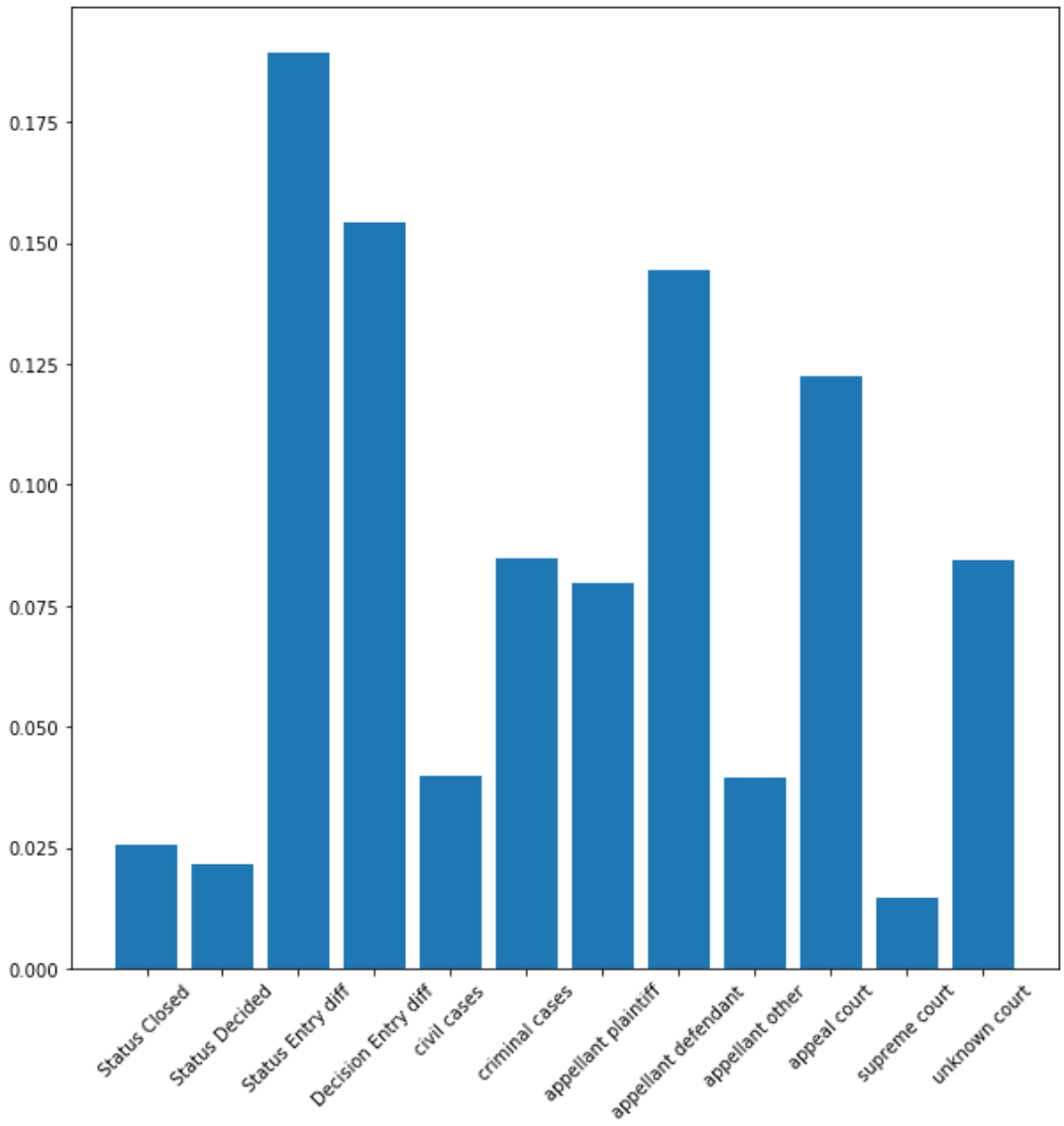


Figure 2: Feature importances

The top 3 features turn out to be-

- Difference between Status date and Entry date (‘Status Entry Diff’)
- Difference between Decision date and Entry date (‘Decision Entry Diff’)
- Appellant Defendant

On the whole, I must emphasize that steps 1 and step 2 are certainly more important than step 3 when it comes to the long term objective. The current data is not descriptive enough to discover any pattern to differentiate when a case is going to be reversed and when it isn't.

6. Future Steps

Based on the increase in the difficulty of collecting information from the government website, we believe that Benchmarks need to discover other methods for obtaining this information; what previous group and we provided are a good foundation, but limited to utilize for pattern recognition with machine learning algorithms. These algorithms rely on a lot of information to be successful for a given task, and the majority of analysis that revolves with these algorithms were not able to provide much insight with the data.

The next group can try to get older data from either website, however, they would need to investigate if this is appropriate. Primarily, they would need to be able to find a to take into account differences in the law, as what is legal today may not have been a few years ago. If there is no way to take into account the changes of the law, these cases could negatively alter a machine learning algorithm for finding patterns for modern cases or predictions if a case will be reversed or not. If the next group still wants to attempt to scrape data, they can try to use a VPN as soon as the project is assigned, but due to extra security measures in place, we cannot say with certainty that it will resolve the connection errors or not allowing requests.

Another avenue the next group can do is do a geographical visualization to see which courts are getting reversed more frequently compared to others, and what type of cases each court handles. Finally, if there is enough data collected, the next group can attempt to use anomaly detection algorithms to hopefully find more insight into the data, as well as other robust clustering measures that can handle the variance of the data. Finally, the suggestions under the discussion section should be used to ensure better prediction models.