

Improving Data Quality in the Smart Template Project

Final Project Report

Master Of Science in Information Quality

University of Arkansas at Little Rock

By: **Romeo Klamadji**

Faculty Advisor: **Dr. John Talburt**

Sponsoring Organization: **U.S. Food and Drug Administration, NCTR**

Sponsor Supervisor: **Meehan Joe**, Computer Scientist, Joe.Meehan@fda.hhs.gov,

(870) – 543 - 7658

Contents

Executive Summary:.....	3
Introduction	4
Deliverables.....	5
Data Presentation	5
Data Extraction	7
Data Analyzation	8
Data Uniqueness Assessment	10
Data Cleansing	13
Data Measurement	14
Conclusion.....	16

Executive Summary:

The Sponsor for my project is the Executive Carcinogenicity Assessment Committee (ECAC) of the FDA. The FDA is the Food and Drug Administration, a federal agency of the Department of Health and Human Services. The Exec CAC is the department that ensures consistency in recommendations and conclusions regarding protocols for carcinogenicity studies across review divisions. The Exec CAC meets regularly to review all carcinogenicity protocols and final study reports. These reviews allow them to determine how safe a drug is and gives the FDA managers the opportunity to ameliorate review methods for faster drug approval. The record minutes of the Exec CAC meetings constitute meeting minutes documents. They are composed of the Executive Carcinogenicity Assessment Committee (ECAC), Special Protocol Assessments, Protocol Modifications, and Final Study Reports. These meeting minutes documents are used by Pharmacology/Toxicology reviewers who review Investigational New Drug Applications to gather useful information from previous reviews.

Meeting minutes documents are generated in unstructured text documents. Without prior knowledge, it was difficult for other personnel to gather detailed information from them because of the lack of normalization in the field tables. Also, users couldn't launch a search that would return the content results from all these documents at the same time. To execute this type of search, we needed to link all these files together using a common identifier.

The long-term goal of ECAC is to create a search engine for meeting using a Smart Template system developed by the U.S. Food and Drug Administration. The objective is to have a dashboard where users can access information such as Protocol, Modification and Final Study from meeting minutes documents by just typing the application numbers of these documents. The goal is for the dashboard to return quick search results associated with all the data present in the different documents.

The main objective of this project was to support this long-term goal of the ECAC. First and foremost, we were curious to see if we can link different type of data together and pull them using a common identifier. For a trial, we decided to search how many "MEETING_FILE_TYPE 3", protocol procedure have corresponding "MEETING_FILE_TYPE 4", final study. For this, we needed a common identifier, so we took "APPLICATION_NUMBER", since "MEETING_FILE_TYPE 3" and "MEETING_FILE_TYPE 4" have matching "APPLICATION_NUMBER". For this to work, I must set up a method that will return "MEETING_FILE_TYPE 3" corresponding to "MEETING_FILE_TYPE 4" using their matching "APPLICATION_NUMBER".

Secondly, after finding matching "APPLICATION_NUMBER" for both "MEETING_FILE_TYPE 3" and "MEETING_FILE_TYPE 4", I must incorporate a normalization method in this data because the data present was of low quality and not appropriate to use. This is the second problem I am helping the ECAC solve, the data normalization problem. Finding these application numbers was just a first step because the existing data showed some quality issues and had serious inconsistent format. To get the best performance from the data, it must be clean, so cleaning and normalization of the data become an important task in the execution of this project.

Therefore, for a start, my project helped the ECAC find matching application numbers from two different file types, 3 and 4 present in the meeting minutes documents. The hypothesis was that matching these two file types using their application numbers, can help create a link for a machine learning algorithm that can be used to return research results associated to every document. An automated search can then be implemented, one that can fetch all the information from each study using that kind of connection. This aligns with the long-term goal of ECAC, allowing users to perform automated searches by just typing application numbers as a search key in the FDA Dashboard and have the system return all the information present in these reports stored in the database. This allows the users to access all the information in one place.

Then, normalization and cleaning the application numbers obtained from process one was the final help that I provided to the ECAC. They wanted to make sure that the data collected was ready to be used as expected.

Introduction

To get a better understanding on the research study process for any given drug put on the market by the FDA, we basically must gather the data from Protocol modification, file type 3, till the Final Study, file type 4, for each study report. How do we gather all the information from different meeting minutes documents and link them together? Since each study has an application number, either IND or NDA numbers associated to it, we can use this to track each file and collect all the information.

Primarily, all the data are extracted from meeting minutes using a variety of text mining techniques and deposited in an Oracle database. A relational database based on a relational model that represents an organization data in the form of tables. In general, the data are stored in the database exactly as they appear in the source document with no standardization or regularization. Since the data are primarily intended to be read by experienced reviewers who can easily disambiguate differing representations, the lack of standardization is not usually a problem. However, when attempting to collect all the information for a particular drug search, the lack of standardization in the representation of application numbers makes it difficult to correctly match all the information to a specific type of drug. So, it is hard for reviewers to track information based on criteria search such as animal species (rat, mouse, and hamster), drug substance or pharmacological class.

Hence, the other objective of this project is to categorize and normalize the application numbers we have obtained previously, so that they can be used as a future reference for the dashboard engine the FDA will build.

For my project, I had to conduct an analysis of these meeting minutes documents. Using the python software, I was able to access the content from each table. I have extracted this information and reorganized them by field names based on the results that were expected. Then, using the python libraries, I proceeded to analyze these contents and develop scripts that will provide better results and return requested matching fields.

Secondly, I had to recreate the table results, so that they can be exported through csv files. Then, based on Excel, I implemented a normalization method so that the tables can maintain a certain consistency in the nomination.

All these were aimed to provide a solution, to find all the application numbers that matched the two types of files that were present in the meeting minutes. Then, improve the data quality from these application numbers which helps to better understand them and use them properly.

Deliverables

This was the following list of the deliverables that ECAC has tasked me to submit as part of my project:

- Python and Excel files with the scripts of the data manipulation methods used for analyzing the data.
- The visualization showing the matching application numbers of all the meeting minutes documents.
- The csv files constituting the search results from the project.

Data Collection

Accessing the data was the priority, and these documents were extracted from the oracle database and provided to me in a form of xml files. XML files are files that can be read by computer programs used to store data. Xml files consist of plain text and tags. The plain text is the actual data being stored and the tags indicates what that data is and its type. Each tag should have its respectively closing tag.

In the Figure 1 below, on the file to the left, we have a preview of an xml file:

- **<ROWSET>**: The root element containing all the other tags.
- **<APPLICATION_NUMBER>**: One of the many child elements of ROWSET.

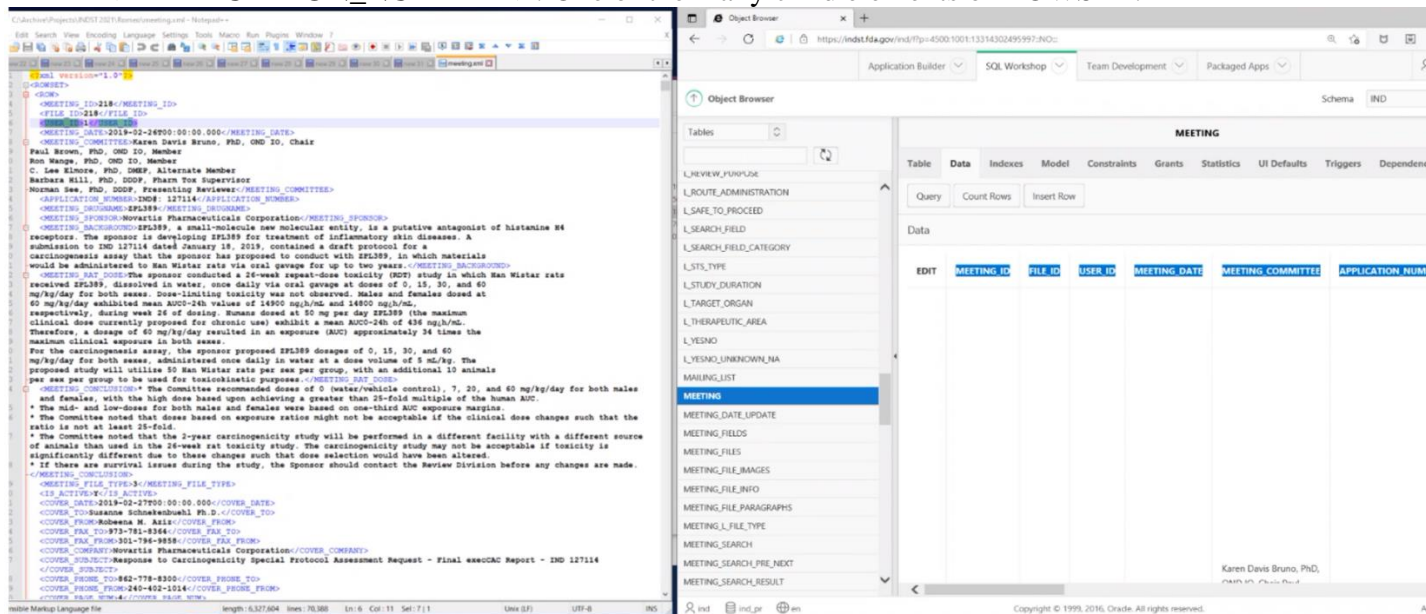


Figure 1: Oracle Database Table Composition

Still using Figure 1, it displayed on the left end side a table. This table shows how in a relational database, the columns of the table hold the attributes of the data, and every record contains a value for each attribute, and this establishes a relation between the data points. So, in this case, each information or attribute extracted has its own column, and the database assigns a unique identifier like an ID or a key to each row. So, records present in these files includes APPLICATION_NUMBER, MEETING_ID, MEETING_FILE_TYPE, MEETING_DATE and so on. Generally, the key or ID column is used as a common column to create a relationship between two tables, which in this case, it is the APPLICATION_NUMBER column. This will allow us to link attributes even if they are in different tables and return them as a result.

Indeed, it's the meeting minutes data table from the right on the Figure 1 that produced the table on the left of that. As reported before, it was exported from the oracle database into that xml file. That created the xml file showcased to the right on Figure 1. A closer observation (from Figure 2 below) of one of the xml files I have used will reveal its deep composition. In this file, like it was mentioned before, each tag constituted an element, and those elements are arranged in hierarchy, which gives them the capability to contain other elements. Mostly, the file starts with a topmost element, the root, which here is 'ROWSET'. 'ROWSET' contains diverse 'ROW' elements, which are its element child. These elements also contain other elements such as MEETING_ID, FILE_ID, MEETING_DATE and so on. If you can recall, these were the column names of our data table.

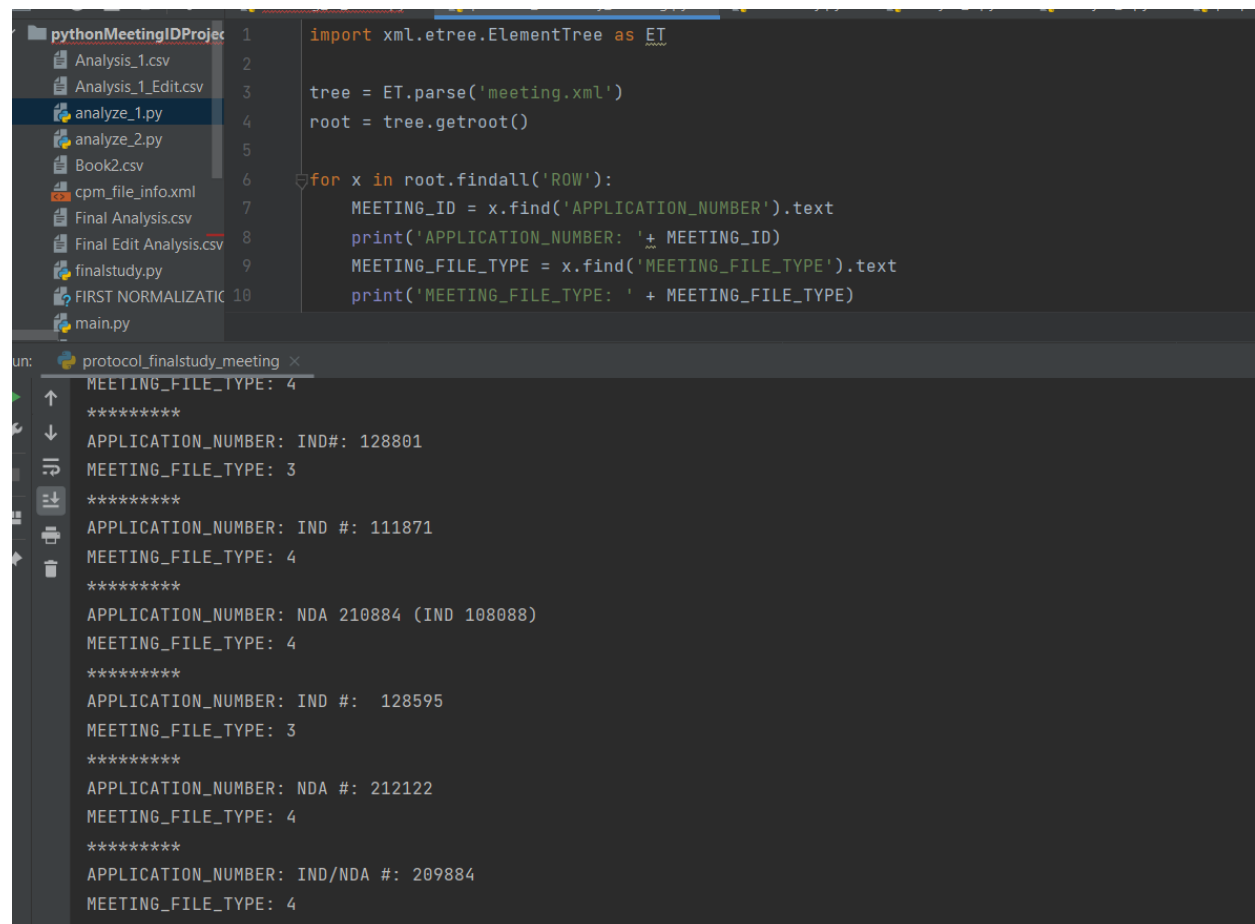
```
<?xml version="1.0"?>
- <ROWSET>
  - <ROW>
    <MEETING_ID>218</MEETING_ID>
    <FILE_ID>218</FILE_ID>
    <USER_ID>1</USER_ID>
    <MEETING_DATE>2019-02-26T00:00:00.000</MEETING_DATE>
    <MEETING_COMMITTEE>Karen Davis Bruno, PhD, OND IO, Chair Paul Brown, PhD, OND IO, Member Ron Wange, PhD, OND IO, Member C. Lee Elmore, PhD, DMEP, Alternate Member Barbara Hill, PhD, DDDP, Pharm Tox Supervisor Norman See, PhD, DDDP, Presenting Reviewer</MEETING_COMMITTEE>
    <APPLICATION_NUMBER>IND#: 127114</APPLICATION_NUMBER>
    <MEETING_DRUGNAME>ZPL389</MEETING_DRUGNAME>
    <MEETING_SPONSOR>Novartis Pharmaceuticals Corporation</MEETING_SPONSOR>
    <MEETING_BACKGROUND>ZPL389, a small-molecule new molecular entity, is a putative antagonist of histamine H4 receptors. The sponsor is developing ZPL389 for treatment of inflammatory skin diseases. A submission to IND 127114 dated January 18, 2019, contained a draft protocol for a carcinogenesis assay that the sponsor has proposed to conduct with ZPL389, in which materials would be administered to Han Wistar rats via oral gavage for up to two years.</MEETING_BACKGROUND>
    <MEETING_RAT_DOSE>The sponsor conducted a 26-week repeat-dose toxicity (RDT) study in which Han Wistar rats received ZPL389, dissolved in water, once daily via oral gavage at doses of 0, 15, 30, and 60 mg/kg/day for both sexes. Dose-limiting toxicity was not observed. Males and females dosed at 60 mg/kg/day exhibited mean AUC0-24h values of 14900 ngch/mL and 14800 ngch/mL, respectively, during week 26 of dosing. Humans dosed at 50 mg per day ZPL389 (the maximum clinical dose currently proposed for chronic use) exhibit a mean AUC0-24h of 436 ngch/mL. Therefore, a dosage of 60 mg/kg/day resulted in an exposure (AUC) approximately 34 times the maximum clinical exposure in both sexes. For the carcinogenesis assay, the sponsor proposed ZPL389 dosages of 0, 15, 30, and 60 mg/kg/day for both sexes, administered once daily in water at a dose volume of 5 mL/kg. The proposed study will utilize 50 Han Wistar rats per sex per group, with an additional 10 animals per sex per group to be used for toxicokinetic purposes.</MEETING_RAT_DOSE>
    <MEETING_CONCLUSION>* The Committee recommended doses of 0 (water/vehicle control), 7, 20, and 60 mg/kg/day for both males and females, with the high dose based upon achieving a greater than 25-fold multiple of the human AUC. * The mid- and low-doses for both males and females were based on one-third AUC exposure margins. * The Committee noted that doses based on exposure ratios might not be acceptable if the clinical dose changes such that the ratio is not at least 25-fold. * The Committee noted that the 2-year carcinogenicity study will be performed in a different facility with a different source of animals than used in the 26-week rat toxicity study. The carcinogenicity study may not be acceptable if toxicity is significantly different due to these changes such that dose selection would have been altered. * If there are survival issues during the study, the Sponsor should contact the Review Division before any changes are made.</MEETING_CONCLUSION>
    <MEETING_FILE_TYPE>3</MEETING_FILE_TYPE>
    <IS_ACTIVE>Y</IS_ACTIVE>
    <COVER_DATE>2019-02-27T00:00:00.000</COVER_DATE>
    <COVER_TO>Susanne Schneckebuehl Ph.D.</COVER_TO>
    <COVER_FROM>Robeena M. Aziz</COVER_FROM>
    <COVER_FAX_TO>973-781-8364</COVER_FAX_TO>
    <COVER_FAX_FROM>301-796-9858</COVER_FAX_FROM>
    <COVER_COMPANY>Novartis Pharmaceuticals Corporation</COVER_COMPANY>
    <COVER_SUBJECT>Response to Carcinogenicity Special Protocol Assessment Request - Final execCAC Report - IND 127114</COVER_SUBJECT>
    <COVER_PHONE_TO>862-778-8300</COVER_PHONE_TO>
    <COVER_PHONE_FROM>240-402-1014</COVER_PHONE_FROM>
    <COVER_PAGE_NUM>4</COVER_PAGE_NUM>
    <COVER_COMMENTS>Email to susanne.schneckebuehl@novartis.com</COVER_COMMENTS>
  </ROW>
- <ROW>
  <MEETING_ID>219</MEETING_ID>
```

Figure 2: XML File Description

Data Extraction

As my mentor suggested to me, I used the Pandas library to read the xml files and extract the data. It is a software library written for the Python programming language. The Panda library is mainly used for data analysis. It allows importing data from various file formats such as xml files, csv files, comma-separated values, JSON, SQL database tables, Microsoft Excel and so on. Also, it allows diverse data manipulation methods such as merging, reshaping, selecting, as well as data cleaning.

Figure 3 below shows a data extraction operation I executed on the xml files using the Panda library with some Python scripts. I used the 'xml.etree.Element', a python module from the Panda library that breaks down the data into smaller parts from the uploaded file for a better analysis. I used it on the 'meeting' xml file I received and broke down the data into smaller pieces of data. I used the 'ROW' elements that I mentioned before and retrieved the 'Application Number' and 'Meeting_File_Type' for each of these elements present in the meeting xml file from the smaller pieces. Then, I built a new dataframe, which allowed me to save these results.



```
pythonMeetingIDProject
├── Analysis_1.csv
├── Analysis_1_Edit.csv
├── analyze_1.py
├── analyze_2.py
├── Book2.csv
├── cpm_file_info.xml
├── Final Analysis.csv
├── Final Edit Analysis.csv
├── finalstudy.py
├── FIRST NORMALIZATIO
└── main.py

1  import xml.etree.ElementTree as ET
2
3
4  tree = ET.parse('meeting.xml')
5  root = tree.getroot()
6
7  for x in root.findall('ROW'):
8      MEETING_ID = x.find('APPLICATION_NUMBER').text
9      print('APPLICATION_NUMBER: ' + MEETING_ID)
10     MEETING_FILE_TYPE = x.find('MEETING_FILE_TYPE').text
11     print('MEETING_FILE_TYPE: ' + MEETING_FILE_TYPE)
```

protocol_finalstudy_meeting

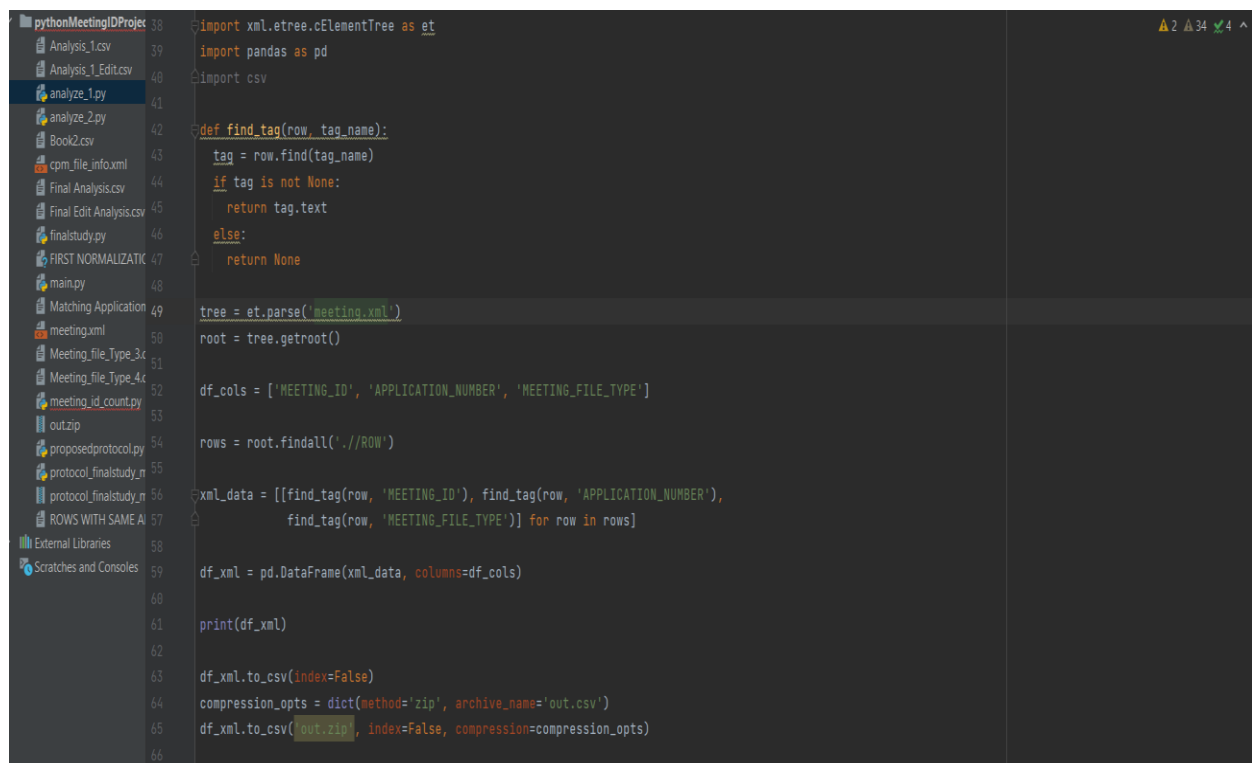
```
MEETING_FILE_TYPE: 4
*****
APPLICATION_NUMBER: IND#: 128801
MEETING_FILE_TYPE: 3
*****
APPLICATION_NUMBER: IND #: 111871
MEETING_FILE_TYPE: 4
*****
APPLICATION_NUMBER: NDA 210884 (IND 108088)
MEETING_FILE_TYPE: 4
*****
APPLICATION_NUMBER: IND #: 128595
MEETING_FILE_TYPE: 3
*****
APPLICATION_NUMBER: NDA #: 212122
MEETING_FILE_TYPE: 4
*****
APPLICATION_NUMBER: IND/NDA #: 209884
MEETING_FILE_TYPE: 4
*****
```

Figure 3: Data Extraction From XML File

Even though I built the dataframe, it was not accessible outside PyCharm, a software suited to write Python codes for further data analysis operations. PyCharm is the integrated development

environment that I was using for python. This brings us to our next problem which is the extraction of our dataframe from the PyCharm.

The solution that was suggested to me was to create a 2D arrays table to extract the dataframe from PyCharm. A two-dimensional table is a table that represents the data in forms of rows and columns. Figure 4 below shows the step-by-step procedure to create a 2D table. After parsing or breaking down the data from the meeting xml into pieces, I had to give the same names of our dataframe table columns to the new columns that I just created. I added the column “MEETING_ID” just so it can be used as a reference when this table will be added into the Oracle Database. Then, I populated the rows with the values found from the ‘ROW’ tags. It was necessary to create a small function, ‘find_tag’, to check cases where if row.find(‘tag name’) returns None. Because if it is None, I cannot access its ‘text’ attribute. Finally, I exported the new 2D table from PyCharm through a zip folder. All these constitute the operation needed to export the 2D table from the PyCharm software. I will be using this 2D table for further analysis in this project.



```

pythonMeetingIDProj 38
  Analysis_1.csv      39
  Analysis_1_Edit.csv 40
  analyze_1.py        41
  analyze_2.py        42
  Book2.csv           43
  cpm_file_info.xml   44
  Final Analysis.csv  45
  Final Edit Analysis.csv 46
  finalstudy.py       47
  FIRST NORMALIZATK  48
  main.py             49
  Matching Application 50
  meeting.xml         51
  Meeting_file_Type_3c 52
  Meeting_file_Type_4c 53
  meeting_id_count.py 54
  out.zip             55
  proposedprotocol.py 56
  protocol_finalstudy.m 57
  protocol_finalstudy.m 58
  ROWS WITH SAME A    59
  External Libraries  60
  Scratches and Consoles 61
  62
  63
  64
  65
  66

import xml.etree.cElementTree as et
import pandas as pd
import csv

def find_tag(row, tag_name):
    tag = row.find(tag_name)
    if tag is not None:
        return tag.text
    else:
        return None

tree = et.parse('meeting.xml')
root = tree.getroot()

df_cols = ['MEETING_ID', 'APPLICATION_NUMBER', 'MEETING_FILE_TYPE']

rows = root.findall('./ROW')

xml_data = [[find_tag(row, 'MEETING_ID'), find_tag(row, 'APPLICATION_NUMBER'),
             find_tag(row, 'MEETING_FILE_TYPE')] for row in rows]

df_xml = pd.DataFrame(xml_data, columns=df_cols)

print(df_xml)

df_xml.to_csv(index=False)
compression_opts = dict(method='zip', archive_name='out.csv')
df_xml.to_csv('out.zip', index=False, compression=compression_opts)

```

Figure 4: 2D Table Creation Process

Data Analyzation

Data Primary Assessment

Figure 5 below shows the 2D table that I exported from PyCharm. The result was to get a table with 3 columns for attributes which were “MEETING_ID”, “APPLICATION_NUMBER” and “MEETING_FILE_TYPE”.

As it was mentioned before in the hypothesis, the application numbers collected were in an inconsistent format. There were so many irregularities in the Application_Number. There were missing values, letters, spaces, symbols and so on. There was no constraint on the value entered nor formal standardization that was present. The only type of normalization present in the application numbers is the “IND#: or NDA#:” which is usually present, see Figure 5 below. However, this is not the case for every single number. These irregularities made the application numbers difficult to work with and unreliable to trust any result coming from it. Hence, using these application numbers at this stage of the project will certainly generate fake results.

	A	B	C	D
1	MEETING_ID	APPLICATION_NUMBER	MEETING_FILE_TYPE	
2	218	IND#: 127114		3
3	219	IND #: 127960		3
4	262	NDA #: 208794		4
5	263	NDA/IND #: 210251/ 121318		4
6	264	NDA #: 211172		4
7	251	NDA #: 211996/212161		4
8	221	IND#: 128801		3
9	268	IND #: 111871		4
10	252	NDA 210884 (IND 108088)		4
11	220	IND #: 128595		3
12	269	NDA #: 212122		4
13	253	IND/NDA #: 209884		4
14	254	IND #: 109678		4
15	222	IND #: 129181 and 129182		3
16	223	IND#: 130687		3
17	346	IND #: 122503		3
18	370	P-IND: 128,091		3
19	371	IND #: 128625		3
20	290			3
21	291	IND #: 112780		3
22	317	IND # 116,335		3
23	70	IND #: 129980		3
24	236	PIND #: 140789		3
25	216	IND#: 124547		3
26	217	IND#: 126396		3
27	260	NDA #: 206488		4
28	261	NDA #: 207488/16-00		4

Figure 5: Result 2D Table

• Data Completeness Measurement

If we take a close look at the Figure 5 above, we can observe the missing columns with empty values. However, each “MEETING_FILE_TYPE” element, either 3 or 4 should have an “APPLICATION_NUMBER” attribute associated to it. The “APPLICATION_NUMBER” shouldn’t have a blank nor a null field.

Advance analysis of this table was completed with the software Streamlit Python, an open-source app framework that helps create data science visualization. Figure 6 below gives a better analysis of the variables present. We can see that “APPLICATION_NUMBER” had 3.4% missing values compare to “MEETING_FILE_TYPE”, the other variable that had 0.0% missing values. Like I stated before, it will be difficult to predict any type of results using the “APPLICATION_NUMBER” element with that many missing values.

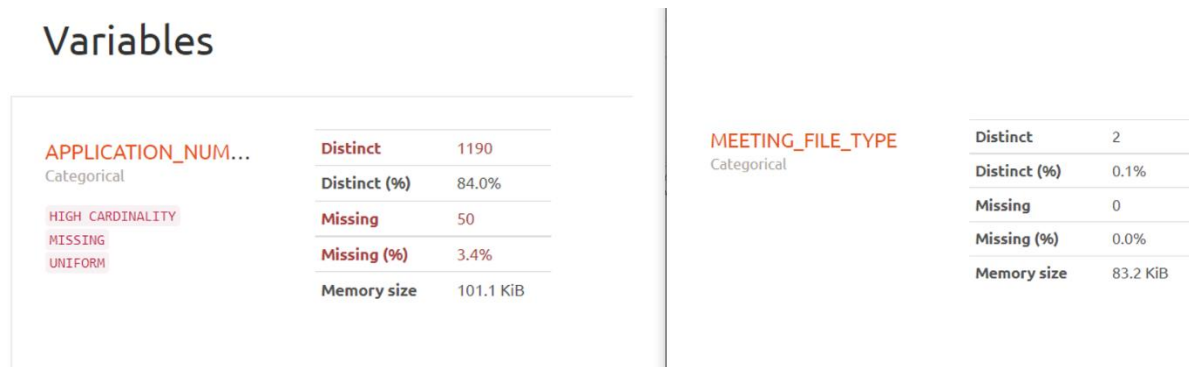


Figure 6: Analysis of Variables Present

- Data Consistency Measurement

Since for this project I am mainly searching to find matching “APPLICATION_NUMBER” elements for the two types, 3 and 4 “MEETING_FILE_TYPE”, it is natural to conduct a consistency assessment. This is a great method to measure the similarities between data items representing the same objects based on a given specific information requirements.

Based on the results from Figure 5, I wanted to conduct an initial consistency analysis for the matching “APPLICATION_NUMBER” that were already present in the tables. I was curious to see how many initial matching “APPLICATION_NUMBER” I can find for both 3 and 4 “MEETING_FILE_TYPE”. This process was done prior to any data manipulation or cleaning. For this operation, I used Powerquery from excel. It lets you analyze and transform your data within excel. I used it to merge tables and create a new join table. To create this join table, I used the “APPLICATION_NUMBER” column attribute as a Unique identifier. Since in the table, there were two different “MEETING_FILE_TYPE” types, 3 and 4, with a distinct “MEETING_ID” respectively for each. The result is shown on Figure 6 below, the first column indicated the “MEETING_ID” values, then the next 2 columns besides it indicated the matching “APPLICATION_NUMBER” found for both 3 and 4 “MEETING_FILE_TYPE”.

As noticed on Figure 7 below, I only found 6 matching “APPLICATION_NUMBER” elements for both 3 and 4 “MEETING_FILE_TYPE”. So, we have 6 matching “APPLICATION_NUMBER” from the original 1468 application numbers of entries.

	A	B	C	D	E	F
77	2052		3		4	
78	2053		3		4	
79	2053		3		4	
80	2187		3		4	
81	2187		3		4	
82	2057		3		4	
83	2057		3		4	
84	2188		3		4	
85	2188		3		4	
86	2189		3		4	
87	2189		3		4	
88	2062		3		4	
89	2062		3		4	
90	2065		3		4	
91	2065		3		4	
92	2068		3		4	
93	2068		3		4	
94	2070		3		4	
95	2070		3		4	
96	2071		3		4	
97	2071		3		4	
98	850 NDA 22-571		3 NDA 22-571		4	
99	1165 IND # 70,893		3 IND # 70,893		4	
100	1019 IND #: 70,568		3 IND #: 70,568		4	
101	1134 IND # 62,482		3 IND # 62,482		4	
102	1227 NDA 21-852		3 NDA 21-852		4	
103	1166 IND #: 70,961		3 IND #: 70,961		4	

Figure 7: Table of Initial Matching Application Numbers

• Data Accuracy Measurement

After the consistency measurement, I should measure the data accuracy next because this directly impacts the correctness of decision making in the usage of the data. It is a key component in data analysis practices. The accuracy measurement will determine the extent to which the data item correctly describes the object it is supposed to. So, I am going to measure the accuracy of the hypothesis that these two elements, “MEETING_FILE_TYPE 3” and “MEETING_FILE_TYPE 4” are linked or correlated using their matching “APPLICATION_NUMBER”.

For this, I have used was the Pearson correlation method trough the Streamlit Python Software. I used this to measure the strength of the linear relationship between the two variables, “MEETING_FILE_TYPE 3” and “MEETING_FILE_TYPE 4”. This will show if there is a connection between these variables.

Based on Figure 8 below, we can see that the Pearson's correlation coefficient (r), the measure of linear correlation between “MEETING_FILE_TYPE 3” and “MEETING_FILE_TYPE 4” lies in +1. Indeed, this proves that there is a strong relationship between “MEETING_FILE_TYPE 3” and “MEETING_FILE_TYPE 4”.

Correlations

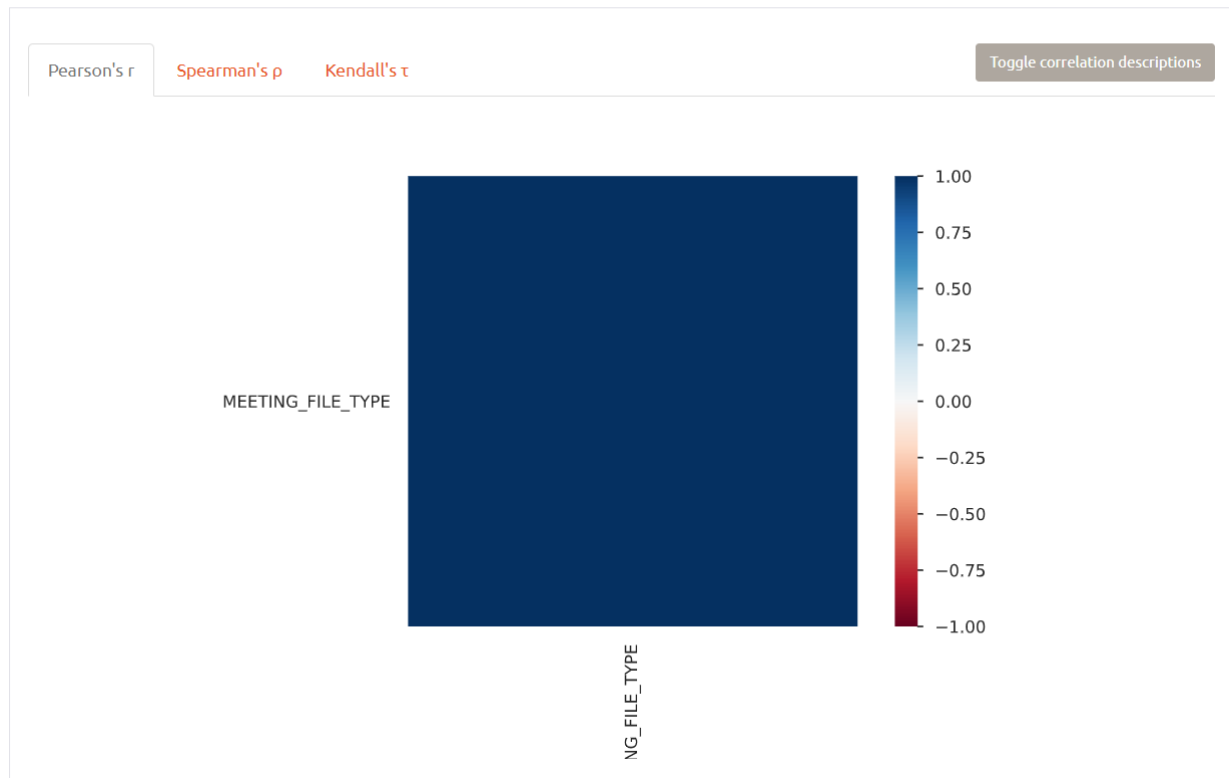


Figure 8: Pearson Correlation Graph

- Data Accuracy Metric

Another method to calculate the dependency between “MEETING_FILE_TYPE 3” and “MEETING_FILE_TYPE 4” is by using the data accuracy metric formula.

Measure of Accuracy Table 1= 1 – Number of Matching Application numbers / Total Application numbers

$$= 1 - (6 / 1467)$$

$$= 0.9959$$

This indicates the correlation coefficient of the relationship between “MEETING_FILE_TYPE 3” and “MEETING_FILE_TYPE 4”. So, if there is a strong dependency between these 2 values, we will get a correlation of 1 or closer to 1. Using this, we can conclude that there is a strong relationship between “MEETING_FILE_TYPE 3” and “MEETING_FILE_TYPE 4”. It shows a presence of a strong relationship since I am using matching “APPLICATION_NUMBER” from “MEETING_FILE_TYPE 3” and “MEETING_FILE_TYPE 4”. This supported our previous claim, and I expected nothing else.

Data Cleansing

Now, it's time to clean and manipulate the "APPLICATION_NUMBER" data and rerun the previous data analysis processes to see if I can get better results.

In this process, I will be detecting and correcting all the corrupt and inaccurate "APPLICATION_NUMBER" records present in the table. My first focus was to delete the irregularities that were most frequent in the table

The first problem present in the table was all the empty cells. Using the Excel "find & search" method, I found and removed the multiple empty cells present in the table.

The second problem was the presence of extra spacing in the "APPLICATION_NUMBER" records. I used the Excel function 'SUBSTITUTE (Column num, " ", "")', to get rid of all the trailing and extra spaces present in the table for each "APPLICATION_NUMBER" record.

Finally, I used the excel search function to make an auto search for multiple cells and replaced the nominations and format of the "APPLICATION_NUMBER" records, so that they can all have the same format.

After deep cleaning and categorization of all the "APPLICATION_NUMBER" records using the Excel functions, the results can be seen on Figure 9 below. The "APPLICATION_NUMBER" records were transformed starting from the process on the right until the final format in Table 3 on the left.

→ 222 IND#: 129181 and 129182 3			Table 1 Rows with MEETING_FILE_TYPE 3		
222	IND#: 129181	3	218	IND#: 127114	3
222	IND#: 129182	3	219	IND#: 127960	3
			262	NDA#: 208794	3
			263	NDA#: 210251	3
			221	IND#: 128801	3
			Table 2 Rows with MEETING_FILE_TYPE 4		
			262	NDA#: 208794	4
			263	NDA#: 210251	4
			263	IND#: 121318	4
			264	NDA#: 211172	4
			251	NDA#: 211996	4
			Table 3 Find rows with the same application number:		
			Table 1		Table 2
			262	NDA#: 208794	3
			263	NDA#: 210251	3
			262	NDA#: 208794	4
			263	NDA#: 210251	4

Figure 9: New Normalization Model for Matching Application Number

Data Secondary Assessment

After the normalization and the cleaning of the "APPLICATION_NUMBER" records, I obtained a better table. This new updated dataset will provide a better analysis than the previous one.

Therefore, I wanted to put this new table to the same measurement processes (Completeness, Consistency and Accuracy) as I put the former table.

- **Data Secondary Completeness Measurement**

Figure 10 below showed the dataset statistics from Steamlit Python from the new table. There were not any anomalies present in the table. We can observe that even the missing fields from the “APPLICATION_NUMBER” fields that were present previously have all been eradicated. Compared to a 3.4 % missing value percentage at the beginning to a 0.0%, this is a net improvement.

From here, I figured that I would have better results.

Dataset statistics

Number of variables	3
Number of observations	91
Missing cells	0
Missing cells (%)	0.0%

Figure 10: Dataset Statistics

- **Data Secondary Consistency Measurement**

Using the same guidelines used for the primary consistency measurement analysis, I am going to conduct a second consistency assessment on the new dataset that is cleaned. I was curious to see if this will increase the number of similar “APPLICATION_NUMBER” present in both 3 and 4 “MEETING_FILE_TYPE”.

Using the Powerquery method from excel, I recreated the join table found on Figure 11 below. This returned more “APPLICATION_NUMBER” elements than what I gathered with the previous join table. There were 91 matching application numbers that were returned compared to the previous 6. Indeed, normalization has improved the return number of matchings “APPLICATION_NUMBER” for both “MEETING_FILE_TYPE”, type 3 and 4. This was my expected result.

	A	B	C	D	E	F	G	H	I
1	MEETING_ID	APPLICATION_NUMBER	MEETING_FILE_TYPE	Meeting_file_Type_4.MEETING_ID	Meeting_file_Type_4.APPLICATION_NUMBER	Meeting_file_Type_4.MEETING_FILE_TYPE			
2	339	IND#: 121318	3	263	IND#: 121318	4			
3	339	IND#: 121318	3	142	IND#: 121318	4			
4	340	IND#: 121318	3	263	IND#: 121318	4			
5	340	IND#: 121318	3	142	IND#: 121318	4			
6	289	IND#: 111871	3	268	IND#: 111871	4			
7	569	IND#: 111871	3	268	IND#: 111871	4			
8	466	IND#: 109678	3	254	IND#: 109678	4			
9	588	IND#: 70125	3	147	IND#: 70125	4			
10	587	IND#: 70125	3	147	IND#: 70125	4			
11	581	NDA#: 22405	3	144	NDA#: 22405	4			
12	581	NDA#: 22405	3	538	NDA#: 22405	4			
13	572	NDA#: 203756	3	424	NDA#: 203756	4			
14	572	NDA#: 203756	3	124	NDA#: 203756	4			
15	573	NDA#: 203756	3	424	NDA#: 203756	4			
16	573	NDA#: 203756	3	124	NDA#: 203756	4			
17	499	NDA#: 21894	3	430	NDA#: 21894	4			
18	499	NDA#: 21894	3	866	NDA#: 21894	4			
19	566	IND#: 110180	3	117	IND#: 110180	4			
20	505	IND#: 73916	3	139	IND#: 73916	4			
21	308	IND#: 115670	3	141	IND#: 115670	4			
22	482	IND#: 115670	3	141	IND#: 115670	4			
23	571	IND#: 112796	3	163	IND#: 112796	4			
24	571	IND#: 112796	3	294	IND#: 112796	4			
25	571	IND#: 112796	3	293	IND#: 112796	4			
26	320	IND#: 117352	3	171	IND#: 117352	4			
27	81	IND#: 132547	3	185	IND#: 132547	4			
28	81	IND#: 132547	3	185	IND#: 132547	4			

Figure 11: Matching Application Number Table after Normalization

• Data Secondary Accuracy Measurement

Here, the Pearson correlation method was returning the exact same response as for the previous table. This is comprehensible since correlation between “MEETING_FILE_TYPE 3” and “MEETING_FILE_TYPE 4” is not affected by how clean “APPLICATION_NUMBER” is. It just used “APPLICATION_NUMBER” as a medium interpreter to show that “MEETING_FILE_TYPE 3” and “MEETING_FILE_TYPE 4” are varying at the same point.

So, I went with a different approach and used the scatter plot in excel to build different scatter plot graphs to compare the variation of the “MEETING_FILE_TYPE 3” and “MEETING_FILE_TYPE 4” using their matching “APPLICATION_NUMBER”.

Based on Figure 12 below, we can constate that “MEETING_FILE_TYPE 3” and “MEETING_FILE_TYPE 4” have basically the same variation, which prove they have a strong correlation. Also, we can see that the blue variation, the few matching “APPLICATION_NUMBER” I obtained from the first dataset, are less than the orange ones which are the variation of the matching “APPLICATION_NUMBER” from the clean dataset. This makes sense because while cleaning the dataset, I have added new “APPLICATION_NUMBER” while separating old ones into 2 set of “APPLICATION_NUMBER”. This ended up giving more matching “APPLICATION_NUMBER” than the previous one from the first dataset.

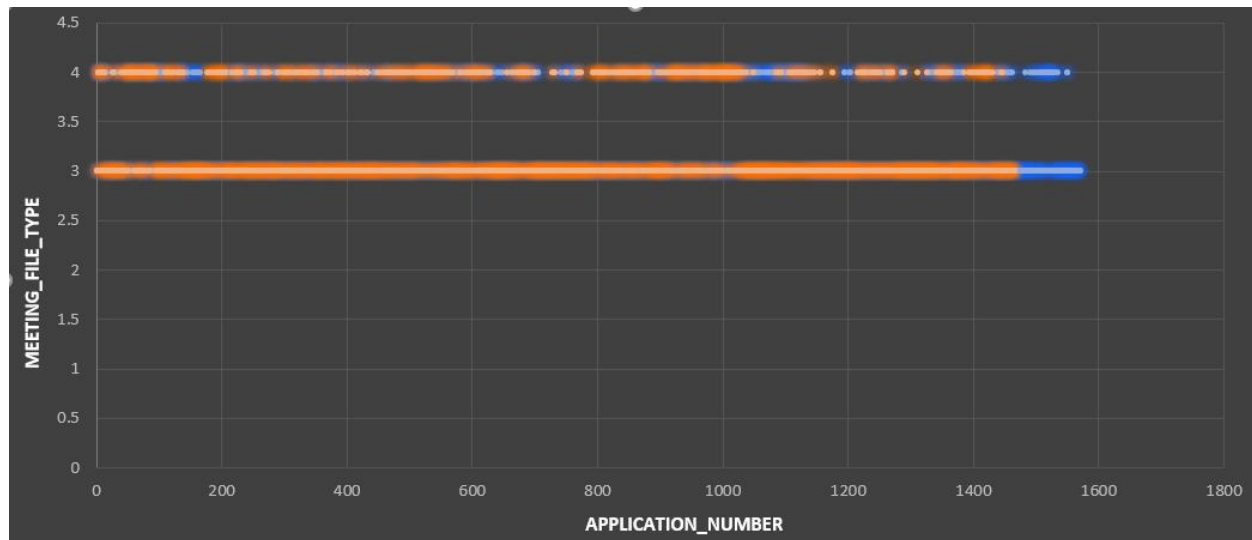


Figure 12: Scatter Chart for Matching Application Number

- Data Secondary Accuracy Metric

Let's apply the data accuracy metric formula one more time to see if it aligns with all the previous findings.

Measure of Accuracy Table 2 = 1 – Number of Matching Application numbers / Total Application numbers

$$= 1 - (91 / 1467)$$

$$= 0.9379$$

This number is still relatively close to 1, so using their matching "APPLICATION_NUMBER", we can definitively conclude that there is a strong dependency between "MEETING_FILE_TYPE 3" and "MEETING_FILE_TYPE 4".

Conclusion

The priority of this project was to access the "MEETING_FILE_TYPE 3", protocol procedure that had "MEETING_FILE_TYPE 4", final study for each meeting minutes documents. To execute this search, I had to find a common attribute to link them, which was "APPLICATION_NUMBER". Since, each meeting minutes document had its "APPLICATION_NUMBER", and "MEETING_FILE_TYPE 3" and "MEETING_FILE_TYPE 4" have matching "APPLICATION_NUMBER", it should have been easy to execute this task. However, the most important part of the project was to improve the data quality of the matching "APPLICATION_NUMBER".

Throughout this project, the Python and Excel software were great resources that helped me acquire all the matching "APPLICATION_NUMBER" for "MEETING_FILE_TYPE 3" and "MEETING_FILE_TYPE 4". The extracted matching "APPLICATION_NUMBER" had several flaws such as format inconsistency, incompleteness and invalid data. I had to analyze and understand the present issues and come up with adequate solutions to solve them. All the

analysis procedures and statistical analysis of the metrics helped me ensure that the matching “APPLICATION_NUMBER” reached the highest quality of data they could possibly reach.

All these will provide tremendous support to the end goal project of the ECAC in their quest of building a new dashboard system. Also, this project can be use in the future as a blueprint reference to enhance the data quality of the matching “APPLICATION_NUMBER”. In the future, this will reduce the time required to ECAC to understand the data before it can add value and benefits to the organization.

Acknowledgement

I want to take this time to thank my mentors, Dr. Xu Joshua, Dr. Jae Hyun and Meehan Joe for their support throughout this project. They never stopped guiding me and providing me constant support throughout this entire project. They gave me a chance to work on a project that I was passionate about which improved my knowledge in my field of study. I would also like to thank The Sponsor for my project, the Executive Carcinogenicity Assessment Committee (ECAC) for providing me all the resources necessary for this project. To all these people that gave me everything necessary in timely matter so that I can reach my project guidelines and timelines, for their valuable guidance and immense support, you have my sincere gratitude.

References

- [1]. “Management of CDER Executive Carcinogenicity Assessment Committee and Communication of Committee Proceedings”. Associate Director for Pharmacology and Toxicology; Office of New Drugs. Print.
- [2]. “pandas.read_xml”. *Pandas*,
https://pandas.pydata.org/docs/reference/api/pandas.read_xml.html.
- [3]. Juviler, Jamie. “XML Files: What They Are & How to Open Them”. *HubSpot*,
<https://blog.hubspot.com/website/what-is-xml-file>.
- [4]. “What is a Relational Database (RDBMS)?”. *OCI*,
<https://www.oracle.com/database/what-is-a-relational-database/>.
- [5]. Nettleton, David. “Pearson Correlation”. *ScienceDirect*, 2014,
<https://www.sciencedirect.com/topics/computer-science/pearson-correlation#:~:text=The%20Pearson%20correlation%20measures%20the,meaning%20a%20total%20positive%20correlation>.
- [6]. Sebastian, Laura. “Data Quality Metric”. *ScienceDirect*, 2013,
<https://www.sciencedirect.com/topics/computer-science/data-quality-metric>.
- [7]. Kim, Farah. “What is Data Accuracy, why it matters and How Companies Can Ensure They Have Accurate Data”. *Data Ladder*, 25 Sept 2020, <https://dataladder.com/what-is-data-accuracy/>.
- [8]. Ladson, J.A. “Measurement Accuracy”. *ScienceDirect*, 2010,
<https://www.sciencedirect.com/topics/engineering/measurement-accuracy>.