Predicting the trends of Azithromycin consumption for the treatment of infectious diseases in the USA by 2025.

Sai Sravanthi Kilari

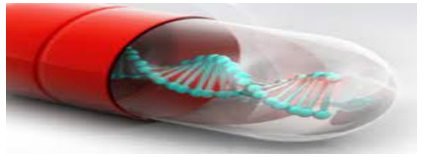
Introduction

Azithromycin is a broad-spectrum antibiotic prescribed for many bacterial infections, majorly in pulmonary conditions, genital, and enteric infections.

Azithromycin resistance is a major concern in the current situation. Because, as mentioned in “rationale for Azithromycin in covid-19”, there is a significant rise in the usage of Azithromycin due to post and present COVID-19 complications such as respiratory distress, throat infections.

Past studies like “Mass distribution of Azithromycin for trachoma” have shown that excessive usage of Azithromycin causes resistance for specific bacteria like Escherichia coli, streptococcus pneumonia and many other gram negative bacteria that cause several infections .

The present scenario is exhibiting mass usage of Azithromycin that may repeat similar situation. So, we analyse and interpret the data to predict the future Azithromycin resistance and consumption.



Aim

This study aims to determine the various unitigs of the bacterium Neisseria Gonorrhoeae that contribute to increase or decrease the Azithromycin consumption for the treatment of infectious diseases, using various genomic and csv data analysis tools and visualization techniques, and predict the trends of Azithromycin consumption in the USA by 2025.

Purpose

The purpose of the study is to identify, test, and investigate different unitigs of the Bacterium Neisseria Gonorrhoea associated with Azithromycin resistance and predict the increase or decrease in the consumption of Azithromycin to treat bacterial infections by 2025 in the United States.

This study may helps understand about the impact of Azithromycin usage in infectious diseases.

Research Hypothesis

Null Hypothesis: There is no association among the factors tested and cannot predict the consumption of Azithromycin.

Alternate Hypothesis: There is a significant association among the factors tested and can predict the consumption of Azithromycin.

PROCESS OVERVIEW

- **DATA COLLECTION**

  Predicting antibiotic resistance in gonorrhea.

  https://www.kaggle.com/nwheeler443/gono-unitigs

  It is the resistance exhibited by different strains of Neisseria Gonorrhoeae to various antibiotics in different countries from 1979-2015.

- **DATA CLEANING, EXTRACTION**

  Finding missing values

  Finding Correlation

- **DATA EXPLORATION**

  Visualisation of the data

  Normality test

- **METHODOLOGY**

  Developing a Model

  Performance analysis

- **STATISTICAL TESTING**

  MANCOVA

- **RESULTS**

DATA COLLECTION AND DESCRIPTION

Metadata:

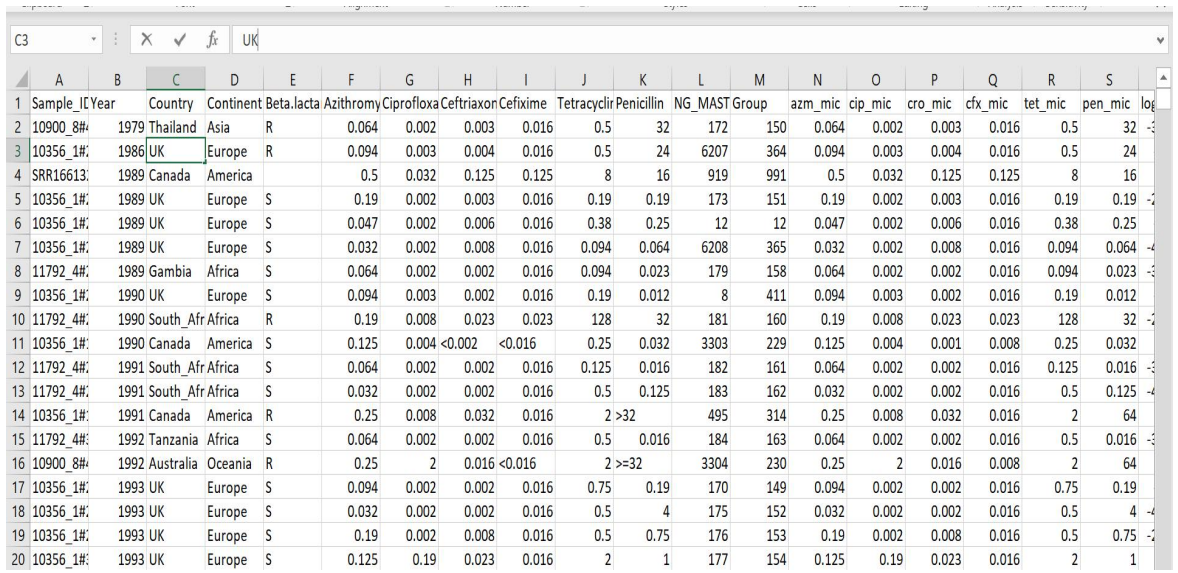
consists of 3700 rows and 31 columns.
columns related to Azithromycin were retained.

GWAS(Genome Wide Association Studies) Data:

consists of unitigs of Neisseria Gonorrhoeae bacteria:

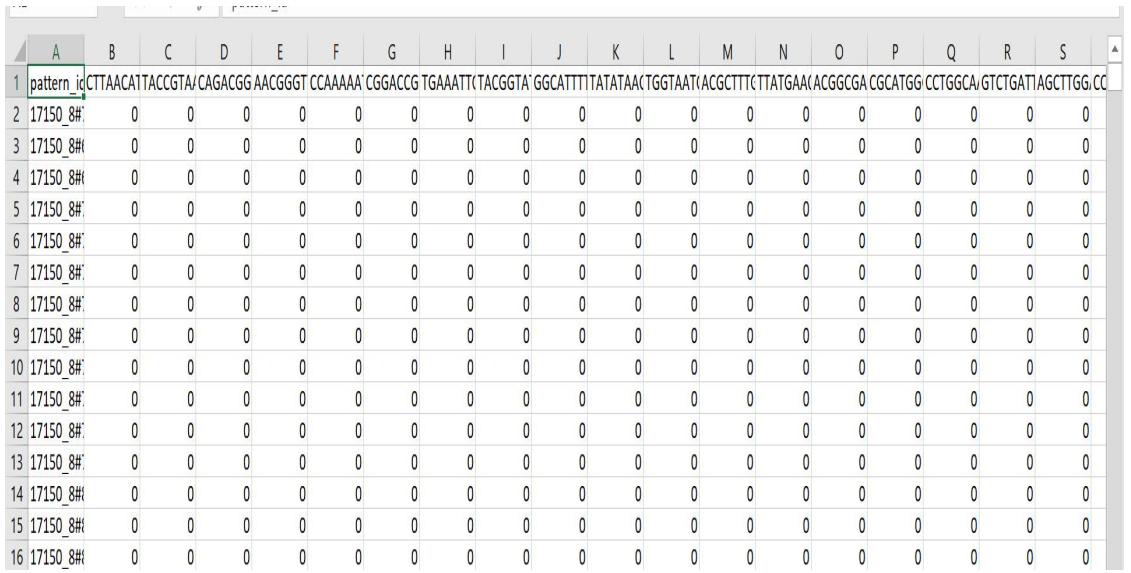
0 - mutation absent
1 -mutation present

Changed from Rtab to csv file and is uploaded to the jupyter notebook of python



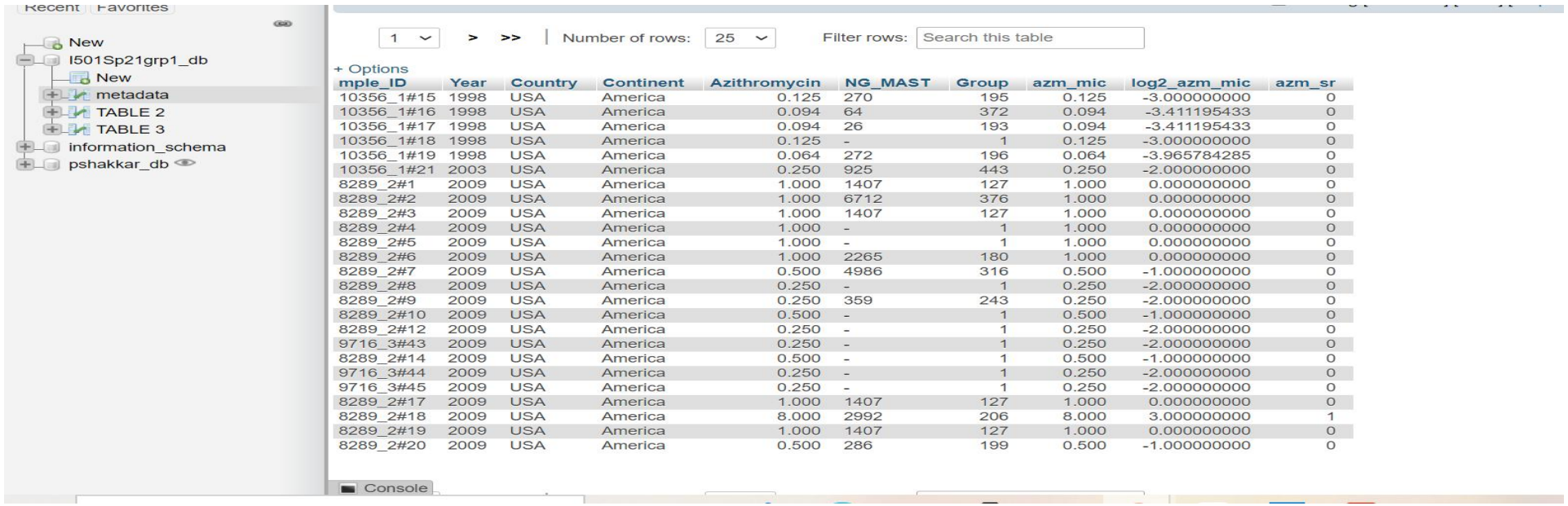
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	Sample_ID	Year	Country	Continent	Beta.lacta	Azithromy	Ciprofloxa	Ceftriaxon	Cefixime	Tetracyclir	Penicillin	NG_MAST	Group	azm_mic	cip_mic	cro_mic	cfx_mic	tet_mic	pen_mic
2	10900_8#	1979	Thailand	Asia	R	0.064	0.002	0.003	0.016	0.5	32	172	150	0.064	0.002	0.003	0.016	0.5	32
3	10356_1#	1986	UK	Europe	R	0.094	0.003	0.004	0.016	0.5	24	6207	364	0.094	0.003	0.004	0.016	0.5	24
4	SRR16613	1989	Canada	America		0.5	0.032	0.125	0.125	8	16	919	991	0.5	0.032	0.125	0.125	8	16
5	10356_1#	1989	UK	Europe	S	0.19	0.002	0.003	0.016	0.19	0.19	173	151	0.19	0.002	0.003	0.016	0.19	0.19
6	10356_1#	1989	UK	Europe	S	0.047	0.002	0.006	0.016	0.38	0.25	12	12	0.047	0.002	0.006	0.016	0.38	0.25
7	10356_1#	1989	UK	Europe	S	0.032	0.002	0.008	0.016	0.094	0.064	6208	365	0.032	0.002	0.008	0.016	0.094	0.064
8	11792_4#	1989	Gambia	Africa	S	0.064	0.002	0.002	0.016	0.094	0.023	179	158	0.064	0.002	0.002	0.016	0.094	0.023
9	10356_1#	1990	UK	Europe	S	0.094	0.003	0.002	0.016	0.19	0.012	8	411	0.094	0.003	0.002	0.016	0.19	0.012
10	11792_4#	1990	South_Afr	Africa	R	0.19	0.008	0.023	0.023	128	32	181	160	0.19	0.008	0.023	0.023	128	32
11	10356_1#	1990	Canada	America	S	0.125	0.004	<0.002	<0.016	0.25	0.032	3303	229	0.125	0.004	0.001	0.008	0.25	0.032
12	11792_4#	1991	South_Afr	Africa	S	0.064	0.002	0.002	0.016	0.125	0.016	182	161	0.064	0.002	0.002	0.016	0.125	0.016
13	11792_4#	1991	South_Afr	Africa	S	0.032	0.002	0.002	0.016	0.5	0.125	183	162	0.032	0.002	0.002	0.016	0.5	0.125
14	10356_1#	1991	Canada	America	R	0.25	0.008	0.032	0.016	2	>32	495	314	0.25	0.008	0.032	0.016	2	64
15	11792_4#	1992	Tanzania	Africa	S	0.064	0.002	0.002	0.016	0.5	0.016	184	163	0.064	0.002	0.002	0.016	0.5	0.016
16	10900_8#	1992	Australia	Oceania	R	0.25	2	0.016	<0.016	2	>=32	3304	230	0.25	2	0.016	0.008	2	64
17	10356_1#	1993	UK	Europe	S	0.094	0.002	0.002	0.016	0.75	0.19	170	149	0.094	0.002	0.002	0.016	0.75	0.19
18	10356_1#	1993	UK	Europe	S	0.032	0.002	0.002	0.016	0.5	4	175	152	0.032	0.002	0.002	0.016	0.5	4
19	10356_1#	1993	UK	Europe	S	0.19	0.002	0.008	0.016	0.5	0.75	176	153	0.19	0.002	0.008	0.016	0.5	0.75
20	10356_1#	1993	UK	Europe	S	0.125	0.19	0.023	0.016	2	1	177	154	0.125	0.19	0.023	0.016	2	1

Original metadata



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	pattern_id	CTTAACAT	TACCGTA/	CAGACGG	AACGGGT	CCAAAAA	CGGACCG	TGAAATT(	TACGGTA	GGCATTT	TATATAA(	TGGTAAT(	ACGCTTT(	TTATGAA(	ACGGCGA	CGCATGG	CCTGGCA/	GTCTGAT	AGCTTGG/	CC
2	17150_8#7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	17150_8#(	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	17150_8#(	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	17150_8#7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	17150_8#7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	17150_8#7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	17150_8#7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	17150_8#7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	17150_8#7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	17150_8#7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	17150_8#7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	17150_8#7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	17150_8#8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	17150_8#8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	17150_8#8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

GWAS data



mple_ID	Year	Country	Continent	Azithromycin	NG_MAST	Group	azm_mic	log2_azm_mic	azm_sr
10356_1#15	1998	USA	America	0.125	270	195	0.125	-3.000000000	0
10356_1#16	1998	USA	America	0.094	64	372	0.094	-3.411195433	0
10356_1#17	1998	USA	America	0.094	26	193	0.094	-3.411195433	0
10356_1#18	1998	USA	America	0.125	-	1	0.125	-3.000000000	0
10356_1#19	1998	USA	America	0.064	272	196	0.064	-3.965784285	0
10356_1#21	2003	USA	America	0.250	925	443	0.250	-2.000000000	0
8289_2#1	2009	USA	America	1.000	1407	127	1.000	0.000000000	0
8289_2#2	2009	USA	America	1.000	6712	376	1.000	0.000000000	0
8289_2#3	2009	USA	America	1.000	1407	127	1.000	0.000000000	0
8289_2#4	2009	USA	America	1.000	-	1	1.000	0.000000000	0
8289_2#5	2009	USA	America	1.000	-	1	1.000	0.000000000	0
8289_2#6	2009	USA	America	1.000	2265	180	1.000	0.000000000	0
8289_2#7	2009	USA	America	0.500	4986	316	0.500	-1.000000000	0
8289_2#8	2009	USA	America	0.250	-	1	0.250	-2.000000000	0
8289_2#9	2009	USA	America	0.250	359	243	0.250	-2.000000000	0
8289_2#10	2009	USA	America	0.500	-	1	0.500	-1.000000000	0
8289_2#12	2009	USA	America	0.250	-	1	0.250	-2.000000000	0
9716_3#43	2009	USA	America	0.250	-	1	0.250	-2.000000000	0
8289_2#14	2009	USA	America	0.500	-	1	0.500	-1.000000000	0
9716_3#44	2009	USA	America	0.250	-	1	0.250	-2.000000000	0
9716_3#45	2009	USA	America	0.250	-	1	0.250	-2.000000000	0
8289_2#17	2009	USA	America	1.000	1407	127	1.000	0.000000000	0
8289_2#18	2009	USA	America	8.000	2992	206	8.000	3.000000000	1
8289_2#19	2009	USA	America	1.000	1407	127	1.000	0.000000000	0
8289_2#20	2009	USA	America	0.500	286	199	0.500	-1.000000000	0

Metadata imported to MYSQL database.

Importing Data from SQL

```
pip install tabulate
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: tabulate in ./.local/lib/python3.8/site-packages (0.8.9)
Note: you may need to restart the kernel to use updated packages.
```

```
import MySQLdb
import getpass
username = input('enter the username:')
password = getpass.getpass(prompt="enter password")
conn = MySQLdb.connect(host="localhost", user= username, passwd=password, db= 'I501Sp21grp1_db')
cursor = conn.cursor()

cursor.execute("select * from metadata")
rows = cursor.fetchall()
headers = [i[0] for i in cursor.description]
headers[0] = 'Sample_ID'
data_dict = {}

for row in rows:
    for i in range(len(row)):
        if not headers[i] in data_dict:
            data_dict[headers[i]] = [row[i]]
        else:
            data_dict[headers[i]].append(row[i])

import pandas as pd
metadata_df = pd.DataFrame.from_dict(data_dict)
metadata_df= metadata_df.set_index('Sample_ID')
metadata_df

nan_value = float("NaN")
metadata_df['NG_MAST']=metadata_df['NG_MAST'].replace(to_replace="-", value = nan_value)

from tabulate import tabulate
metadata_df = metadata_df.dropna()
tabulate(metadata_df, headers=headers, tablefmt='psql')
```

DATA CLEANING AND ENHANCEMENT

Out[89]: <AxesSubplot:>

Correlation matrix

```
from tabulate import tabulate
metadata_df = metadata_df.dropna()

metadata_df=metadata_df.drop(["Country" , "Continent", "Group", "azm_mic", "NG_MAST"], axis=1)


metadata_headers = ['Sample_ID', 'Year', 'Azithromycin', 'Log2(Azithromycin)', 'azm_sr']

metadata_df = metadata_df.sort_values(by='Year')
print(tabulate(metadata_df, headers=metadata_headers, tablefmt='grid'))
```

```
+------------+--------+---------------+---------------------+----------+
| Sample_ID  |   Year |   Azithromycin |   Log2(Azithromycin) |   azm_sr |
+============+========+===============+=====================+==========+
| 10356_1#15 |   1998 |         0.125 |              -3     |        0 |
+------------+--------+---------------+---------------------+----------+
| 10356_1#16 |   1998 |         0.094 |              -3.4112 |        0 |
+------------+--------+---------------+---------------------+----------+
| 10356_1#17 |   1998 |         0.094 |              -3.4112 |        0 |
+------------+--------+---------------+---------------------+----------+
| 10356_1#19 |   1998 |         0.064 |              -3.96578 |        0 |
+------------+--------+---------------+---------------------+----------+
| 10356_1#20 |   1999 |         4     |               2     |        1 |
+------------+--------+---------------+---------------------+----------+
| 17150_8#68 |   2000 |         1     |               0     |        0 |
+------------+--------+---------------+---------------------+----------+
| 17176_1#72 |   2000 |         0.25  |              -2     |        0 |
+------------+--------+---------------+---------------------+----------+
| 17150_8#83 |   2000 |         2     |               1     |        0 |
+------------+--------+---------------+---------------------+----------+
| 17176_1#70 |   2000 |         0.25  |              -2     |        0 |
```

Drop columns Country, Continent since all the values are same.
Drop column azm_mic since it is same as column Azithromycin.

DATA EXPLORATION

DESCRIPTIVE STATISTICS

```
enter password·····
```

```
In [3]: metadata_df.info()
```

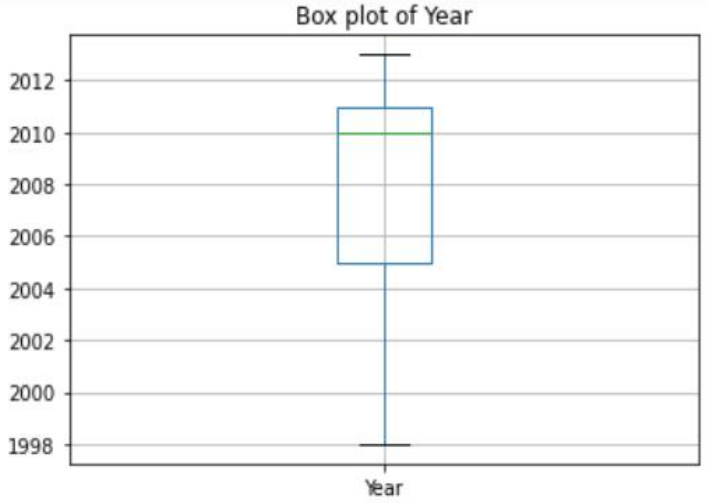
```
<class 'pandas.core.frame.DataFrame'>
Index: 1121 entries, 10356_1#15 to 8727_5#89
Data columns (total 9 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   Year          1121 non-null   int64
 1   Country       1121 non-null   object
 2   Continent     1121 non-null   object
 3   Azithromycin  1121 non-null   object
 4   NG_MAST       1121 non-null   object
 5   Group         1121 non-null   int64
 6   azm_mic       1121 non-null   object
 7   log2_azm_mic  1121 non-null   object
 8   azm_sr        1121 non-null   int64
dtypes: int64(3), object(6)
memory usage: 87.6+ KB
```

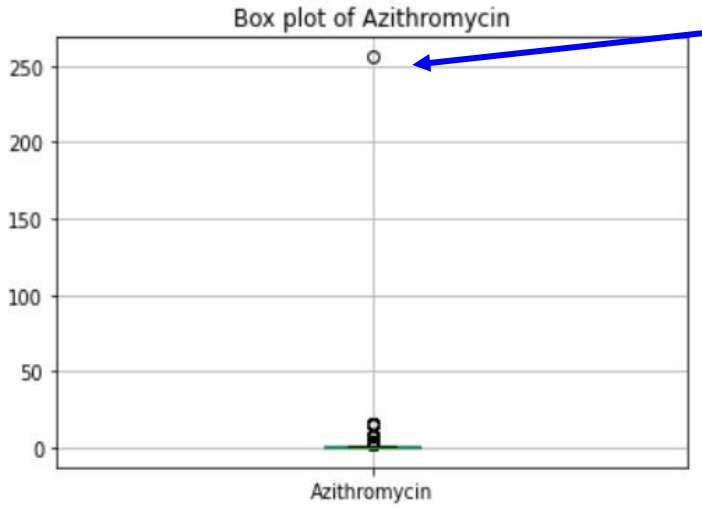
```
In [88]: metadata_df.describe()
```

Out[88]:

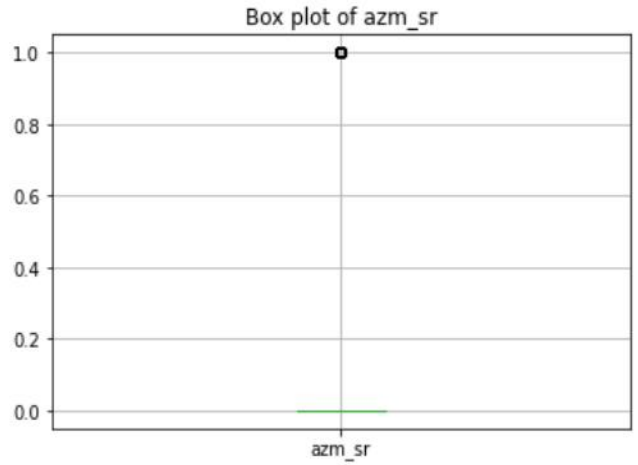
	Year	Azithromycin	Group	Log2(Azithromycin)	azm_sr
count	1121.000000	1121.000000	1121.000000	1121.000000	1121.000000
mean	2008.183764	1.748425	247.394291	-0.524671	0.109723
std	3.811887	8.115535	229.524135	1.652990	0.312684
min	1998.000000	0.030000	1.000000	-5.058894	0.000000
25%	2005.000000	0.250000	127.000000	-2.000000	0.000000
50%	2009.000000	0.500000	191.000000	-1.000000	0.000000
75%	2011.000000	1.000000	372.000000	0.000000	0.000000
max	2013.000000	256.000000	1004.000000	8.000000	1.000000

DATA VISUALIZATION OF CLEANED METADATA

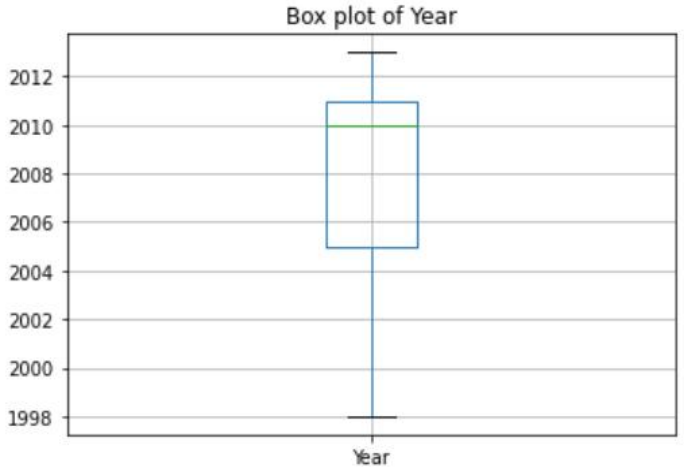


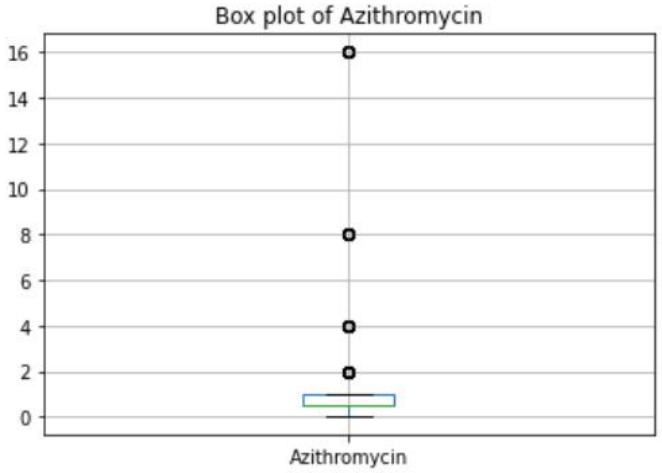


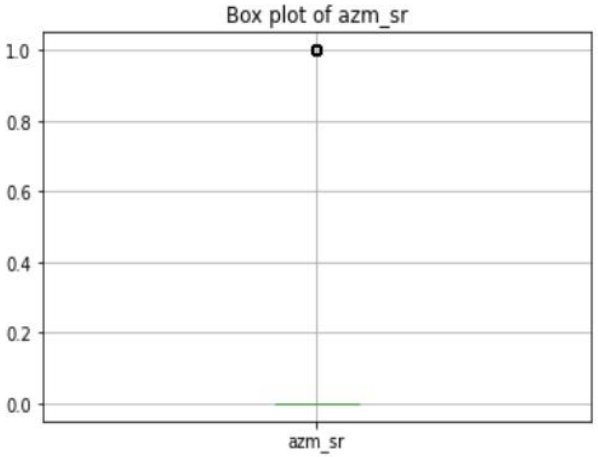
An outlier is observed.



After removal of outlier

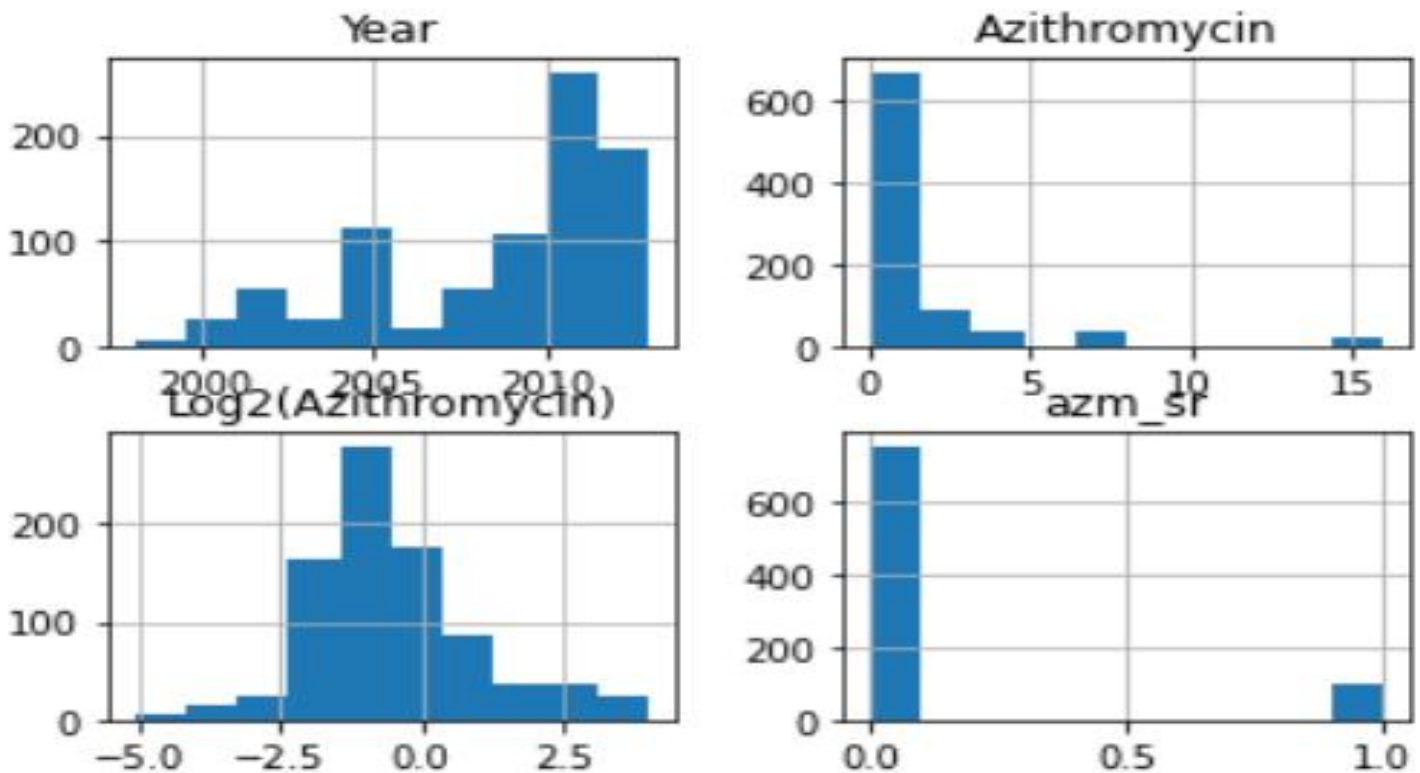


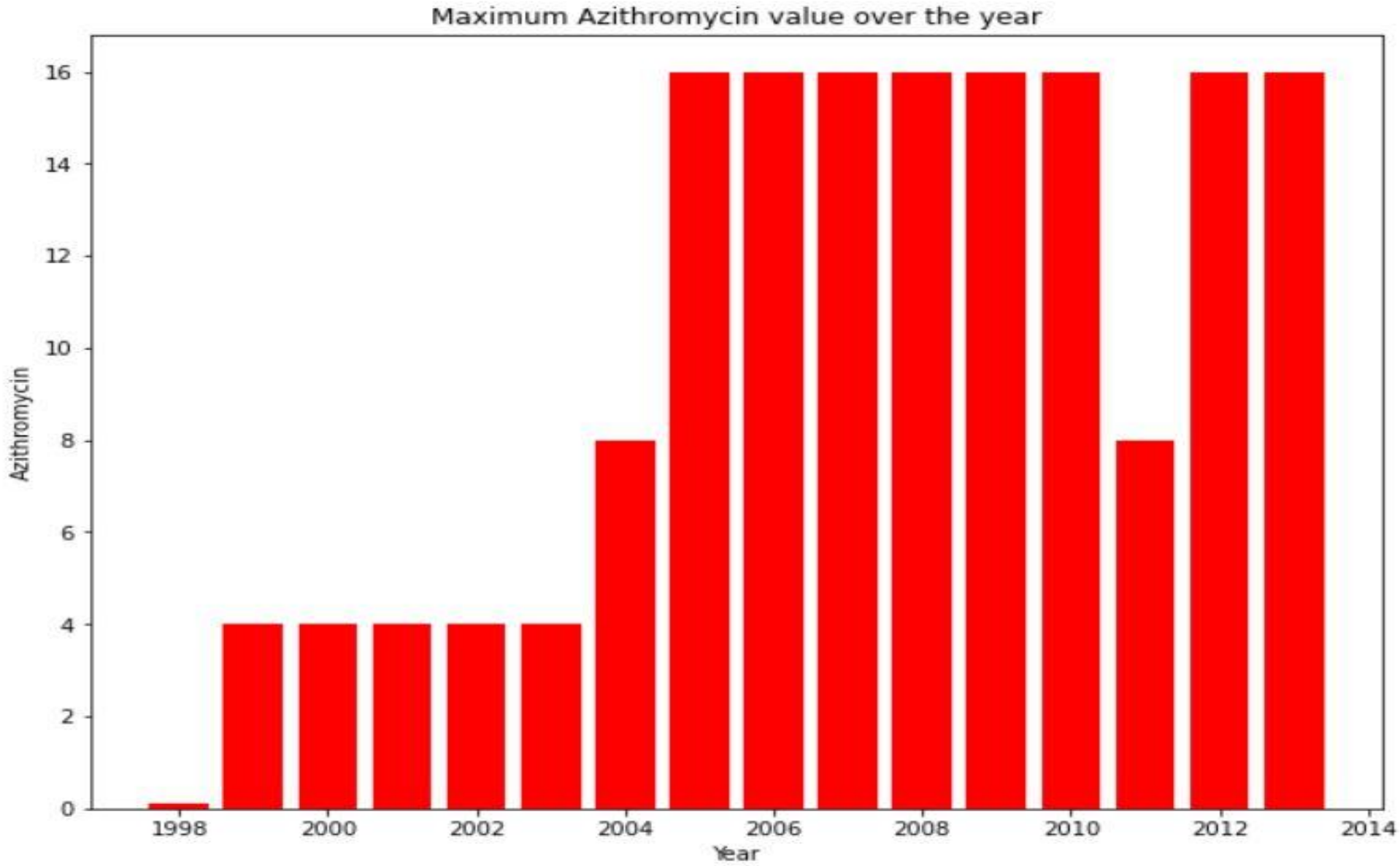




In [12]:
```
metadata_df.hist()
```

Out[12]:
```
array([[<AxesSubplot:title={'center':'Year'}>,
        <AxesSubplot:title={'center':'Azithromycin'}>],
       [<AxesSubplot:title={'center':'Log2(Azithromycin)'}>,
        <AxesSubplot:title={'center':'azm_sr'}>]], dtype=object)
```



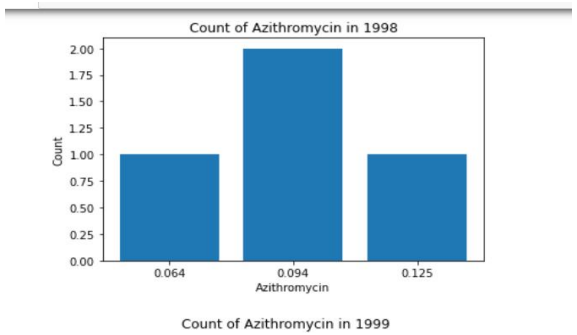

Maximum Azithromycin value over the year
Azithromycin
0
2
4
6
8
10
12
14
16
1998
2000
2002
2004
2006
2008
2010
2012
2014
Year

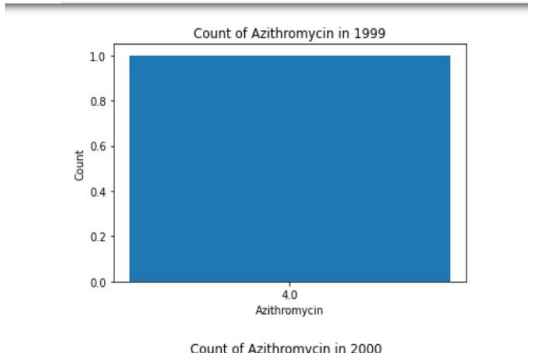
Normality test of the cleaned data

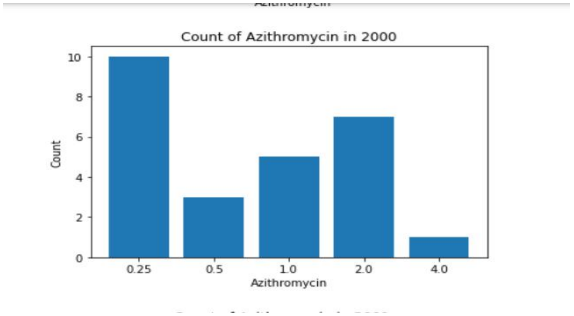
In [53]:
```
# Normality tests
from scipy import stats

# null hypothesis: columns of metadata.csv are normally distributed

for col in list(metadata_df.columns):
    stat, pval = stats.normaltest(metadata_df[col])
    if pval >= 0.05:
    print(pval,'The null hypothesis that the column \"%s\" of metadata.csv are normally distributed is ACCEPTED.' %(col))
    else:
    print(pval,'The null hypothesis that the column \"%s\" of metadata.csv are normally distributed is REJECTED.' %(col))
```

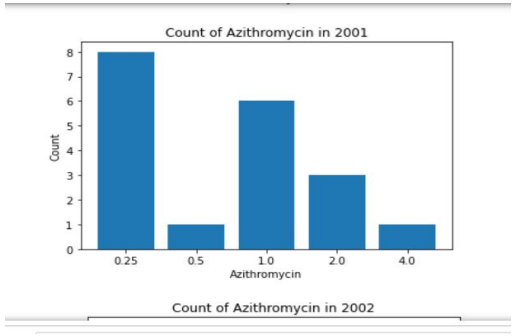
```
6.343360626046418e-21 The null hypothesis that the column "Year" of metadata.csv are normally distributed is REJECTED.
3.6970454395592264e-143 The null hypothesis that the column "Azithromycin" of metadata.csv are normally distributed is REJECTE
D.
4.540637521758421e-14 The null hypothesis that the column "Log2(Azithromycin)" of metadata.csv are normally distributed is REJE
CTED.
1.0618144834258459e-87 The null hypothesis that the column "azm_sr" of metadata.csv are normally distributed is REJECTED.
```

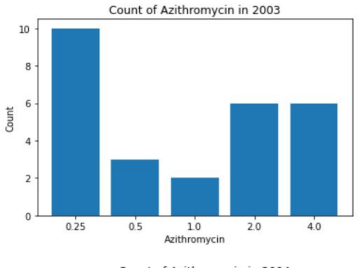
Count of samples per year per Azithromycin

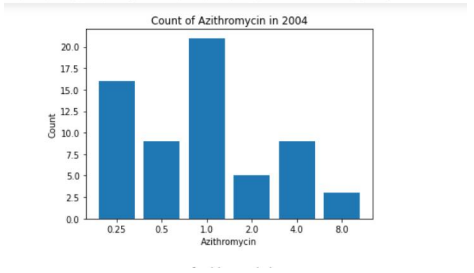


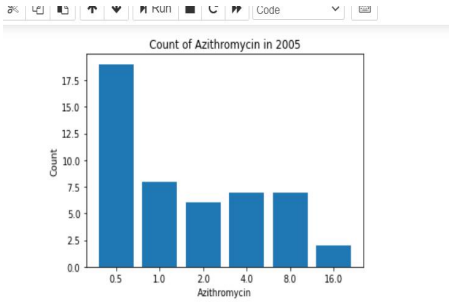


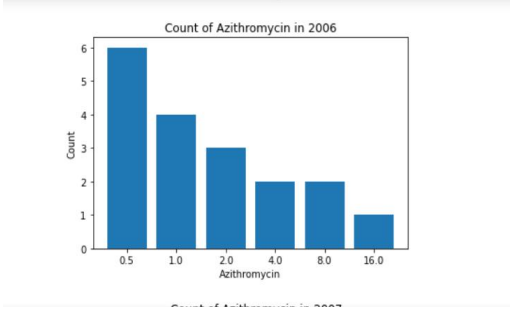


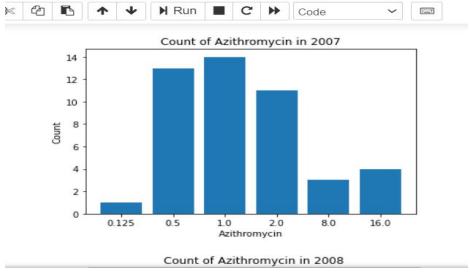








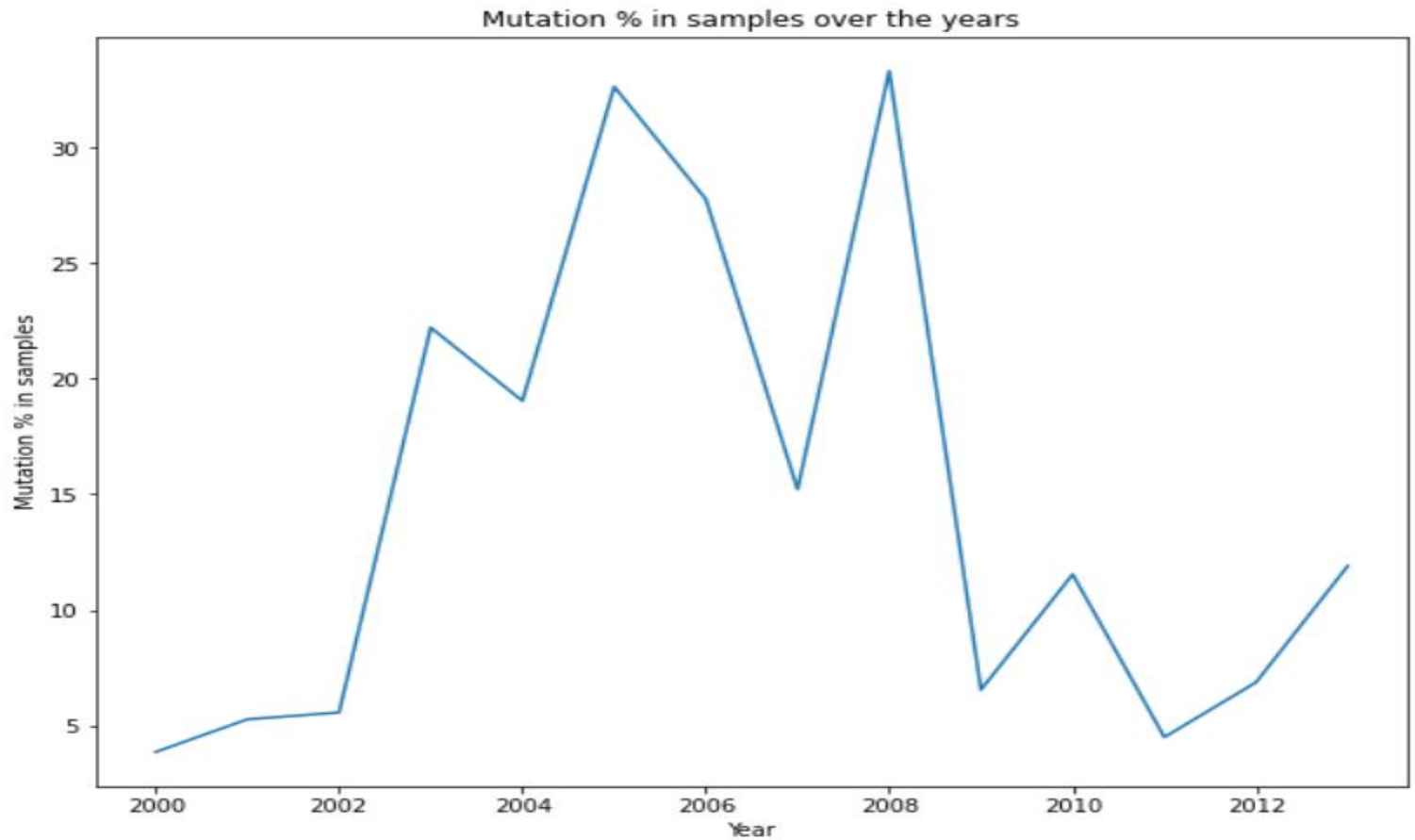


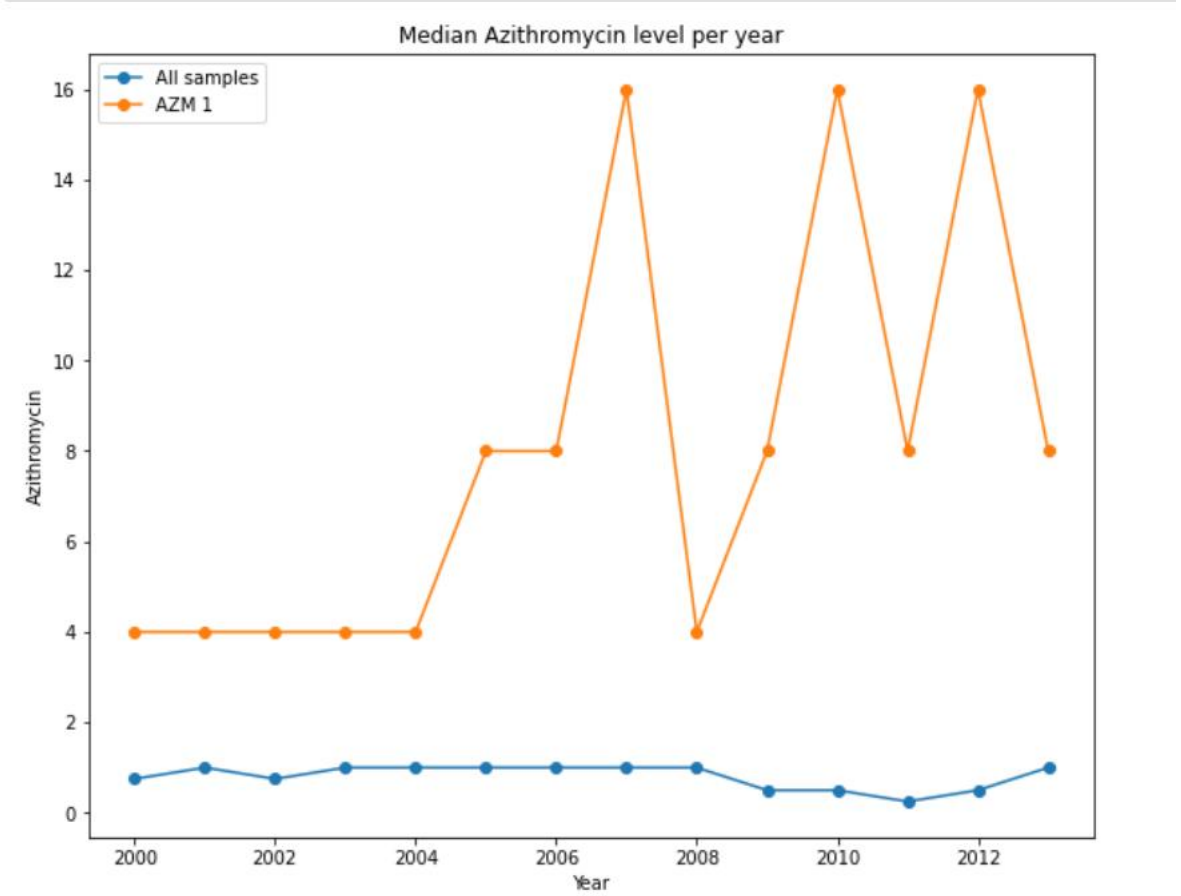


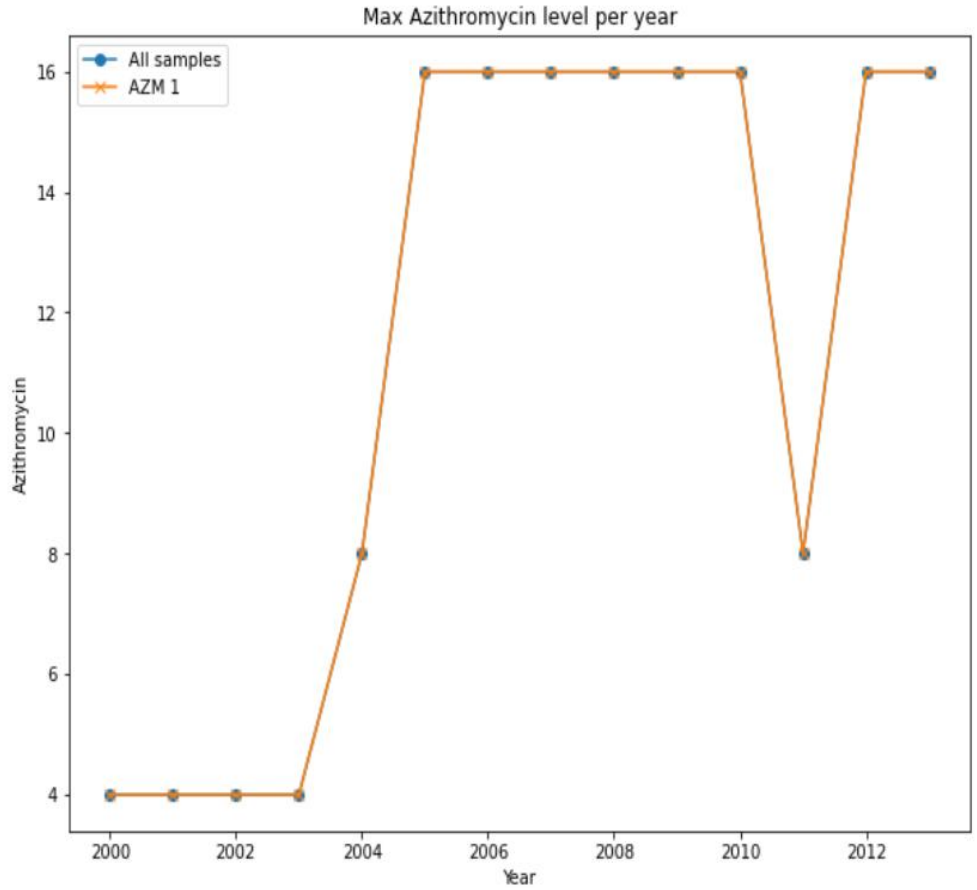
Percentage of mutations and non-mutations per year

```
metadata_year_df = metadata_df.copy()
metadata_year_df['Tot_count'] = 1
metadata_year_df = metadata_year_df.groupby(['Year'], as_index=False).sum()
metadata_year_df = metadata_year_df.drop(columns=['Azithromycin', 'Log2(Azithromycin)', 'azm_sr'])
metadata_year_df = metadata_year_df[metadata_year_df['Tot_count']>5]

metadata_year_azm_df = metadata_df.copy()
metadata_year_azm_df['Count'] = 1
metadata_year_azm_df = metadata_year_azm_df.groupby(['Year','azm_sr'], as_index=False).sum()
metadata_year_azm_df = metadata_year_azm_df.drop(columns=['Azithromycin', 'Log2(Azithromycin)'])
metadata_year_azm_df = pd.merge(metadata_year_azm_df, metadata_year_df, on='Year')
metadata_year_azm_df['%'] = metadata_year_azm_df['Count']*100/metadata_year_azm_df['Tot_count']

print(tabulate(metadata_year_azm_df, headers=metadata_year_azm_df.columns, tablefmt='grid'))

metadata_year_azm_1_perc_dict = {}

for i in range(len(metadata_year_azm_df['Year'])):
    if (metadata_year_azm_df['azm_sr'][i] == 1):
        year = metadata_year_azm_df['Year'][i]
        metadata_year_azm_1_perc_dict[year] = metadata_year_azm_df['%'][i]

figure(figsize=(10, 8))
plt.plot(list(metadata_year_azm_1_perc_dict.keys()), list(metadata_year_azm_1_perc_dict.values()))
plt.xlabel("Year")
plt.ylabel("Mutation % in samples")
plt.title("Mutation % in samples over the years")
plt.show()
```

	Year	azm_sr	Count	Tot_count	%
0	2000	0	25	26	96.1538
1	2000	1	1	26	3.84615
2	2001	0	18	19	94.7368
3	2001	1	1	19	5.26316
4	2002	0	34	36	94.4444
5	2002	1	2	36	5.55556
6	2003	0	21	27	77.7778
7	2003	1	6	27	22.2222
8	2004	0	51	63	80.9524
9	2004	1	12	63	19.0476
10	2005	0	33	49	67.3469
11	2005	1	16	49	32.6531
12	2006	0	13	18	72.2222
13	2006	1	5	18	27.7778
14	2007	0	39	46	84.7826
15	2007	1	7	46	15.2174


Mutation % in samples over the years
Mutation % in samples
30
25
20
15
10
5
2000
2002
2004
2006
2008
2010
2012
Year


Median Azithromycin level per year
All samples
AZM 1
Azithromycin
Year
0
2
4
6
8
10
12
14
16
2000
2002
2004
2006
2008
2010
2012


Max Azithromycin level per year
All samples
AZM 1
Azithromycin
Year
4
6
8
10
12
14
16
2000
2002
2004
2006
2008
2010
2012

METHODOLOGY

GWAS data description and merging of GWAS data with Metadata

In [22]:
```
gwas_data = pd.read_csv("gwas.csv")
gwas_data
```

Out[22]:

	pattern_id	CTTAACATATTTGCCTTTGATTTTTGAAGAAGCTGCCACGCCGGCAG	TACCGTAACCGGCAATGCGGATATTACGGTC	CAGACGGCATTTTTTTTGCGTTT
0	17150_8#77	0	0	
1	17150_8#68	0	0	
2	17150_8#69	0	0	
3	17150_8#70	0	0	
4	17150_8#71	0	0	
...	...	...	...	
1480	SRR2736302	0	0	
1481	SRR2736303	0	0	
1482	SRR2736304	0	0	
1483	SRR2736305	0	0	
1484	SRR2736306	0	0	

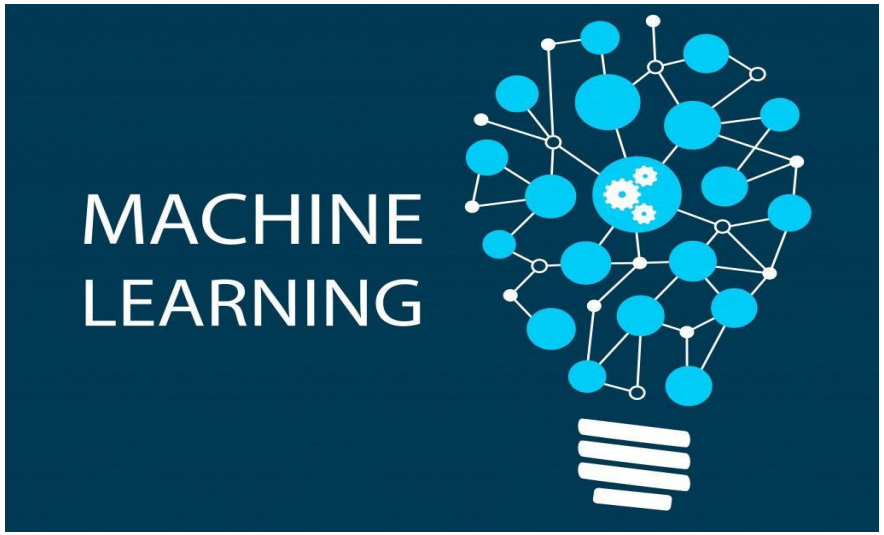
1485 rows × 516 columns

In [24]:
```
meta_gwas_data = gwas_data.copy()
meta_gwas_data = meta_gwas_data.rename(columns={'pattern_id':'Sample_ID'})
meta_gwas_data = pd.merge(metadata_df, meta_gwas_data, on='Sample_ID')
print(meta_gwas_data)
```

```
     Sample_ID  Year  Azithromycin  Log2(Azithromycin)  azm_sr  \
0    17150_8#68  2000          1.00                 0.0       0
1    17176_1#72  2000          0.25                -2.0       0
2    17150_8#83  2000          2.00                 1.0       0
3    17176_1#70  2000          0.25                -2.0       0
4    17176_1#58  2000          0.25                -2.0       0
..          ...   ...           ...                 ...     ...
840  17176_1#16  2013          0.25                -2.0       0
841  15335_7#53  2013          2.00                 1.0       0
842  16043_2#18  2013          0.50                -1.0       0
843  15335_7#41  2013          8.00                 3.0       1
844   17176_1#4  2013          0.25                -2.0       0

     CTTAACATATTTGCCTTTGATTTTTGAAGAAGCTGCCACGCCGGCAG  \
0                                                  0
1                                                  0
2                                                  0
3                                                  0
4                                                  0
..                                               ...
```

Classification Models

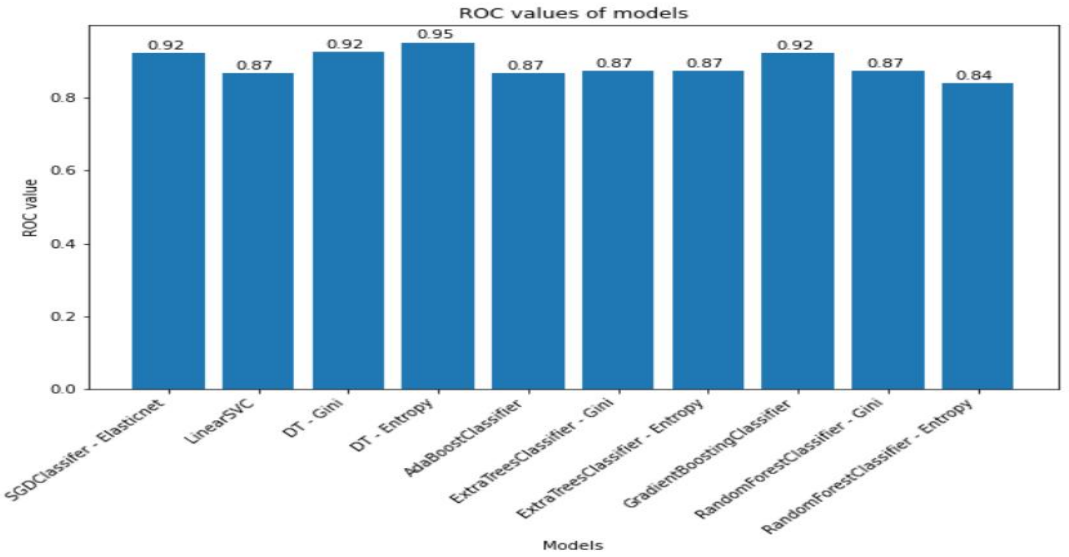
- SGDClassifier
- LinearSVC
- DT (Decision Tree) - Gini
- DT (Decision Tree)- Entropy
- AdaBoostClassifier
- ExtraTreesClassifier - Gini
- ExtraTreesClassifier - Entropy
- GradientBoostingClassifier
- RandomForestClassifier - Gini
- RandomForestClassifier - Entropy
- Linear Regression

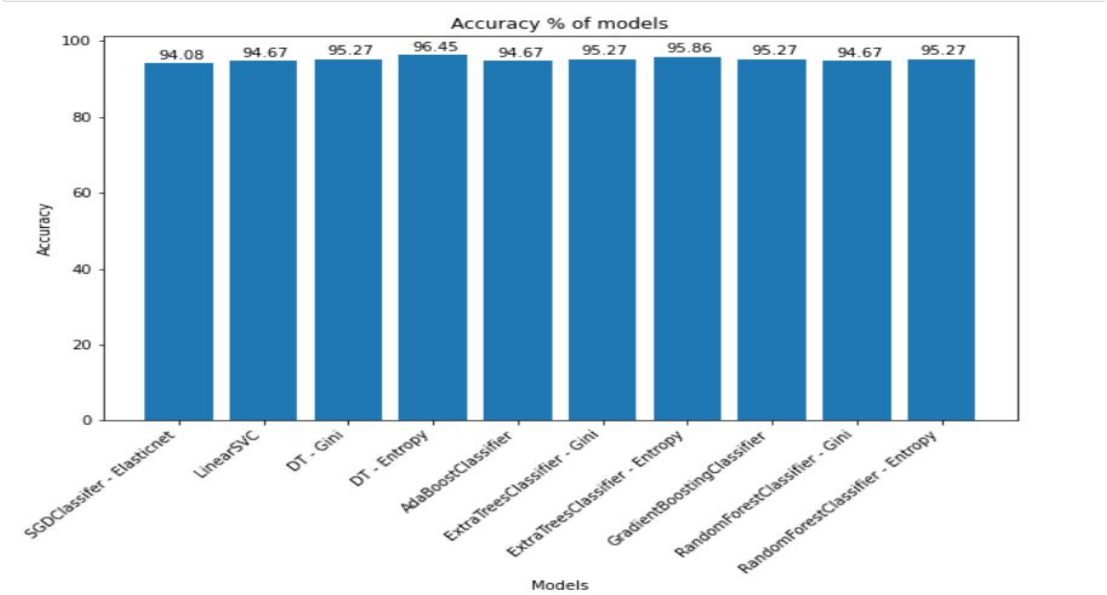


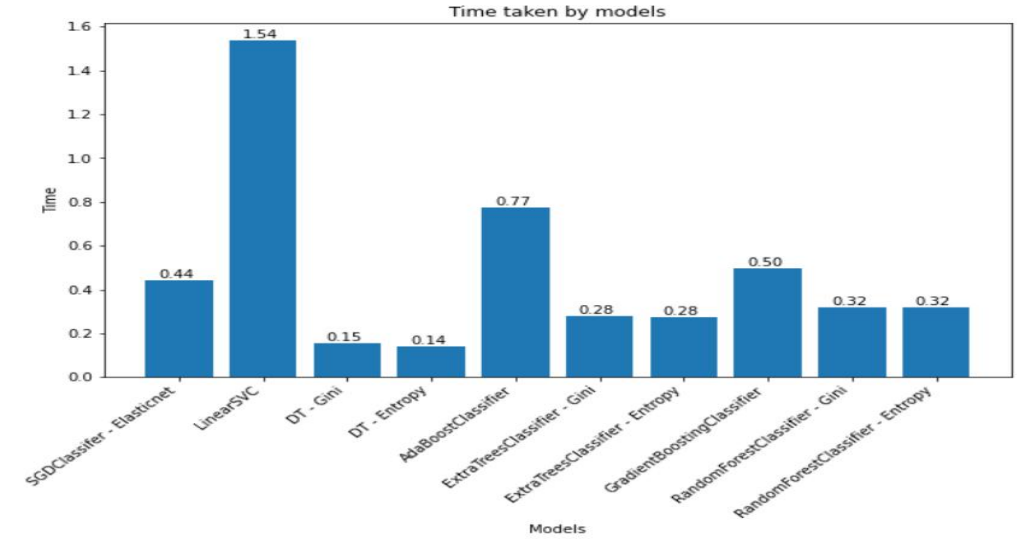
Building various Machine Learning Models and performance analysis

```
from sklearn.model_selection import train_test_split
from sklearn import metrics
from sklearn.linear_model import SGDClassifier
from sklearn import tree
from sklearn.ensemble import AdaBoostClassifier
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.ensemble import ExtraTreesClassifier
from sklearn.svm import LinearSVC
from sklearn.ensemble import RandomForestClassifier
import time

X = gwas_data.reindex(columns=list(gwas_data.columns)[1:])
y = metadata_df.azm_sr

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=0)

model_result = {}

def fit_model(classifier, X_train, X_test, y_train, y_test):
  result = {}
  start = time.process_time()
  model = classifier.fit(X_train, y_train)
  y_pred = model.predict(X_test)
  y_pred[y_pred<0.5] = 0
  y_pred[y_pred>=0.5] = 1

  result['time'] = time.process_time() - start
  result['accuracy'] = metrics.accuracy_score(y_test,y_pred)*100
  result['roc_auc_score'] = metrics.roc_auc_score(y_test, y_pred)
  result['model'] = model

  return result

sgdc_enet = SGDClassifier(loss="log", penalty="elasticnet", l1_ratio=0.1)
model_result['SGDClassifer - Elasticnet']  = fit_model(sgdc_enet, X_train, X_test, y_train, y_test)

lsvc = LinearSVC(max_iter=7000)
model_result['LinearSVC'] = fit_model(lsvc. X_train. X_test. y_train. y_test)
dec_tree_gini = tree.DecisionTreeClassifier(criterion='gini')
model_result['DT - Gini'] = fit_model(dec_tree_gini, X_train, X_test, y_train, y_test)

dec_tree_entropy = tree.DecisionTreeClassifier(criterion='entropy')
model_result['DT - Entropy'] = fit_model(dec_tree_entropy, X_train, X_test, y_train, y_test)

abc = AdaBoostClassifier()
model_result['AdaBoostClassifier'] = fit_model(abc, X_train, X_test, y_train, y_test)

etc_gini = ExtraTreesClassifier(criterion='gini')
model_result['ExtraTreesClassifier - Gini'] = fit_model(etc_gini, X_train, X_test, y_train, y_test)

etc_entropy = ExtraTreesClassifier(criterion='entropy')
model_result['ExtraTreesClassifier - Entropy'] = fit_model(etc_gini, X_train, X_test, y_train, y_test)

gbc = GradientBoostingClassifier()
model_result['GradientBoostingClassifier'] = fit_model(gbc, X_train, X_test, y_train, y_test)

rfc_gini = RandomForestClassifier(criterion='gini')
model_result['RandomForestClassifier - Gini'] = fit_model(rfc_gini, X_train, X_test, y_train, y_test)

rfc_gini = RandomForestClassifier(criterion='entropy')
model_result['RandomForestClassifier - Entropy'] = fit_model(rfc_gini, X_train, X_test, y_train, y_test)

ml_model_df = pd.DataFrame.from_dict(model_result)
ml_model_df = ml_model_df.drop(index='model')
print(tabulate(ml_model_df, headers = ml_model_df.columns, tablefmt='grid'))
```

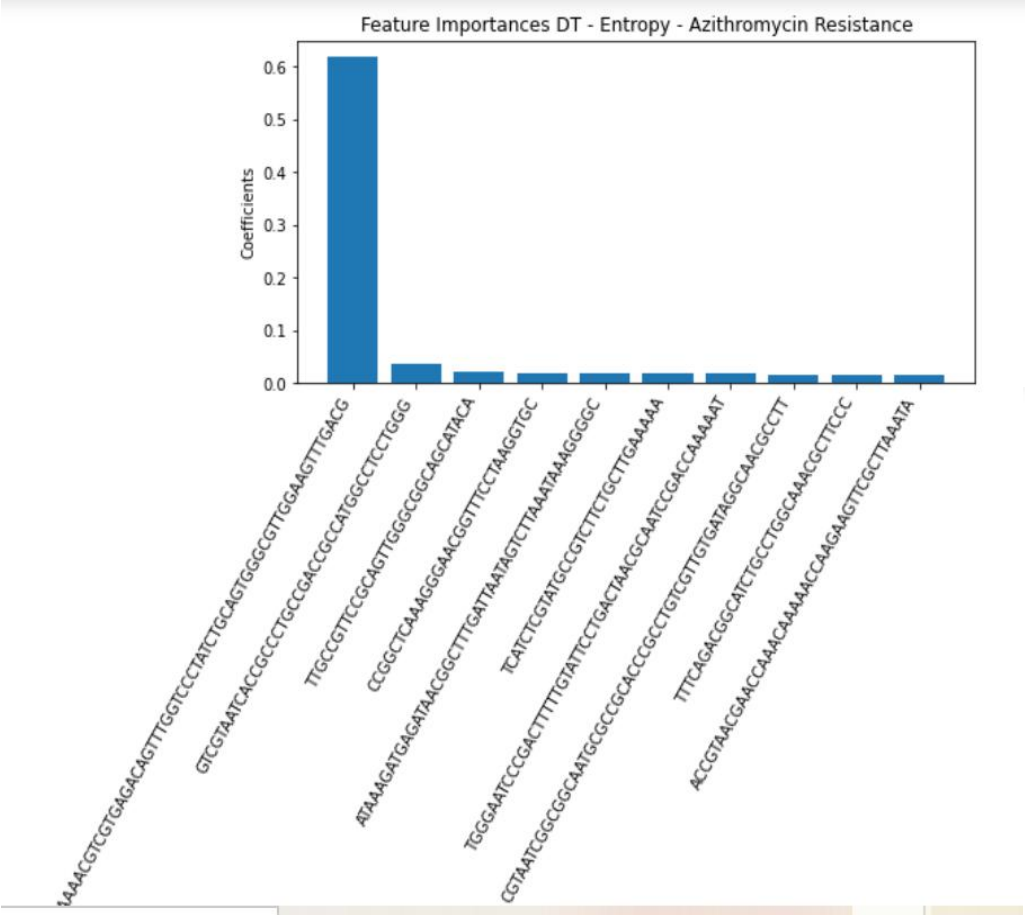
Building various Machine Learning Models and performance analysis

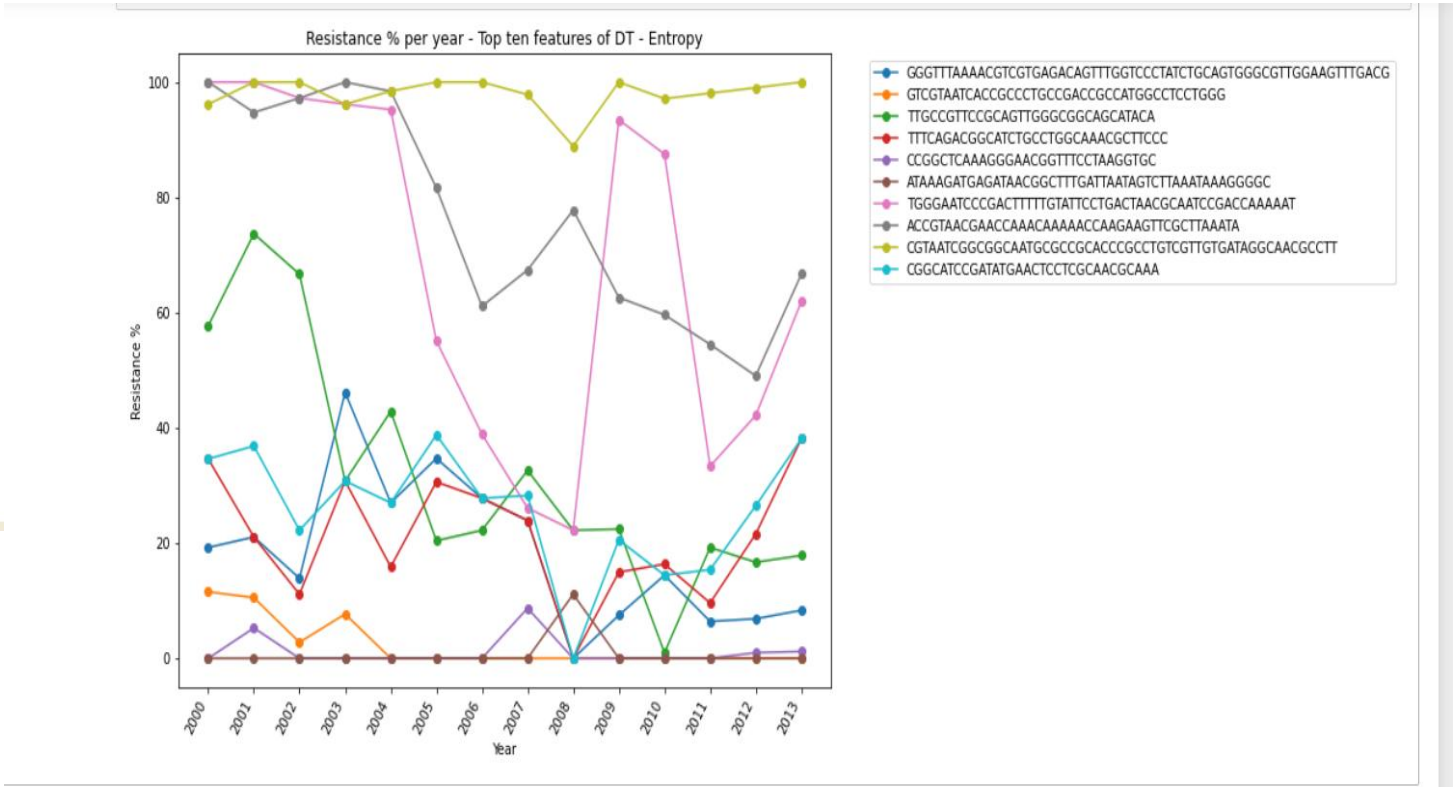






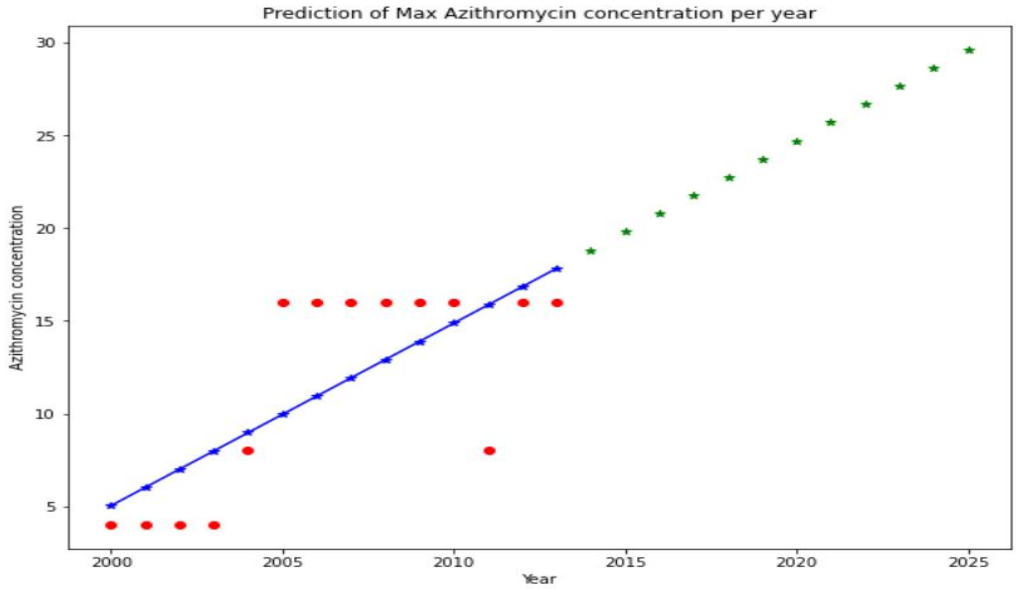
Feature importance of the best suited models for this data set

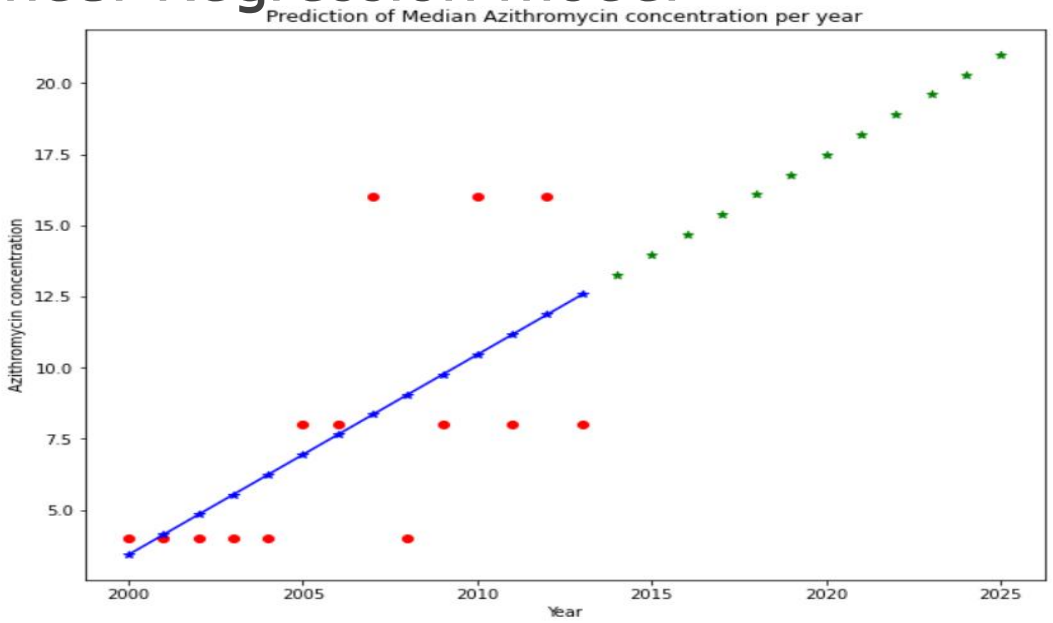


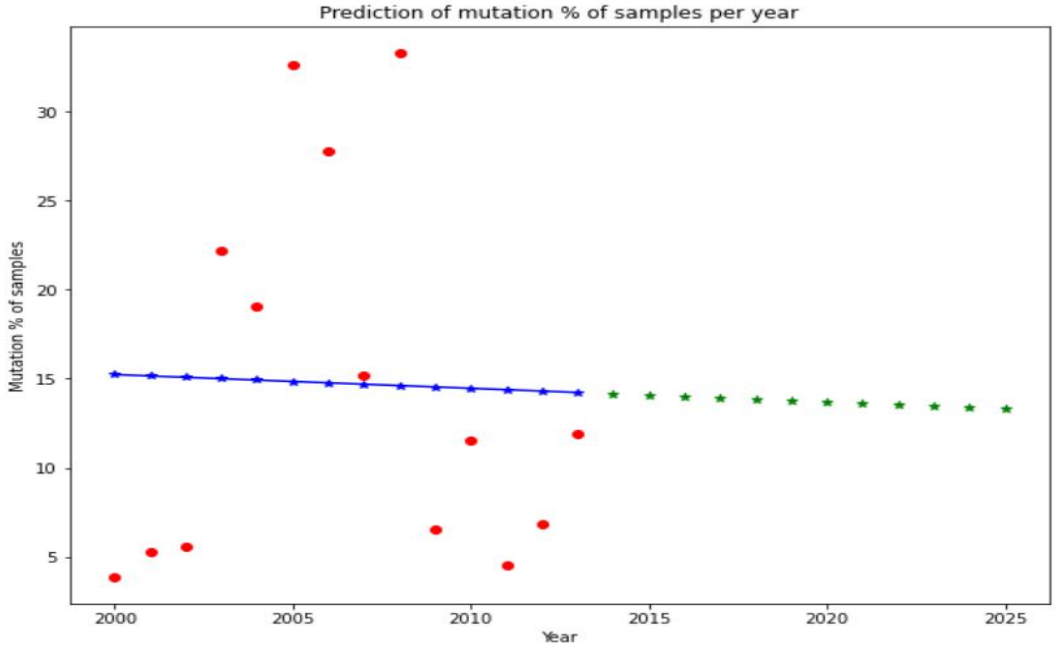


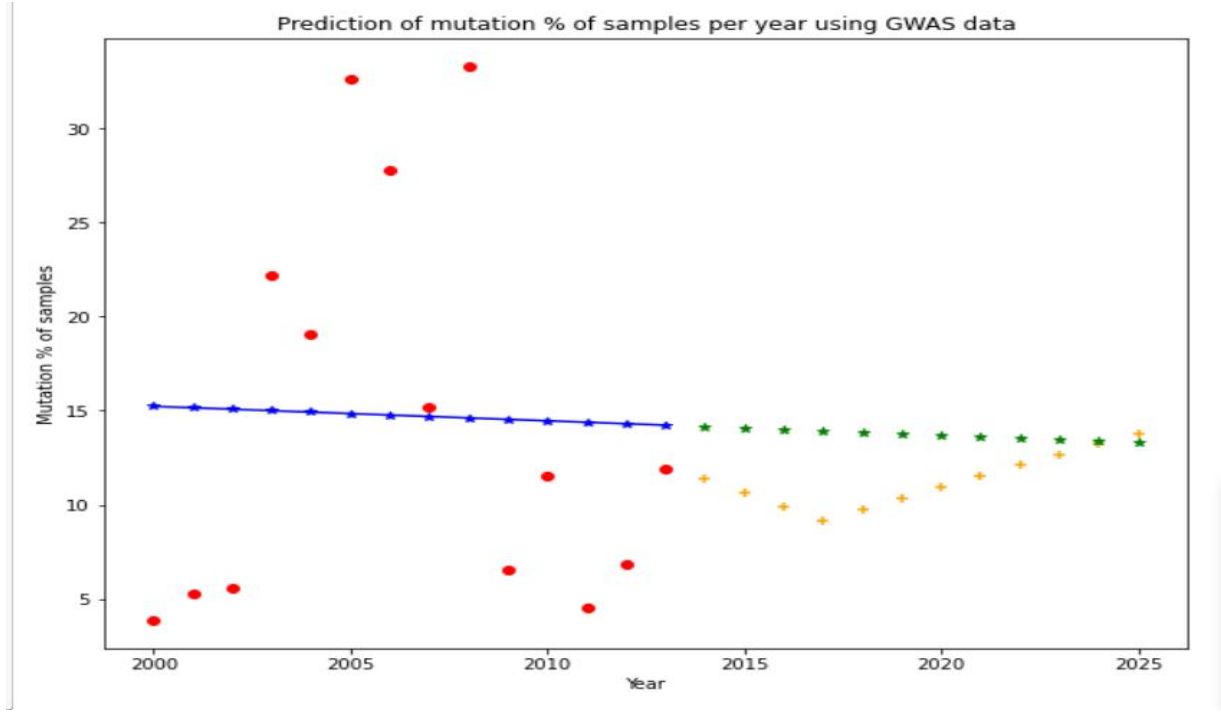
S.NO	Bacterial DNA	Gene Name
1.	CATCACCTTAGGGAATCGTTCCCTTTGGGCC	**NGMG_00872**
2.	TGGGAATCCCGACTTTTTGTATTCCTGACTAACGCAATCCGACCAAAAAT	**NGMG_02304**
3.	GTGTTACGCAATATATAAGGGGTTGCCGTTCC	**N776_04030**
4.	ACCGTAACGAACCAAACAAAAACCAAGAAGTTCGCTTAAATA	**NEIPOLOT_01195**
5.	CCGTTGCCGTTGCCGTCGCCGTCGCCGCTGCCG	**BHV70_00400**
6.	CTTGGATATGTCCAATCCTACAGTGTTACGCA	**N776_04030**
7.	AAGTCGGGAAATGCCCTTATCCGGTATGCGACCA	**N776_04030**
8.	GGAAGGCGTTCCCCGGAGCACCCAGGAGGCCATGGC	**A2X74_05240**
9.	ATGCGCGTCGCCTACGGACACGTCAGACACG	**N776_05720**
10.	GTTGAAAAAATCTTTAGCTACGTCAACGCGGGTAATTTTT	**EGK74_13440**

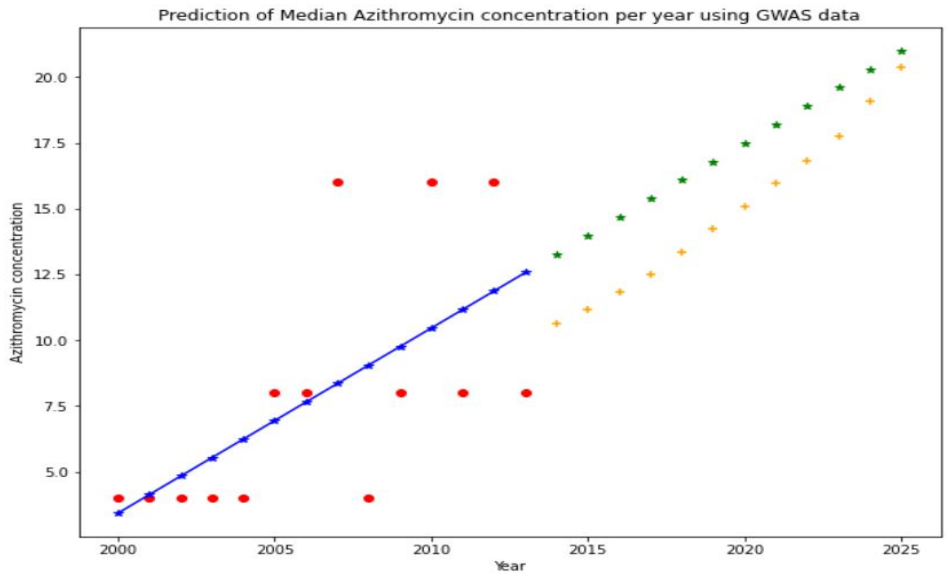
Predicting trend upto year 2025 through Linear Regression model

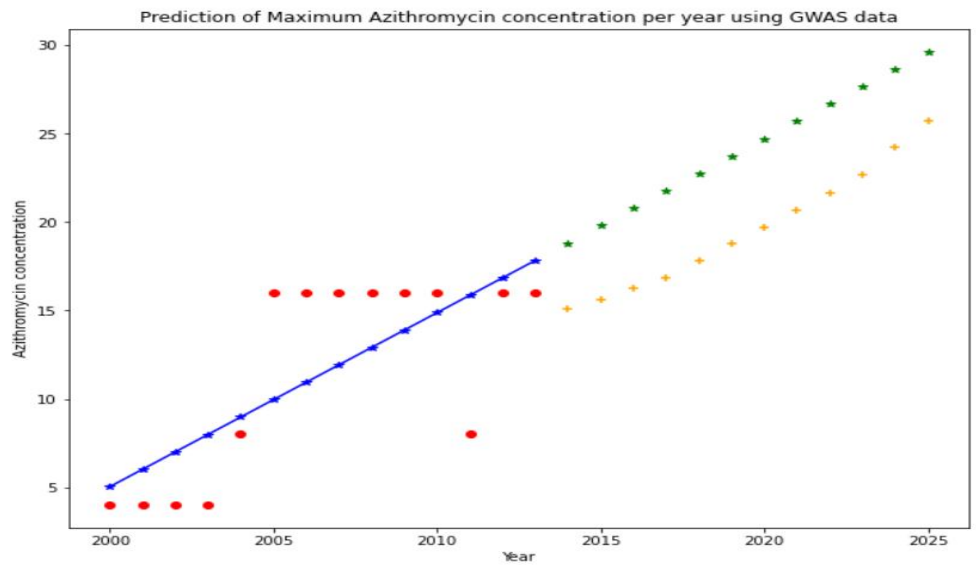







Prediction of mutation % of samples per year using GWAS data
Mutation % of samples
Year
5
10
15
20
25
30
2000
2005
2010
2015
2020
2025


Prediction of Median Azithromycin concentration per year using GWAS data
Azithromycin concentration
Year
5.0
7.5
10.0
12.5
15.0
17.5
20.0
2000
2005
2010
2015
2020
2025


Prediction of Maximum Azithromycin concentration per year using GWAS data
Azithromycin concentration
Year
5
10
15
20
25
30
2000
2005
2010
2015
2020
2025

Statistical Testing
Mancova

MANCOVA

In [153]:
```
import pandas as pd
import statsmodels.api as sm
from statsmodels.formula.api import ols
from statsmodels.stats.multicomp import pairwise_tukeyhsd
from statsmodels.sandbox.stats.multicomp import MultiComparison

reg = ols(' azm_sr ~ CATCACCTTAGGGAATCGTTCCCTTTGGGCC + GTGTTACGCAATATATAAGGGGTTGCCGTTCC + \
                                GTTGAAAAAATCTTTAGCTACGTCAACGCGGGTAATTTTT +CTTGGATATGTCCAATCCTACAGTGTTACGCA + \
                                   CCGTTGCCGTTGCCGTCGCCGTCGCCGCTGCCG +AAGTCGGGAAATGCCCTTATCCGGTATGCGACCA + \
                                     ACCGTAACGAACCAAACAAAAACCAAGAAGTTCGCTTAAATA + GGAAGGCGTTCCCCGGAGCACCCAGGAGGCCATGGC + \
                                       ATGCGCGTCGCCTACGGACACGTCAGACACG + \
                                               AAAAACCAAGAAGTTCGCTTAAATAATATAG ', data = meta_gwas_data).fit()
```

```
mc = pairwise_tukeyhsd(meta_gwas_data['azm_sr'],meta_gwas_data['CATCACCTTAGGGAATCGTTCCCTTTGGGCC'], alpha=0.05)
print(mc)
```

```
Multiple Comparison of Means - Tukey HSD, FWER=0.05
=================================================
group1 group2 meandiff p-adj  lower   upper  reject
-------------------------------------------------
     0      1  -0.1451 0.001 -0.1874 -0.1029   True
-------------------------------------------------
```

```
mc = pairwise_tukeyhsd(meta_gwas_data['azm_sr'], meta_gwas_data['GTTGAAAAAATCTTTAGCTACGTCAACGCGGGTAATTTTT'], alpha=0.01)
print(mc)
```

```
Multiple Comparison of Means - Tukey HSD, FWER=0.01
=================================================
group1 group2 meandiff p-adj  lower   upper  reject
-------------------------------------------------
     0      1  -0.1242 0.001 -0.1957 -0.0526   True
-------------------------------------------------
```

In [136]:
```
mc = pairwise_tukeyhsd(meta_gwas_data['azm_sr'],meta_gwas_data['AAGTCGGGAAATGCCCTTATCCGGTATGCGACCA'], alpha=0.05)
print(mc)
```

```
Multiple Comparison of Means - Tukey HSD, FWER=0.05
=================================================
group1 group2 meandiff p-adj lower   upper  reject
-------------------------------------------------
     0      1  -0.1094 0.001 -0.156 -0.0629   True
-------------------------------------------------
```

MANCOVA Interpretation

There is a significant relationship between the factors tested and Azithromycin Resistance.

Result Summary

1. The NGMG_00872, EGK74_13440, N776_04030, A2X74_05240, BHV70_00400 genes displayed the high resistance to Azithromycin among all the gene sequences.

2. The Minimum inhibitory concentration of Azithromycin is increasing year by year till 2025.

3. The Azithromycin resistance is decreasing year by year till 2025.

4. The Null Hypothesis is rejected.

5. Since the mutations are decreasing and the azithromycin levels are increasing by the year 2025, we are predicting that the Azithromycin consumption will increase.

Appendix:

DATA:

https://drive.google.com/drive/folders/1TF7VQmWLOmaxYpVefVCZ19furp76sosB?usp=sharing

CODE (Jupyter notebook):

Informatics Project - Group 1.ipynb file