

STATISTICS

1. Purpose of Statistical Analysis:

As quoted in the project proposal ...

“In the second stage, the project will apply off-the-shelf statistical and probabilistic machine learning algorithms in order to identify the relevant factors and quantify their impact on the sentencing decision-making process.”

Put simply, the statistical analysis is to identify if and how various aggravating and mitigating factors affect the sentence decisions made by judges.

This phrasing suggests several concepts:

- a. There will be various aggravating and mitigating factors
- b. Factors need to be analysed in order to identify if they are relevant to sentencing outcomes
- c. The effect on sentencing outcome the relevant factors do have must be quantified

When framing the question like this, it becomes apparent that the sentencing outcome is a dependent variable, and the various aggravating and mitigating factors are independent variables. The purpose of this investigation is to deduce which of the independent variables have a statistically significant effect on the dependent variable, and of the ones that are statistically significant, what is the extent of their influence on the dependent variable?

2. Analysis Methodology:

There are various statistical tests which can investigate this type of problem, but an appropriate methodology cannot be used until the nature of the dataset is known and various questions answered. For example, the dependent variable, “sentencing outcome” can take on many forms, which are all equally valid but predict to different degrees of accuracy.

If the possible values for “sentencing outcome” are ...

- Imprisonment
- Community Corrections Order

... Then “sentencing outcome” can be considered a qualitative or categorical variable which means only certain statistical analysis techniques are applicable.

If the possible values for “sentencing outcome” are ...

- 100 days in prison
- 5 years in prison

... Then sentencing outcome can be considered a quantitative variable which means other types of statistical analysis techniques are applicable.

If the possible values for sentencing outcome are ...

- 2 years imprisonment and 5 years community correction order
- 6 months of community corrections order and financial penalty

... Then sentencing outcome is a very complex dependent variable encompassing both a qualitative component (Prison, CCO, Fine) and a quantitative component (2 years imprisonment, 5 years CCO, \$50,000 fine)

Equally as important to the dependent variable is how the independent variable(s) are measured. An example of an aggravating factor in a legal case is “Showing No Remorse”. Like above, this variable’s value could theoretically take many forms:

- Categories of “Showing No Remorse”: Example: “**Large amount of** showing no remorse” and “**Small amount of** showing no remorse”,
- Remorse scale, such as “0 = **Showing no remorse** a little bit” and “100 = **Showing no remorse** a lot”
- Binary category, such as the “**party showed** no remorse”, or the “**party did not show** no remorse”?

Another key consideration of the independent variables is what constitutes an aggravating or mitigating factor?

These types of questions would have been ideally put towards legal experts to provide insight into how these aggravating and mitigating factors are viewed and interpreted by legal professionals. This insight would have guided how the statistical inquiry structured its questions / research and allowed the team to develop methods and concepts which better suit the legal professionals which would be interested in such research

As this was entirely absent, the team had to come to these decisions on their own. After examining many of the court documents and conducting some in-team research we decided on the following points: The analysis would only be looking at sentencing decision which encompassed imprisonment for a specified length of time. When looking at the court documents, majority of the cases included a prison sentence and as such we used this as the foundation of our decision.

We also decided to use a binary categorisation of the aggravating and mitigating factors meaning we viewed them as either being present or not present. The reason for this is that the court documents themselves only reference whether these factors were present or not, and the description of this part of the project stated ...

“quantify their impact on the sentencing decision-making process.” Suggesting that these factors have not formally been investigated and quantified yet.

3. Difficulties and Problems:

Before starting the statistical analysis, there were 2 clear major problems that the team faced:

- a. Task Dependency: The data which would act as an input to this statistical analysis is the output of the NLP classifier (Reminder: The Task of the NLP classifier is to automatically read court documents and identify the aggravating and mitigating factors as well as the sentencing outcomes). This task proved to be incredibly complex and as such, proper statistical investigation could not be done until this task was successfully completed.
- b. Lack of legal expertise: The project team had no legal experts involved in the development of the statistical analysis. Input from legal experts could have helped via clarifying concepts such as the ones described above and many other nuanced characteristics of the legal system.

Given that we had a large task dependency in the form of the NLP classifier’s output being the input for the statistical analysis, but developments in the statistical analysis portion had to take place, the project team generated a dataset in order to test how this process could work. Unfortunately, the team had no legal knowledge and had to make some assumptions.

Assumption:

- Assumption 1: There exist some base line for a crime.
- Assumption 2: The aggravating and mitigating factors act as modifiers to this baseline either increasing or decreasing the final sentencing outcome respectively.
- Assumption 3: The aggravating and mitigating factors are categorical in nature and have two levels “Factor Present” and “Factor NOT Present”

Furthermore, as there was no input from the client and / or legal expert stake holders, the level of accuracy / type of sentencing outcome to analyse was not clear.

4. Initial Investigation:

The initial dataset provided was generated by python script found in Appendix A.

General:

The dataset is comprised of 21 variables.

The dependent variable is “Sentence Length (days)”

The independent variables represent the type of crime committed and mitigating / aggravating factors.

Given this dataset, we have:

- 1 DEPENDENT VARIABLE that is NUMERIC – RATIO
- 20 INDEPENDENT VAIRABLES that are CATEGORICAL – 2 Levels

A summary table of the dataset:

<u>Variable Name:</u>	<u>Variable Type:</u>	<u>Usage:</u>
Sentence Length (days)	Numeric – Ratio	DV
Assault with weapon or instrument	Categorical – 2 levels	IV
Theft of a motor vehicle	Categorical – 2 levels	IV
Trafficking in a non-commercial quantity of a drug of dependence	Categorical – 2 levels	IV
Trafficking in a large commercial quantity of a drug of dependence	Categorical – 2 levels	IV
Trafficking in a commercial quantity of a drug of dependence	Categorical – 2 levels	IV
Sentence Length (days)	Categorical – 2 levels	IV
Assault (Common Law)	Categorical – 2 levels	IV
Common assault	Categorical – 2 levels	IV
Gambling addiction	Categorical – 2 levels	IV
Theft of a firearm	Categorical – 2 levels	IV
Aggravated burglary	Categorical – 2 levels	IV
General deterrence	Categorical – 2 levels	IV
Aggravated burglary	Categorical – 2 levels	IV
General deterrence	Categorical – 2 levels	IV
Theft	Categorical – 2 levels	IV
No remorse	Categorical – 2 levels	IV
Incest	Categorical – 2 levels	IV
Specific deterrence	Categorical – 2 levels	IV
Community protection	Categorical – 2 levels	IV
Plead guilty	Categorical – 2 levels	IV
Remorse	Categorical – 2 levels	IV
Burglary	Categorical – 2 levels	IV

This dataset doesn’t take into consideration if the sentence type is prison, detention, community corrections order, etc.

A good way to move forward may be: 1 Crime variable with 12 factor levels for the different crimes.

Given that we have ...

- 1 DV (Numeric – Ratio)
- 20 IV's (Categorical – 2 levels)

... we are only able to conduct statistical analysis which accounts for this type of data.

The statistical tests available and what they accomplish are as follows (note: each one of these tests may have a unique set of assumptions that must be met in order to use them and have sensible output).

2 independent sample t-test: (1 DV and 1 IV - 2 levels only)

Description: Can be used to identify a Statistical difference between the means of two groups. In this context the two groups would be the two factor levels for any given IV. Assumes the DV is interval and normal.

Example:

- Show Remorse:
 - Group 1 = Yes showed remorse
 - Group 2 = No did not show remorse
- Plead guilty:
 - Group 1 = Yes, plead guilty
 - Group 2 = No, plead not guilty

This will allow us to answer the following question:

“If we look at ONE factor, and ignore all other factors and their potential interaction, is there a difference in the average sentence given when that factor is present or not?”

Problem:

- Ignores all factors except for one at a time therefore ignoring any interaction effects
- Would require 20 tests to be conducted (one for each IV)

Guide: <https://libguides.library.kent.edu/SPSS/IndependentTTest>

One-way ANOVA: (1 DV and 1 IV – 2/3 levels only)

Description: Same as 2 independent sample t-test: (1 DV and 1 IV - 2 levels only) but can account for factors with 3 levels as well

Assumes DV is interval & normal

Example: Same as 2 independent sample t-test

Problem: Same as 2 independent sample t-test

Guide: <https://libguides.library.kent.edu/SPSS/OneWayANOVA>

Factorial ANOVA: (1 DV and 2+ IV's)

Description: Factorial ANOVA compares means across two or more independent variables. A one-way ANOVA has one independent variable that splits the sample into two or more groups, whereas the factorial ANOVA has two or more independent variables that split the sample in four or more groups

Example:

- Sentence Duration explained through the presence or not of each factor

This will allow us to answer the following types of questions: "Which factors influence the sentence given" or "How does each factor influence the sentence given?"

Problem:

- Should the different types of crimes be considered different types of factors?
 - If yes, we need a very large amount of data to have a balanced dataset
 - There needs to be observations for all combinations of factors
 - There needs to be several of each of these observations
 - If no, do we only analyse one crime at a time?
 - We would need a lot of observations for each type of crime
 - Each type of crime would also need observations for all combinations of factors
- We have a lot of IV's which can cause problems such as over fitting or not modelling the data well at all.

Guide: <https://www.statisticssolutions.com/conduct-interpret-factorial-anova/>

Other possible tests (only considers 1 DV and 1 IV)

- Wilcoxon-Mann Whitney test
- Kruskal Wallis
- Paired t-test
- Wilcoxon signed ranks test
- One-way repeated measures ANOVA
- Friedman test

Unsure of usefulness as they all pose the same issues as described in the independent samples t test, mainly that they can only take into consideration 1 IV at a time and do not look at the interaction between IV's

Using other statistical models:

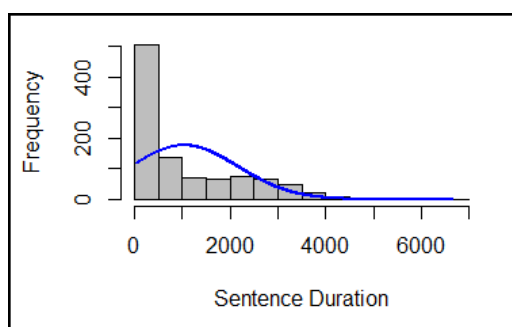
- Decision Tree
- Random Forrest
- Neural net
- Linear Models

Can be used for prediction purposes but unsure of usefulness in determining the individual influence of each factor on sentence duration.

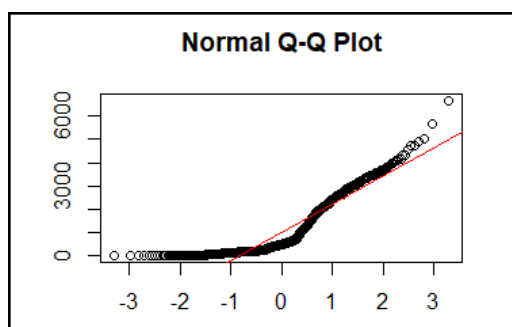
DATASET INVESTIGATION:

<u>Dependent Variable: SENTENCE DURATION IN DAYS</u>	
<u>Statistic:</u>	<u>Explanation</u>
Mean = 1037.568	The average length of sentence duration is 1037 days for any crime
Median = 491	50% of sentence durations are less than 491 days 50% of sentence durations are more than 491 days
Min = 17 Max = 6658 Range = 6641	The smallest sentence duration is 17 days The largest sentence duration is 6658 days This means in the given dataset; the sentences have a spread of 6641 days
Q1 = 188 Q3 = 1813 IQR = 1625	The lower 25% of sentence durations are less than 188 The upper 25% of sentence durations are above 1813 This means in the given dataset, 50% of the sentences have a spread of 1625 days
<u>This already suggests the possibility for some outliers at both ends of the spectrum for sentence duration.</u>	

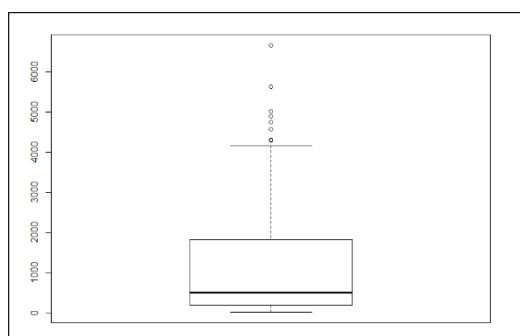
Histogram / QQ Plot / Box Plot of Sentence Duration:



This histogram shows a strong positive skew which means that the bulk of the sentence durations exist towards the lower end of the axis thus an unsymmetrical / non normally distributed dataset.



The QQ plot and QQ Line suggests a similar conclusion as the histogram showing a clear deviation from a normal distribution.



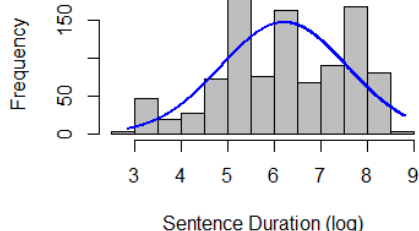
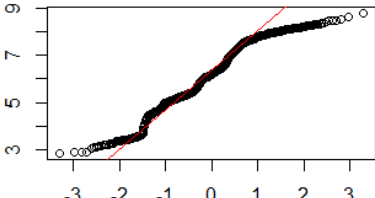
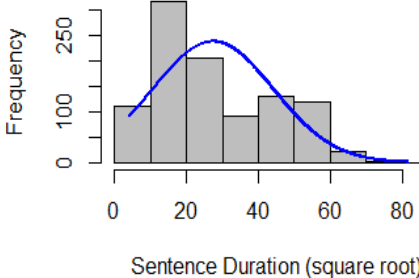
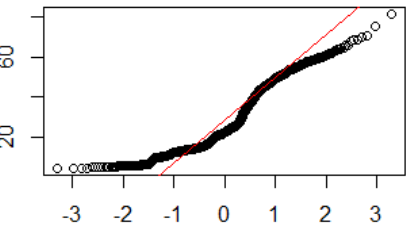
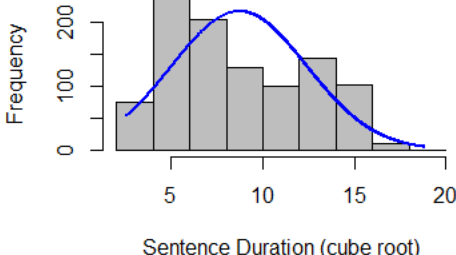
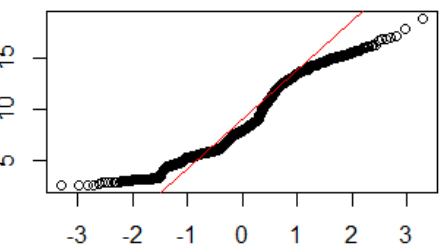
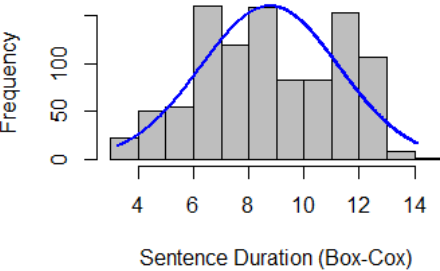
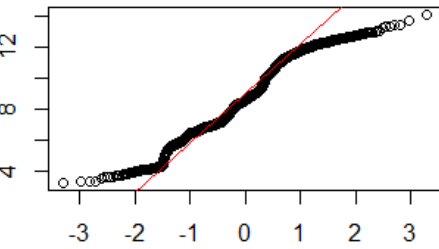
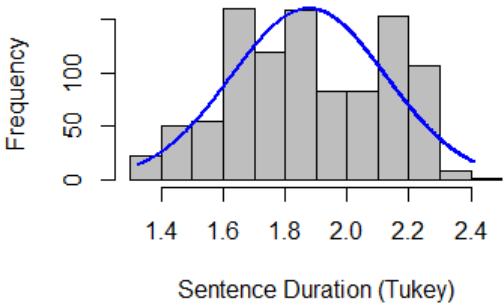
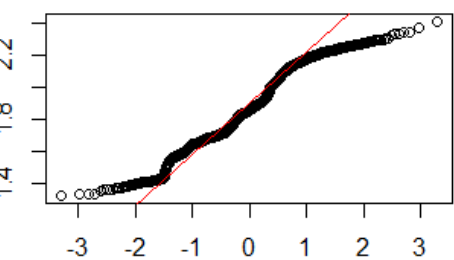
The boxplot finally confirms what the previous plots have shown, and that is a clear group of outliers which are influencing the distribution of the sentence duration heavily.

The boxplot has identified the following entries to be outliers (ID numbers: 147, 207, 279, 330, 730, 869, 925, 933, 935. All these observations have incredibly large sentence durations.

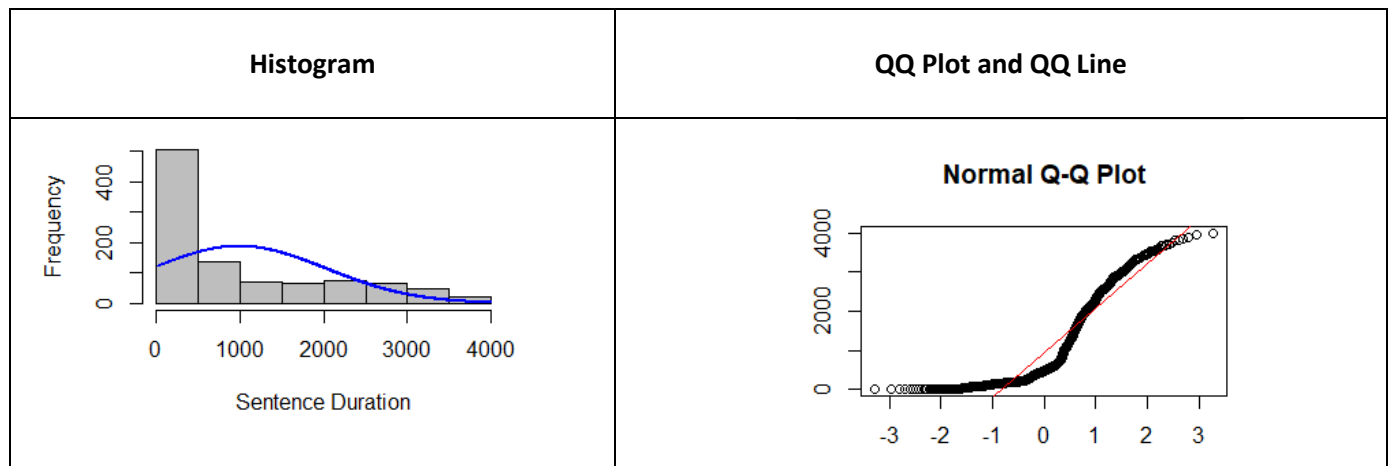
There are a few ways to deal with outliers:

- Include Outliers: This may lead to DV being too heavily skewed meaning certain statistical tests and models won't be applicable.
- Remove Outliers: This may lead to losing out on important information.
- Recalculate Outliers: Write a new value to all outliers to make them conform to a more normal distribution.
- Transformation All Data: Conducting mathematical operation on observations to normalise dataset.

Summary distributions of DV (Transformation used; Outliers **NOT** removed):

Transformation	Histogram	QQ Plot and QQ Line
Log		
Square Root		
Cube Root		
Box-Cox		
Tukey		

Summary distributions of DV (Raw Data - Outliers removed):



Summary distributions of DV (Outliers removed; Transformations used):

All distributions were similar with the outliers removed or not after transformations had taken place.

Summary of results of normalisation of DV:

Operation	Result
Transformations	Improvement but not close enough to normal for parametric testing
Raw Data – Removal of Outliers	Improvement but less than transformation
Removal of Outliers -> Transformations	Similar situation to only using transformations.

Box-Cox and Tukey show the closest to normal distribution although it poses the problem of interpreting information after transformations have taken place. When conducting post hoc on a model to determine its validity and results, the results will be in the transformed state making it harder to interpret in relation to real world numbers.

Factorial ANOVA: (1 DV and 2+ IV's)

Description: Factorial ANOVA allows for multiple categorical IV's and 1 continuous DV to be analysed and identify any effects the categorical IV's may have on the continuous DV.

Assumptions: Factorial ANOVA relies on several assumptions regarding the distribution of the dependent variable for each level of the independent variables

Reference:

- https://en.m.wikiversity.org/wiki/Analysis_of_variance/Assumptions
- <https://biostats.w.uib.no/test-for-homogeneity-of-variances-levenes-test/>

1. Normality of the DV distribution: The data in each cell should be approximately normally distributed. Check via histograms, skewness and kurtosis overall and for each cell (i.e. for each group for each DV). This would mean for each IV's factor level, check the distribution of sentence length duration for normality.
2. Homogeneity of variance: The variance in each cell should be similar. Check via Levene's test or other homogeneity of variance tests which are generally produced as part of the ANOVA statistical output.

3. Sample size: per cell > 20 is preferred; aids robustness to violation of the first two assumptions, and a larger sample size increases power
4. Independent observations: scores on one variable or for one group should not be dependent on another variable or group (usually guaranteed by the design of the study)

Does our data match the assumptions?

1. No - The DV is not normally distributed, removal of outliers and transformations bring it closer to normal but never within a good normal range and very far from perfect. Therefore, it is impossible for it to be normally distributed for each factor level.
2. No - The aggravating and mitigating factors have evenly distributed variances against the sentence duration but not the crimes themselves.
3. No – See problem section below for explanation
4. Yes – See script

Interpretation / Explanation of results

Conducting Factorial ANOVA to see what kind of results are obtained:

- Model with ALL factors AND interaction effect / crimes results in R / PC crash.
- Model with only FACTORS AND interaction effect results in an impossible to read output
- Model with only FACTORS AND main effect results in non sensical results. E.g.

```
# $`dataset$GamblingAddiction`
#   diff   lwr   upr   p adj
# 1-0 143.2826 -10.53658 297.1019 0.0678592
# Gambling addiction present increase sentence duration by 143 days

# $`dataset$PleaGuilty`
#   diff   lwr   upr   p adj
# 1-0 -135.9033 -291.2168 19.41013 0.0862703
# Pleading guilty reduces sentence duration by 135 days
```

When conducting the same analysis on a subset of the data only accounting for one type of crime there are no significant results.

Problem: Factorial ANOVA has a “Factorial Design” which describes all the ways the different factors can be combined to create unique conditions. For example:

- If there were 2 factors, each with 2 levels, the factorial design would be a “2*2” with a total of “4” conditions.
- If there were 2 factors, each with 3 levels, the factorial design would be a “3*3” with a total of “9” conditions.
- If there were 2 factors, one with 2 levels and one with 3 levels, the factorial design would be a “2*3” with a total of “6” conditions

The dataset we have has 20 IV’s all with 2 levels, thus the Factorial Design for a Factorial ANOVA which takes all IV’s into consideration is a “2*2*2*2*2*2*2*2*2*2*2*2*2*2*2*2*2*2*2*2” with a total of “1,048,576” conditions. This already poses a problem as this is an absurdly large number of unique conditions to have to consider and many observations of each condition would also be required.

If the dataset were split into 12 unique tables, each only account for 1 crime and the 12 remaining aggravating / mitigating factors, the factorial design would be a “2*2*2*2*2*2*2*2*2*2*2*2” with a total of “4096” conditions which is considerably better than previous, but still far too large when considering the dataset currently used.

Ways forward:

- Do not use ANOVA
- Create dataset with enough observations (May require very powerful computer to process)
- Reformat how we designate crimes / factors

<https://opentextbc.ca/researchmethods/chapter/multiple-independent-variables/>

5. Recreating the dataset and investigation:

As we can see in the above investigation, the initial generated dataset had one major problem: It would not generate a dataset which would be balanced for the purpose of factorial ANOVA. Due to this, our initial results were non sensical and thus changes to the sample dataset generator had to be made.

As initially we assumed base lines for different crimes would be different, we included a modified baseline for each type of crime but this lead to problem of there being so many unique combinations of factors being present or not, as well as which crime or crime(s) had been committed or not that the dataset requirements for a balanced design would be become too large for our computer / R Studio to manage and would lead to crashes.

A potential way forward would be to offload the calculations to an AWS node with the required packages loaded on it as this would allow the analysis code to executed more effectively. Alternative analysis techniques which are not so computationally intensive could also be investigated but as the dataset requirements are so large, some degree of computing power is required.

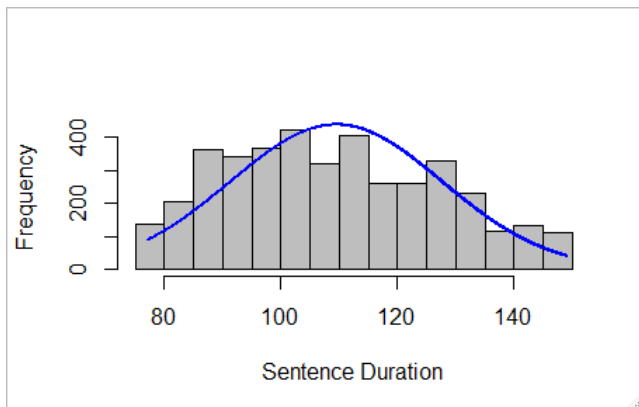
A new dataset generator was scripted an is provided in appendix b. This generator is slightly different to the original with a reduced scope but allowed for arbitrary amounts of factors to be included in the generation of the dataset.

This generator only generates data for 1 type of crime at a time, which is essentially specified by the base line sentence value. The various factors and their modifiers are then applied to the base line with random noise thrown into the formula and a dataset is generated. This dataset represents only one type of crime, for all combinations of factors being present or not, and can include any amount of factors / observations per combination.

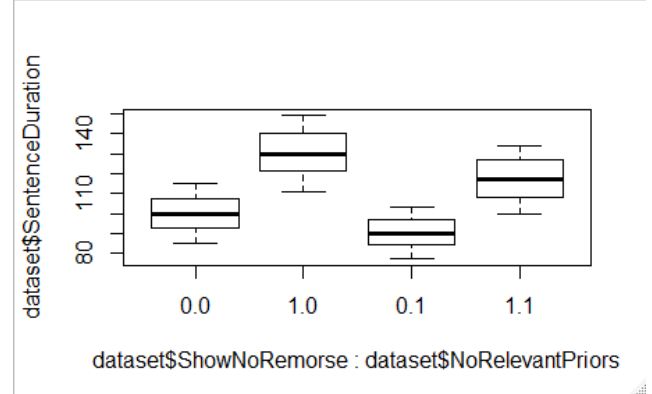
6. Investigation with new dataset:

2 Factors (1 Aggravating, 1 Mitigating): 1000 observation per unique combination, base line sentence = 100 days.

Distribution of DV:



Balance Design:



Results:

Anova Table (Type II tests)

Response: dataset\$SentenceDuration

	Sum Sq	Df	F value	Pr(>F)
dataset\$ShowNoRemorse	835383	1	9202.9	< 2.2e-16 ***
dataset\$NoRelevantPriors	132503	1	1459.7	< 2.2e-16 ***
Residuals	362823	3997		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Tukey multiple comparisons of means
95% family-wise confidence level

Fit: aov(formula = dataset\$SentenceDuration ~ dataset\$ShowNoRemorse + dataset\$NoRelevantPriors)

\$`dataset\$ShowNoRemorse`
diff lwr upr p adj
1-0 28.903 28.31231 29.49369 0

\$`dataset\$NoRelevantPriors`
diff lwr upr p adj
1-0 -11.511 -12.10169 -10.92031 0

We can see that the DV is normally distributed and that there is an equal amount of observations per unique combination. Both factors were significant in their effect on sentence duration, and when conducting post hoc testing, the following results were identified:

Showing no remorse = 28 days increased sentence duration

No relevant priors = 11 days reduced sentence duration

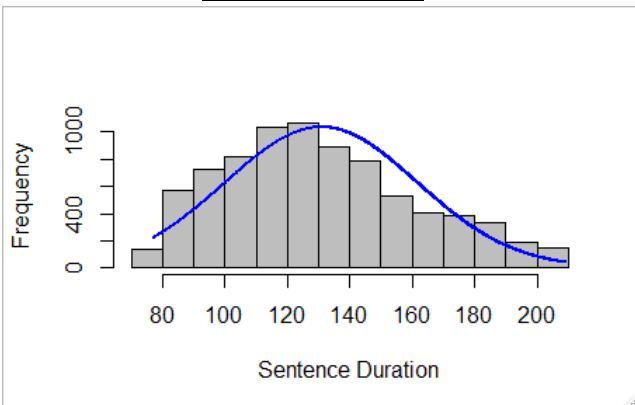
This lines up with the modifiers specified in the dataset generator:

Showing no remorse = 1.3, base line = 100 => $1.3 * 100 = 130$, therefore 30 day increase to be expected if showing no remorse.

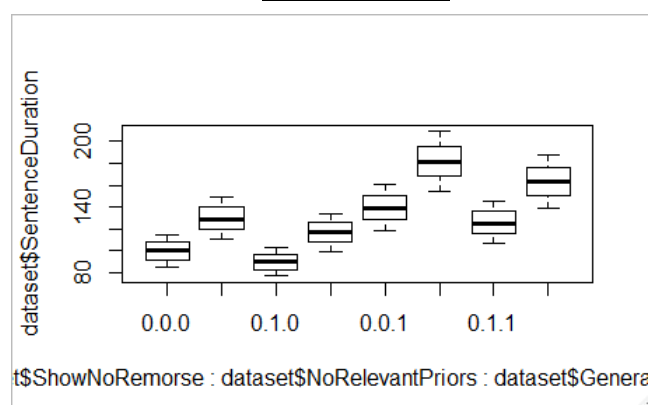
No Relevant Priors = 0.9, baseline = 100 => $0.9 * 100 = 90$, therefore 10 day decrease to be expected if having no relevant priors.

3 Factors (2 Aggravating, 1 Mitigating): 1000 observation per unique combination, base line sentence = 100 days.

Distribution of DV:



Balance Design:



Results:

Anova Table (Type II tests)

Response: dataset\$SentenceDuration

	Sum Sq	Df	F value	Pr(>F)
dataset\$ShowNoRemorse	2334734	1	15937.7	< 2.2e-16 ***
dataset\$NoRelevantPriors	371976	1	2539.2	< 2.2e-16 ***
dataset\$GeneralDeterrence	3759009	1	25660.3	< 2.2e-16 ***
Residuals	1171343	7996		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
$`dataset$ShowNoRemorse`
      diff      lwr      upr p adj
1-0 34.16675 33.63623 34.69727    0

$`dataset$NoRelevantPriors`
      diff      lwr      upr p adj
1-0 -13.63775 -14.16827 -13.10723    0

$`dataset$GeneralDeterrence`
      diff      lwr      upr p adj
1-0 43.35325 42.82273 43.88377    0
```

We can see that the DV is normally distributed and that there is an equal amount of observations per unique combination. Both factors were significant in their effect on sentence duration, and when conducting post hoc testing, the following results were identified:

Showing no remorse = 34 days increased sentence duration

No relevant priors = 13 days reduced sentence duration

General Deterrence = 43 days increases sentence duration

This lines up with the modifiers specified in the dataset generator:

Showing no remorse = 1.3, base line = 100 => $1.3 * 100 = 130$, therefore 30 day increase to be expected if showing no remorse.

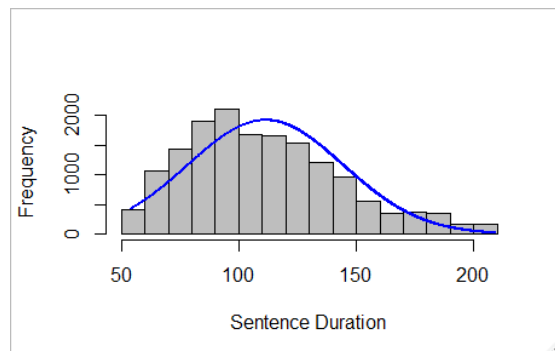
No Relevant Priors = 0.9, baseline = 100 => $0.9 * 100 = 90$, therefore 10 day decrease to be expected if having no relevant priors.

General Deterrence = 1.4, baseline = 100 => $1.4 * 100 = 140$, therefore 40 day increase to be expected if having no relevant priors.

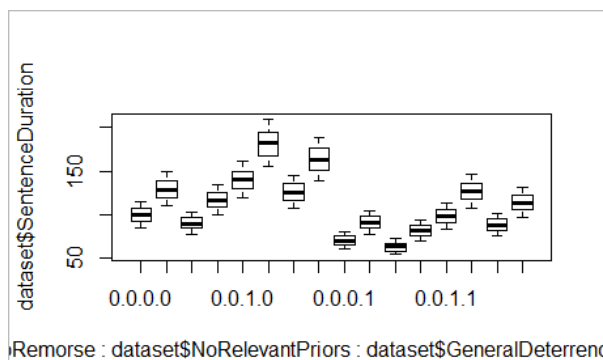
Worth noting: The accuracy of the first two factors has gone down with the introduction of a third factor.

4 Factors: (2 Aggravating, 2 Mitigating): 1000 observation per unique combination, base line sentence = 100 days.

Distribution of DV:



Balance Design:



Results:

Anova Table (Type II tests)

Response: dataset\$SentenceDuration

	Sum Sq	Df	F value	Pr(>F)
dataset\$ShowNoRemorse	3346275	1	26364.8	< 2.2e-16 ***
dataset\$NoRelevantPriors	543123	1	4279.2	< 2.2e-16 ***
dataset\$GeneralDeterrence	5524354	1	43525.6	< 2.2e-16 ***
dataset\$PleaGuilty	6217323	1	48985.4	< 2.2e-16 ***
Residuals	2030117	15995		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
$`dataset$ShowNoRemorse`
      diff      lwr      upr p adj
1-0 28.9235 28.57434 29.27266    0

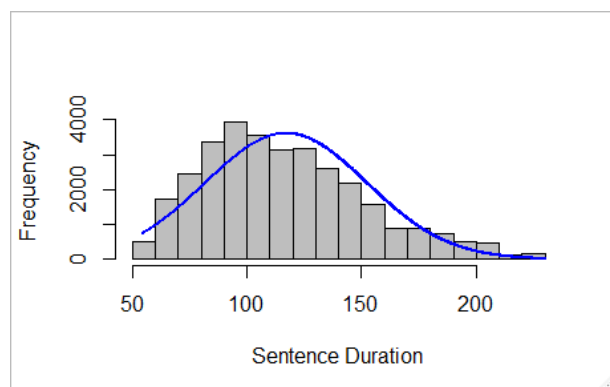
$`dataset$NoRelevantPriors`
      diff      lwr      upr p adj
1-0 -11.6525 -12.00166 -11.30334    0

$`dataset$GeneralDeterrence`
      diff      lwr      upr p adj
1-0 37.163 36.81384 37.51216    0

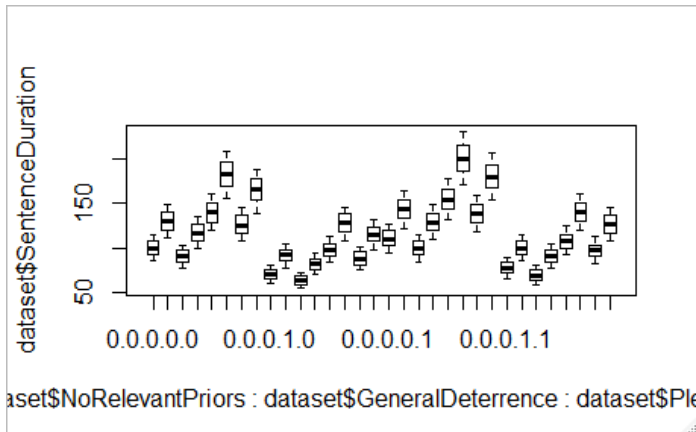
$`dataset$PleaGuilty`
      diff      lwr      upr p adj
1-0 -39.425 -39.77416 -39.07584    0
```

5 Factors: (3 Aggravating, 2 Mitigating): 1000 observation per unique combination, base line sentence = 100 days.

Distribution of DV:



Balance Design:



Results:

Anova Table (Type II tests)

Response: dataset\$SentenceDuration

	Sum Sq	Df	F value	Pr(>F)
dataset\$ShowNoRemorse	7550054	1	53303.3	< 2.2e-16 ***
dataset\$NoRelevantPriors	1191184	1	8409.7	< 2.2e-16 ***
dataset\$GeneralDeterrence	12093546	1	85380.3	< 2.2e-16 ***
dataset\$PleaGuilty	13757175	1	97125.5	< 2.2e-16 ***
dataset\$SpecificDeterrence	977970	1	6904.5	< 2.2e-16 ***
Residuals	4531734	31994		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
$`dataset$ShowNoRemorse`
      diff      lwr      upr p adj
1-0 30.72062 30.45983 30.98142    0

$`dataset$NoRelevantPriors`
      diff      lwr      upr p adj
1-0 -12.20237 -12.46317 -11.94158    0

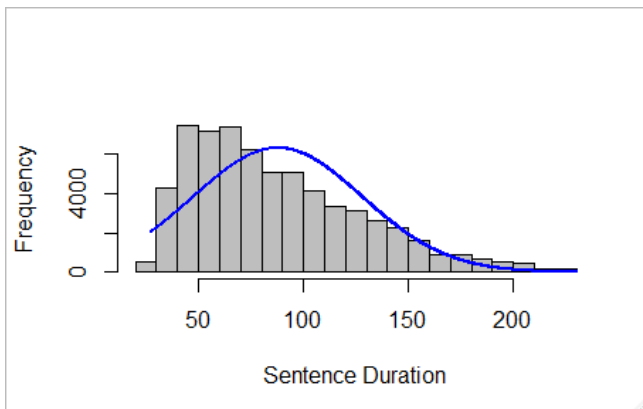
$`dataset$GeneralDeterrence`
      diff      lwr      upr p adj
1-0 38.8805 38.6197 39.1413    0

$`dataset$PleaGuilty`
      diff      lwr      upr p adj
1-0 -41.46862 -41.72942 -41.20783    0

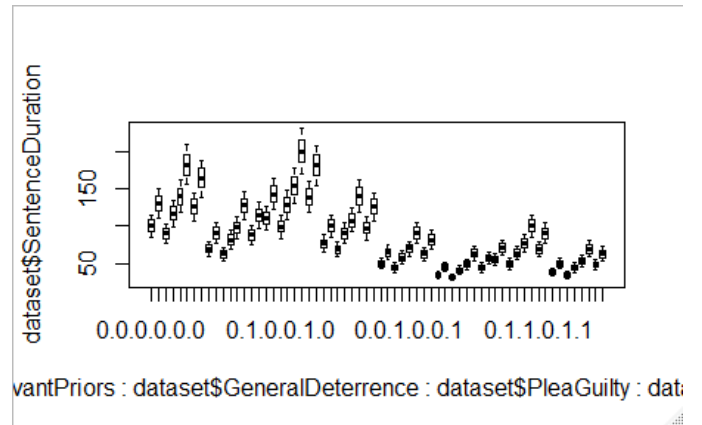
$`dataset$SpecificDeterrence`
      diff      lwr      upr p adj
1-0 11.0565 10.7957 11.3173    0
```

6 Factors: (3 Aggravating, 3 Mitigating): 1000 observation per unique combination, base line sentence = 100 days.

Distribution of DV:



Balance Design:



Results:

Anova Table (Type II tests)

Response: dataset\$SentenceDuration

	Sum Sq	Df	F value	Pr(>F)
dataset\$ShowNoRemorse	8373257	1	52872.5	< 2.2e-16 ***
dataset\$NoRelevantPriors	1347110	1	8506.3	< 2.2e-16 ***
dataset\$GeneralDeterrence	13682566	1	86397.8	< 2.2e-16 ***
dataset\$PleaGuilty	15424088	1	97394.6	< 2.2e-16 ***
dataset\$SpecificDeterrence	1104815	1	6976.3	< 2.2e-16 ***
dataset\$ShowRemorse	54745003	1	345684.4	< 2.2e-16 ***
Residuals	10134381	63993		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
$`dataset$ShowNoRemorse`
      diff      lwr      upr p adj
1-0 22.87637 22.68138 23.07137    0

$`dataset$NoRelevantPriors`
      diff      lwr      upr p adj
1-0 -9.17575 -9.370744 -8.980756    0

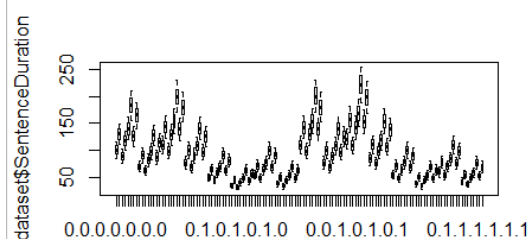
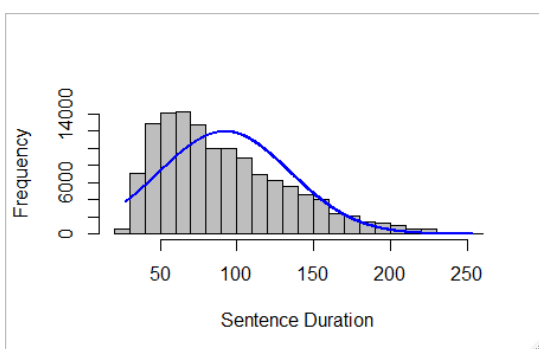
$`dataset$GeneralDeterrence`
      diff      lwr      upr p adj
1-0 29.24313 29.04813 29.43812    0

$`dataset$PleaGuilty`
      diff      lwr      upr p adj
1-0 -31.04844 -31.24343 -30.85344    0

$`dataset$SpecificDeterrence`
      diff      lwr      upr p adj
1-0 8.309687 8.114694 8.504681    0

$`dataset$ShowRemorse`
      diff      lwr      upr p adj
1-0 -58.49412 -58.68912 -58.29913    0
```

7 Factors: (4 Aggravating, 3 Mitigating): 1000 observation per unique combination, base line sentence = 100 days.



Anova Table (Type II tests)

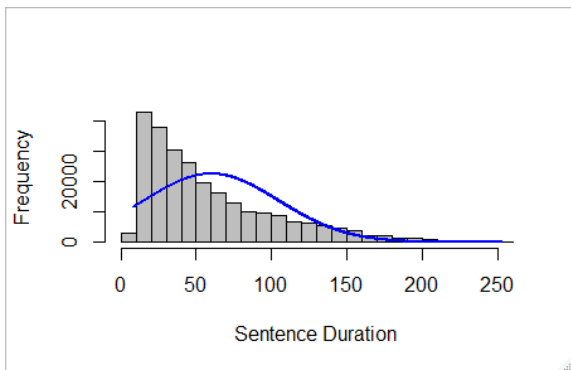
Response: dataset\$SentenceDuration

	Sum Sq	Df	F value	Pr(>F)
dataset\$ShowNoRemorse	18396449	1	103278	< 2.2e-16 ***
dataset\$NoRelevantPriors	3053184	1	17141	< 2.2e-16 ***
dataset\$GeneralDeterrence	30179761	1	169430	< 2.2e-16 ***
dataset\$PleaGuilty	33949079	1	190591	< 2.2e-16 ***
dataset\$SpecificDeterrence	2438184	1	13688	< 2.2e-16 ***
dataset\$ShowRemorse	120569102	1	676877	< 2.2e-16 ***
dataset\$CommunityProtection	2434258	1	13666	< 2.2e-16 ***
Residuals	22798632	127992		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

8 Factors (4 Aggravating, 4 Mitigating): 1000 observation per unique combination, base line sentence = 100 days.

Distribution of DV:



Balance Design:

Results:

Anova Table (Type II tests)

Response: dataset\$SentenceDuration				
	Sum Sq	Df	F value	Pr(>F)
dataset\$ShowNoRemorse	15564715	1	52123.0	< 2.2e-16 ***
dataset\$NoRelevantPriors	2518961	1	8435.5	< 2.2e-16 ***
dataset\$GeneralDeterrence	25520622	1	85463.3	< 2.2e-16 ***
dataset\$PleaGuilty	28621280	1	95846.7	< 2.2e-16 ***
dataset\$SpecificDeterrence	2070864	1	6934.9	< 2.2e-16 ***
dataset\$ShowRemorse	101740346	1	340707.3	< 2.2e-16 ***
dataset\$CommunityProtection	2052011	1	6871.8	< 2.2e-16 ***
dataset\$GamblingAddiction	266536778	1	892576.2	< 2.2e-16 ***
Residuals	76442787	255991		

 Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

$`dataset$ShowNoRemorse`
      diff      lwr      upr p adj
1-0 15.59483 15.46095 15.72871    0

$`dataset$NoRelevantPriors`
      diff      lwr      upr p adj
1-0 -6.273656 -6.407536 -6.139777    0

$`dataset$GeneralDeterrence`
      diff      lwr      upr p adj
1-0 19.96897 19.83509 20.10285    0

$`dataset$PleaGuilty`
      diff      lwr      upr p adj
1-0 -21.14728 -21.28116 -21.0134    0

$`dataset$SpecificDeterrence`
      diff      lwr      upr p adj
1-0 5.688344 5.554464 5.822223    0

$`dataset$ShowRemorse`
      diff      lwr      upr p adj
1-0 -39.87095 -40.00483 -39.73707    0

$`dataset$CommunityProtection`
      diff      lwr      upr p adj
1-0 5.662391 5.528511 5.79627    0

$`dataset$GamblingAddiction`
      diff      lwr      upr p adj
1-0 -64.534 -64.66788 -64.40012    0

```


7. Results:

The results suggest that having a balanced dataset with enough observations is paramount in creating an accurate model for the purposes of quantifying sentence durations. Through other tests not included in this document, the same models were created with datasets that were smaller and resulted in less accurate modelling. It is note worth although that even a smaller dataset which is balanced preformed better than a larger one which is imbalanced.

Factorial ANOVA seemed to be able to determine the factor influence quite accurately meaning if the assumptions the team made about nature of sentencing decisions and how factors interact with them, and enough output from the NLP model, the statistical analysis will be able to accurately quantify the factor impact.

As the number of factors grew, the accuracy of the models lowered, this can be attributed to various causes, for example:

- The influence of several factors may be similar blurring the distinction between the impact of each factor.
- The base line sentence used may be too small leading to smaller overall changes when factors are applied to it.
- The number of observations required as the factor count increase could increase.
- As the number of factors grew, the dataset became more skewed
 - A chance of this is purely the factor modifiers we have chosen, but it is also possible that this is an accurate interpretation and a more complete dataset sourced from legal experts will show a similar pattern. Input from legal experts would be paramount in assessing this

The first two points would be solved to an extent with the input from legal professionals to help guide and fine tune the numbers used to predict for sentencing duration.

The second two points could potentially be solved by offloading the work to an AWS server or some other cloud-based technologies. I attempted to generated datasets with a larger number of factors included with several thousand observations per combination as opposed to 1000 but my computer / other resources available could not handle the workload properly and would often freeze and/or crash.

8. Other Models:

Three other models were developed and tested but their output is not included. The three models that were created were a Tree, a Random Forrest and a Neural Net. These models used a portion of the data, took the factors as their inputs and attempted to learn the rules associated with the factors and sentencing decisions.

All three models resulted in 89-98% accuracy in the predicted vs observed measurement in determining the sentence duration based on the factor input, although when looking at other model evaluators, it became apparent that there was some over fitting taking place. This is likely to have occurred for the reasons discussed above.

9. Conclusion:

The purpose of this statistical analysis was to determine a valid method and conduct hypothetical testing in order to quantify the impact of aggravating and mitigating factors on sentencing decisions. As this part of the project had a task dependency (the Orang NLP model) the team had to generate some test data in order to start investigating a proper process. Unfortunately, the team had to make many assumptions based off our own personal research regarding the nature of base sentences, factor modification and other legal practices. In doing this, it is entirely possible that the team introduced bias, or system inaccuracies as a result of being uninformed on the complex nature of the legal system.

Considering this, the team managed to produce several datasets and conduct various types of investigations to get a wholistic view of the different “off-the-shelf” statistical models and investigate which are best suited for this problem space and how they should be applied.

Our first attempts at investigating this problem lead to the use of several models although our dataset generation was not suited for the task required. As such, the dataset was statistically biased in several ways leading to non-sensical results. This bad dataset lead to us learning a lot about what was required from the NLP model in order to generate data which could be used for the statics, what the statistics process entailed and how to better generate test data.

Our second attempts proved much more fruitful leading to various datasets being generated and various conclusions. Given a proper amount of data in a balanced design, the statistical test / modelling Factorial ANOVA can produce very accurate quantifications of the factors on sentencing decisions. As factor counts increase, this quantification becomes less accurate but there are understandable reasons as to why and potential solutions have been suggested. For example, creating a larger dataset with use of cloud-based technologies to execute the scripts.

Recommendation to future groups would be to investigate more into the base line of sentences for various crimes, and legal professional input regarding the nature of factors. Armed with this information, it would be possible to better tune the NLP model and better tune the analysis leading to better results across the board.

An added benefit would be a better understanding of how to generate test data in order to test new ideas and the other models more accurately (Tree, Random Forrest, Neural Net).

Appendix A:

```
import csv
import random

class Generator:

    # ---- Methods ----
    @classmethod
    def test(cls):

        ADD_RANDOM_NOISE = True
        RANDOM_NOISE_SIGMA = 0.1
        SAMPLE_SIZE = 1000

        l_charges = [
            ("Theft", 0.5),
            ("Theft of a motor vehicle", 0.5),
            ("Theft of a firearm", 1),
            ("Trafficking in a large commercial quantity of a drug of dependence", 7),
            ("Trafficking in a commercial quantity of a drug of dependence", 7),
            ("Trafficking in a non-commercial quantity of a drug of dependence", 1.5),
            ("Burglary", 1.33),
            ("Aggravated burglary", 3),
            ("Assault (Common Law)", 0.5),
            ("Common assault", 0.08),
            ("Assault with weapon or instrument", 0.25),
            ("Incest - sexual penetration of own child or lineal descendant or child of de facto aged under 18", 4.75)
        ]

        l_agg_factors = [
            ("No remorse", 1.35),
            ("General deterrence", 1.2),
            ("Specific deterrence", 1.3),
            ("Community protection", 1.4)
        ]

        l_mit_factors = [
            ("No relevant priors", 1.25),
            ("Plea guilty", 1.5),
            ("Remorse", 1.2),
            ("Gambling addiction", 1.1)
        ]

        l_column_titles = []

        for l_name in l_charges:
            l_column_titles.append(l_name[0])

        for l_name in l_agg_factors:
            l_column_titles.append(l_name[0])

        for l_name in l_mit_factors:
            l_column_titles.append(l_name[0])

        l_column_titles.append("Sentence length (Days)")
        print(l_column_titles[0])

        # Rows
        with open('Test.csv', 'w', newline='') as l_csv_file:
            l_file_writer = csv.writer(l_csv_file, delimiter=',', quotechar='|', quoting=csv.QUOTE_MINIMAL)

            l_file_writer.writerow(l_column_titles)

            for i in range(SAMPLE_SIZE):

                # Setup
                l_base_sentence = 0

                # ----- Charges -----
                # Charge count
```

```

l_charge_count = random.normalvariate(0, 1.5)
l_charge_count = max(1, l_charge_count)
l_charge_count = round(l_charge_count)
l_charge_count = min(len(l_charges), l_charge_count)

# Charge Indexes
l_charges_row = [0] * len(l_charges)
l_charge_index_set = set()

for j in range(l_charge_count):
    l_rand_index = random.randrange(len(l_charges))
    l_charge_index_set.add(l_rand_index)
    l_charges_row[l_rand_index] = 1

# Base sentence
for l_charge_index in l_charge_index_set:
    l_base_sentence += l_charges[l_charge_index][1]

l_sentence = l_base_sentence

# ----- Agg Factors -----

# Agg factor count
l_agg_count = random.normalvariate(0, 1.5)
l_agg_count = max(1, l_agg_count)
l_agg_count = round(l_agg_count)
l_agg_count = min(len(l_agg_factors), l_agg_count)

# Agg Indexes
l_agg_row = [0] * len(l_agg_factors)
l_agg_index_set = set()

for j in range(l_agg_count):
    l_rand_index = random.randrange(len(l_agg_factors))
    l_agg_index_set.add(l_rand_index)
    l_agg_row[l_rand_index] = 1

# Base factor
for l_agg_index in l_agg_index_set:
    l_sentence *= l_agg_factors[l_agg_index][1]

# ----- Mit Factors -----

# Mit factor count
l_mit_count = random.normalvariate(0, 1.5)
l_mit_count = max(1, l_mit_count)
l_mit_count = round(l_mit_count)
l_mit_count = min(len(l_mit_factors), l_mit_count)

# Mit Indexes
l_mit_row = [0] * len(l_mit_factors)
l_mit_index_set = set()

for j in range(l_mit_count):
    l_rand_index = random.randrange(len(l_mit_factors))
    l_mit_index_set.add(l_rand_index)
    l_mit_row[l_rand_index] = 1

# Base factor
for l_mit_index in l_mit_index_set:
    l_sentence /= l_mit_factors[l_mit_index][1]

# ----- Calculate sentence -----

if ADD_RANDOM_NOISE:
    l_random_noise_factor = random.normalvariate(1, RANDOM_NOISE_SIGMA)
    l_random_noise_factor = max(0, l_random_noise_factor)
    l_sentence *= l_random_noise_factor

l_sentence *= 365
l_sentence = round(l_sentence)
l_sentence = max(0, l_sentence)

```

```

# ----- Calculate row -----
l_row = l_charges_row + l_agg_row + l_mit_row + [l_sentence]
l_file_writer.writerow(l_row)

# ---- Main method ----
def main():
    Generator.test()
    print("Done!")

# ---- Main method definition ----
if __name__ == '__main__':
    main()

```

Appendix B:

```

# =====
# == Libaries
# =====

import random
import csv
# =====

# =====
# == Functions
# =====
# Generate random number within range
#         - Used to create random noise in the dataset
def getNoise():
    return random.uniform(LOWER_RANDOM_LIMIT, UPPER_RANDOM_LIMIT)

# Returns a list where each element is a character from the pass string
#         - Used to evalut each individual factor in a binary string
#         - e.g. 0110 = ['0','1','1','0']
def split_str(s):
    return list(s)
# =====

# =====
# == Controls
# =====
# Bounds for random noise
LOWER_RANDOM_LIMIT = 0.85
UPPER_RANDOM_LIMIT = 1.15

# Base duration of all sentences
BASE_SENTENCE_DURATION = 100

# Each factor has 2 levels
#         - Script can only account for all factors having the same level count
FACTOR_LEVEL_COUNT = 2

# How many data points to be generated per unqiue factor combination
# WARNING: If this number becomes big (approx >= 1000) and there are more than approx 10 factors
# will take some time to calculate
OBSERVATIONS_PER_COMBINATION = 1000
# =====

# =====
# == Factors
# =====
# List of Aggravating and Mitigating Factors and their multiplier
factors = [
    # Aggravating
    ("ShowNoRemorse", 1.3),
    ("GeneralDeterrence", 1.4),
    ("SpecificDeterrence", 1.1),
    ("CommunityProtection", 1.1),
    # Mitigating
    ("NoRelevantPriors", 0.9),

```

```

("PleaGuilty", 0.7),
("ShowRemorse", 0.5),
("GamblingAddiction", 0.3)

# Extra for proof
# ("Factor_0", 1.6),
# ("Factor_1", 1.8),
# ("Factor_2", 0.2),
# ("Factor_3", 1.4),
# ("Factor_4", 1.6),
# ("Factor_5", 0.6),
# ("Factor_6", 0.3),
# ("Factor_7", 0.7),
# ("Factor_8", 1.222),
# ("Factor_9", 1.32)
]
# =====
# =====
# == Set Up
# =====

# Calculate all unique combinations the factors can make up
uniqueCombinations = 1
for factor in factors:
    uniqueCombinations *= FACTOR_LEVEL_COUNT

# Convert a value into binary with the width dictated by how many factors there are
# i.e. If there are 4 factors, 01 will = 0010, if there are 5 factors, 01 will = 00010
factorStringWidth = ("0" + str(len(factors))) + "b"

# List to store all unique combinations
combinationList = []

# append combinations to combo list
for i in range(0, uniqueCombinations):
    combinationList.append(split_str(format(i, factorStringWidth)))

# List to store CSV column titles
columnTitles = []
# Append all factors to column title list
for factor in factors:
    columnTitles.append(factor[0])

# Append sentence duration as final column title
columnTitles.append("SentenceDuration")
# =====
# =====
# == Calculate Sentence and Write to CSV
# =====
CSV_FILE_NAME = str(len(factors)) + "_Factors.csv"

# Open CSV file
with open(CSV_FILE_NAME, 'w', newline='') as I_CSV:
    I_FileWriter = csv.writer(I_CSV, delimiter=',', quotechar='|', quoting=csv.QUOTE_MINIMAL)

    # Write column titles as first row
    I_FileWriter.writerow(columnTitles)

    # For each combination
    for combination in combinationList:

        # Create the set amount of obserations
        for i in range(0, OBSERVATIONS_PER_COMBINATION):

            # Calculate the sentence for this observation starting with the base
            I_Sentence = BASE_SENTENCE_DURATION

            # temporary list to hold the row to be written
            I_RowToWrite = []

            # For each factor in the current combination

```

```

        for j in range(0, len(combination)):
            # Append it to the row to be written
            l_RowToWrite.append(combination[j])
            # Recalculate the sentence duration if the current factor was present by that factors multiplier
            if combination[j] == "1":
                l_Sentence = float(l_Sentence) * float(factors[j][1])

        # Add noise and round value
        l_RowToWrite.append(round(l_Sentence * getNoise()))
        # Write entire row to CSV file
        l_FileWriter.writerow(l_RowToWrite)

# =====

# =====
# == Print Dataset Stats
# =====
print("_____")
print("| \t!!! Dataset Succesfully Created !!!\t\t|")
print("|_____|")
print("| File name: \t\t\t", CSV_FILE_NAME, " |")
print("| Total amount of factors:\t\t", len(factors), "\t\t|")
print("| Total amount of levels per factor:\t", FACTOR_LEVEL_COUNT, "\t\t|")
print("| Total unique combinations:\t\t", uniqueCombinations, "\t\t|")
print("| Observations per combination:\t\t", OBSERVATIONS_PER_COMBINATION, "\t\t|")
print("| Dataset total observations: \t\t", uniqueCombinations * OBSERVATIONS_PER_COMBINATION, "\t\t|")
print("|_____|")
# =====

```