{

"cells": [

{

"cell\_type": "code",

"execution\_count": 1,

"id": "7ca2bebc",

"metadata": {},

"outputs": [],

"source": [

"#Import pandas. \n",

"import pandas as pd"

]

},

{

"cell\_type": "code",

"execution\_count": 2,

"id": "7bc4d4a4",

"metadata": {},

"outputs": [],

"source": [

"#Import numpy. \n",

"import numpy as np"

]

},

{

"cell\_type": "code",

"execution\_count": 3,

"id": "c7dd1e8c",

"metadata": {},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

" sub\_icb\_location\_code sub\_icb\_location\_ons\_code \\\n",

"0 00L E38000130 \n",

"1 00L E38000130 \n",

"2 00L E38000130 \n",

"3 00L E38000130 \n",

"4 00L E38000130 \n",

"... ... ... \n",

"137788 X2C4Y E38000254 \n",

"137789 X2C4Y E38000254 \n",

"137790 X2C4Y E38000254 \n",

"137791 X2C4Y E38000254 \n",

"137792 X2C4Y E38000254 \n",

"\n",

" sub\_icb\_location\_name icb\_ons\_code \\\n",

"0 NHS North East and North Cumbria ICB - 00L E54000050 \n",

"1 NHS North East and North Cumbria ICB - 00L E54000050 \n",

"2 NHS North East and North Cumbria ICB - 00L E54000050 \n",

"3 NHS North East and North Cumbria ICB - 00L E54000050 \n",

"4 NHS North East and North Cumbria ICB - 00L E54000050 \n",

"... ... ... \n",

"137788 NHS West Yorkshire ICB - X2C4Y E54000054 \n",

"137789 NHS West Yorkshire ICB - X2C4Y E54000054 \n",

"137790 NHS West Yorkshire ICB - X2C4Y E54000054 \n",

"137791 NHS West Yorkshire ICB - X2C4Y E54000054 \n",

"137792 NHS West Yorkshire ICB - X2C4Y E54000054 \n",

"\n",

" region\_ons\_code appointment\_date actual\_duration \\\n",

"0 E40000012 01-Dec-21 31-60 Minutes \n",

"1 E40000012 01-Dec-21 21-30 Minutes \n",

"2 E40000012 01-Dec-21 6-10 Minutes \n",

"3 E40000012 01-Dec-21 Unknown / Data Quality \n",

"4 E40000012 01-Dec-21 16-20 Minutes \n",

"... ... ... ... \n",

"137788 E40000012 30-Jun-22 31-60 Minutes \n",

"137789 E40000012 30-Jun-22 21-30 Minutes \n",

"137790 E40000012 30-Jun-22 16-20 Minutes \n",

"137791 E40000012 30-Jun-22 11-15 Minutes \n",

"137792 E40000012 30-Jun-22 1-5 Minutes \n",

"\n",

" count\_of\_appointments \n",

"0 364 \n",

"1 619 \n",

"2 1698 \n",

"3 1277 \n",

"4 730 \n",

"... ... \n",

"137788 430 \n",

"137789 751 \n",

"137790 921 \n",

"137791 1439 \n",

"137792 1739 \n",

"\n",

"[137793 rows x 8 columns]\n"

]

}

],

"source": [

"# Read the CSV file.\n",

"# Assign a variable.\n",

"# Use the pd.read\_csv() function.\n",

"# Specify the name of the CSV file.\n",

"ad = pd.read\_csv('actual\_duration.csv')\n",

" \n",

"# Print the DataFrame for actual duration.\n",

"print (ad) "

]

},

{

"cell\_type": "code",

"execution\_count": 4,

"id": "cd1ef95f",

"metadata": {},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

" icb\_ons\_code appointment\_month appointment\_status hcp\_type \\\n",

"0 E54000034 2020-01 Attended GP \n",

"1 E54000034 2020-01 Attended GP \n",

"2 E54000034 2020-01 Attended GP \n",

"3 E54000034 2020-01 Attended GP \n",

"4 E54000034 2020-01 Attended GP \n",

"... ... ... ... ... \n",

"596816 E54000050 2022-06 Unknown Unknown \n",

"596817 E54000050 2022-06 Unknown Unknown \n",

"596818 E54000050 2022-06 Unknown Unknown \n",

"596819 E54000050 2022-06 Unknown Unknown \n",

"596820 E54000050 2022-06 Unknown Unknown \n",

"\n",

" appointment\_mode time\_between\_book\_and\_appointment \\\n",

"0 Face-to-Face 1 Day \n",

"1 Face-to-Face 15 to 21 Days \n",

"2 Face-to-Face 2 to 7 Days \n",

"3 Face-to-Face 22 to 28 Days \n",

"4 Face-to-Face 8 to 14 Days \n",

"... ... ... \n",

"596816 Unknown 2 to 7 Days \n",

"596817 Unknown 22 to 28 Days \n",

"596818 Unknown 8 to 14 Days \n",

"596819 Unknown More than 28 Days \n",

"596820 Unknown Same Day \n",

"\n",

" count\_of\_appointments \n",

"0 8107 \n",

"1 6791 \n",

"2 20686 \n",

"3 4268 \n",

"4 11971 \n",

"... ... \n",

"596816 21 \n",

"596817 8 \n",

"596818 28 \n",

"596819 17 \n",

"596820 10 \n",

"\n",

"[596821 rows x 7 columns]\n"

]

}

],

"source": [

"# Read the CSV file.\n",

"# Assign a variable.\n",

"# Use the pd.read\_csv() function.\n",

"# Specify the name of the CSV file.\n",

"ar = pd.read\_csv('appointments\_regional.csv')\n",

" \n",

"# Print the DataFrame for appointments regional.\n",

"print (ar) "

]

},

{

"cell\_type": "code",

"execution\_count": 5,

"id": "d9363a21",

"metadata": {},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

" appointment\_date icb\_ons\_code \\\n",

"0 2021-08-02 E54000050 \n",

"1 2021-08-02 E54000050 \n",

"2 2021-08-02 E54000050 \n",

"3 2021-08-02 E54000050 \n",

"4 2021-08-02 E54000050 \n",

"... ... ... \n",

"817389 2022-06-30 E54000054 \n",

"817390 2022-06-30 E54000054 \n",

"817391 2022-06-30 E54000054 \n",

"817392 2022-06-30 E54000054 \n",

"817393 2022-06-30 E54000054 \n",

"\n",

" sub\_icb\_location\_name service\_setting \\\n",

"0 NHS North East and North Cumbria ICB - 00L Primary Care Network \n",

"1 NHS North East and North Cumbria ICB - 00L Other \n",

"2 NHS North East and North Cumbria ICB - 00L General Practice \n",

"3 NHS North East and North Cumbria ICB - 00L General Practice \n",

"4 NHS North East and North Cumbria ICB - 00L General Practice \n",

"... ... ... \n",

"817389 NHS West Yorkshire ICB - X2C4Y Extended Access Provision \n",

"817390 NHS West Yorkshire ICB - X2C4Y Extended Access Provision \n",

"817391 NHS West Yorkshire ICB - X2C4Y Extended Access Provision \n",

"817392 NHS West Yorkshire ICB - X2C4Y Extended Access Provision \n",

"817393 NHS West Yorkshire ICB - X2C4Y Extended Access Provision \n",

"\n",

" context\_type national\_category \\\n",

"0 Care Related Encounter Patient contact during Care Home Round \n",

"1 Care Related Encounter Planned Clinics \n",

"2 Care Related Encounter Home Visit \n",

"3 Care Related Encounter General Consultation Acute \n",

"4 Care Related Encounter Structured Medication Review \n",

"... ... ... \n",

"817389 Care Related Encounter Unplanned Clinical Activity \n",

"817390 Care Related Encounter Planned Clinics \n",

"817391 Care Related Encounter Planned Clinical Procedure \n",

"817392 Care Related Encounter General Consultation Routine \n",

"817393 Care Related Encounter General Consultation Acute \n",

"\n",

" count\_of\_appointments appointment\_month \n",

"0 3 2021-08 \n",

"1 7 2021-08 \n",

"2 79 2021-08 \n",

"3 725 2021-08 \n",

"4 2 2021-08 \n",

"... ... ... \n",

"817389 12 2022-06 \n",

"817390 4 2022-06 \n",

"817391 92 2022-06 \n",

"817392 4 2022-06 \n",

"817393 19 2022-06 \n",

"\n",

"[817394 rows x 8 columns]\n"

]

}

],

"source": [

"# Read the CSV file.\n",

"# Assign a variable.\n",

"# Use the pd.read\_excel() function.\n",

"# Specify the name of the Excel file.\n",

"nc = pd.read\_excel('national\_categories.xlsx')\n",

" \n",

"# Print the DataFrame for national categories.\n",

"print (nc) "

]

},

{

"cell\_type": "code",

"execution\_count": 6,

"id": "28fa2960",

"metadata": {},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

"Index(['sub\_icb\_location\_code', 'sub\_icb\_location\_ons\_code',\n",

" 'sub\_icb\_location\_name', 'icb\_ons\_code', 'region\_ons\_code',\n",

" 'appointment\_date', 'actual\_duration', 'count\_of\_appointments'],\n",

" dtype='object')\n"

]

}

],

"source": [

"#Print column names for ad\n",

"print (ad.columns)"

]

},

{

"cell\_type": "code",

"execution\_count": 7,

"id": "414bbf7e",

"metadata": {},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

"Index(['icb\_ons\_code', 'appointment\_month', 'appointment\_status', 'hcp\_type',\n",

" 'appointment\_mode', 'time\_between\_book\_and\_appointment',\n",

" 'count\_of\_appointments'],\n",

" dtype='object')\n"

]

}

],

"source": [

"#Print column names for ar\n",

"print (ar.columns)"

]

},

{

"cell\_type": "code",

"execution\_count": 8,

"id": "e97b43a4",

"metadata": {},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

"Index(['appointment\_date', 'icb\_ons\_code', 'sub\_icb\_location\_name',\n",

" 'service\_setting', 'context\_type', 'national\_category',\n",

" 'count\_of\_appointments', 'appointment\_month'],\n",

" dtype='object')\n"

]

}

],

"source": [

"#Print column names for nc\n",

"print (nc.columns)"

]

},

{

"cell\_type": "code",

"execution\_count": 9,

"id": "22e5a694",

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>count\_of\_appointments</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>count</th>\n",

" <td>137793.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>mean</th>\n",

" <td>1219.080011</td>\n",

" </tr>\n",

" <tr>\n",

" <th>std</th>\n",

" <td>1546.902956</td>\n",

" </tr>\n",

" <tr>\n",

" <th>min</th>\n",

" <td>1.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>25%</th>\n",

" <td>194.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>50%</th>\n",

" <td>696.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>75%</th>\n",

" <td>1621.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>max</th>\n",

" <td>15400.000000</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>"

],

"text/plain": [

" count\_of\_appointments\n",

"count 137793.000000\n",

"mean 1219.080011\n",

"std 1546.902956\n",

"min 1.000000\n",

"25% 194.000000\n",

"50% 696.000000\n",

"75% 1621.000000\n",

"max 15400.000000"

]

},

"execution\_count": 9,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"#Descriptive statistics for ad\n",

"ad.describe()"

]

},

{

"cell\_type": "code",

"execution\_count": 10,

"id": "514d36ac",

"metadata": {},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

"<class 'pandas.core.frame.DataFrame'>\n",

"RangeIndex: 137793 entries, 0 to 137792\n",

"Data columns (total 8 columns):\n",

" # Column Non-Null Count Dtype \n",

"--- ------ -------------- ----- \n",

" 0 sub\_icb\_location\_code 137793 non-null object\n",

" 1 sub\_icb\_location\_ons\_code 137793 non-null object\n",

" 2 sub\_icb\_location\_name 137793 non-null object\n",

" 3 icb\_ons\_code 137793 non-null object\n",

" 4 region\_ons\_code 137793 non-null object\n",

" 5 appointment\_date 137793 non-null object\n",

" 6 actual\_duration 137793 non-null object\n",

" 7 count\_of\_appointments 137793 non-null int64 \n",

"dtypes: int64(1), object(7)\n",

"memory usage: 8.4+ MB\n"

]

}

],

"source": [

"#Meta data for ad\n",

"ad.info()"

]

},

{

"cell\_type": "code",

"execution\_count": 11,

"id": "2229771d",

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>count\_of\_appointments</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>count</th>\n",

" <td>596821.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>mean</th>\n",

" <td>1244.601857</td>\n",

" </tr>\n",

" <tr>\n",

" <th>std</th>\n",

" <td>5856.887042</td>\n",

" </tr>\n",

" <tr>\n",

" <th>min</th>\n",

" <td>1.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>25%</th>\n",

" <td>7.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>50%</th>\n",

" <td>47.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>75%</th>\n",

" <td>308.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>max</th>\n",

" <td>211265.000000</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>"

],

"text/plain": [

" count\_of\_appointments\n",

"count 596821.000000\n",

"mean 1244.601857\n",

"std 5856.887042\n",

"min 1.000000\n",

"25% 7.000000\n",

"50% 47.000000\n",

"75% 308.000000\n",

"max 211265.000000"

]

},

"execution\_count": 11,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"#Descriptive statistics for ar\n",

"ar.describe()"

]

},

{

"cell\_type": "code",

"execution\_count": 12,

"id": "bf7d7e04",

"metadata": {},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

"<class 'pandas.core.frame.DataFrame'>\n",

"RangeIndex: 596821 entries, 0 to 596820\n",

"Data columns (total 7 columns):\n",

" # Column Non-Null Count Dtype \n",

"--- ------ -------------- ----- \n",

" 0 icb\_ons\_code 596821 non-null object\n",

" 1 appointment\_month 596821 non-null object\n",

" 2 appointment\_status 596821 non-null object\n",

" 3 hcp\_type 596821 non-null object\n",

" 4 appointment\_mode 596821 non-null object\n",

" 5 time\_between\_book\_and\_appointment 596821 non-null object\n",

" 6 count\_of\_appointments 596821 non-null int64 \n",

"dtypes: int64(1), object(6)\n",

"memory usage: 31.9+ MB\n"

]

}

],

"source": [

"#Meta data for ar\n",

"ar.info()"

]

},

{

"cell\_type": "code",

"execution\_count": 13,

"id": "ef1052c4",

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>count\_of\_appointments</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>count</th>\n",

" <td>817394.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>mean</th>\n",

" <td>362.183684</td>\n",

" </tr>\n",

" <tr>\n",

" <th>std</th>\n",

" <td>1084.576600</td>\n",

" </tr>\n",

" <tr>\n",

" <th>min</th>\n",

" <td>1.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>25%</th>\n",

" <td>7.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>50%</th>\n",

" <td>25.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>75%</th>\n",

" <td>128.000000</td>\n",

" </tr>\n",

" <tr>\n",

" <th>max</th>\n",

" <td>16590.000000</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>"

],

"text/plain": [

" count\_of\_appointments\n",

"count 817394.000000\n",

"mean 362.183684\n",

"std 1084.576600\n",

"min 1.000000\n",

"25% 7.000000\n",

"50% 25.000000\n",

"75% 128.000000\n",

"max 16590.000000"

]

},

"execution\_count": 13,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"#Descriptive statistics for nc\n",

"nc.describe()"

]

},

{

"cell\_type": "code",

"execution\_count": 14,

"id": "7575a169",

"metadata": {},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

"<class 'pandas.core.frame.DataFrame'>\n",

"RangeIndex: 817394 entries, 0 to 817393\n",

"Data columns (total 8 columns):\n",

" # Column Non-Null Count Dtype \n",

"--- ------ -------------- ----- \n",

" 0 appointment\_date 817394 non-null datetime64[ns]\n",

" 1 icb\_ons\_code 817394 non-null object \n",

" 2 sub\_icb\_location\_name 817394 non-null object \n",

" 3 service\_setting 817394 non-null object \n",

" 4 context\_type 817394 non-null object \n",

" 5 national\_category 817394 non-null object \n",

" 6 count\_of\_appointments 817394 non-null int64 \n",

" 7 appointment\_month 817394 non-null object \n",

"dtypes: datetime64[ns](1), int64(1), object(6)\n",

"memory usage: 49.9+ MB\n"

]

}

],

"source": [

"#Meta data for nc\n",

"nc.info()"

]

},

{

"cell\_type": "code",

"execution\_count": 15,

"id": "4cc1d61b",

"metadata": {},

"outputs": [

{

"data": {

"text/plain": [

"NHS North West London ICB - W2U3Z 13007\n",

"NHS Kent and Medway ICB - 91Q 12637\n",

"NHS Devon ICB - 15N 12526\n",

"NHS Hampshire and Isle Of Wight ICB - D9Y0V 12171\n",

"NHS North East London ICB - A3A8R 11837\n",

" ... \n",

"NHS North East and North Cumbria ICB - 00N 4210\n",

"NHS Lancashire and South Cumbria ICB - 02G 4169\n",

"NHS Cheshire and Merseyside ICB - 01V 3496\n",

"NHS Cheshire and Merseyside ICB - 01T 3242\n",

"NHS Greater Manchester ICB - 00V 2170\n",

"Name: sub\_icb\_location\_name, Length: 106, dtype: int64"

]

},

"execution\_count": 15,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"#List of the different locations in the data set\n",

"nc[\"sub\_icb\_location\_name\"].value\_counts( )"

]

},

{

"cell\_type": "code",

"execution\_count": 24,

"id": "d2a6f69b",

"metadata": {},

"outputs": [

{

"data": {

"text/plain": [

"General Practice 359274\n",

"Primary Care Network 183790\n",

"Other 138789\n",

"Extended Access Provision 108122\n",

"Unmapped 27419\n",

"Name: service\_setting, dtype: int64"

]

},

"execution\_count": 24,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"#List of different service settings\n",

"nc[\"service\_setting\"].value\_counts()"

]

},

{

"cell\_type": "code",

"execution\_count": 25,

"id": "8d08677b",

"metadata": {},

"outputs": [

{

"data": {

"text/plain": [

"Care Related Encounter 700481\n",

"Inconsistent Mapping 89494\n",

"Unmapped 27419\n",

"Name: context\_type, dtype: int64"

]

},

"execution\_count": 25,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"#List of different context types\n",

"nc[\"context\_type\"].value\_counts()"

]

},

{

"cell\_type": "code",

"execution\_count": 27,

"id": "70f55c75",

"metadata": {},

"outputs": [

{

"data": {

"text/plain": [

"Inconsistent Mapping 89494\n",

"General Consultation Routine 89329\n",

"General Consultation Acute 84874\n",

"Planned Clinics 76429\n",

"Clinical Triage 74539\n",

"Planned Clinical Procedure 59631\n",

"Structured Medication Review 44467\n",

"Service provided by organisation external to the practice 43095\n",

"Home Visit 41850\n",

"Unplanned Clinical Activity 40415\n",

"Patient contact during Care Home Round 28795\n",

"Unmapped 27419\n",

"Care Home Visit 26644\n",

"Social Prescribing Service 26492\n",

"Care Home Needs Assessment & Personalised Care and Support Planning 23505\n",

"Non-contractual chargeable work 20896\n",

"Walk-in 14179\n",

"Group Consultation and Group Education 5341\n",

"Name: national\_category, dtype: int64"

]

},

"execution\_count": 27,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"#List of national categories\n",

"nc[\"national\_category\"].value\_counts()"

]

},

{

"cell\_type": "code",

"execution\_count": 28,

"id": "f57f7da6",

"metadata": {},

"outputs": [

{

"data": {

"text/plain": [

"Attended 232137\n",

"Unknown 201324\n",

"DNA 163360\n",

"Name: appointment\_status, dtype: int64"

]

},

"execution\_count": 28,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"#List of appointment statuses\n",

"ar[\"appointment\_status\"].value\_counts()"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "e8f2f84a",

"metadata": {},

"outputs": [],

"source": []

}

],

"metadata": {

"kernelspec": {

"display\_name": "Python 3 (ipykernel)",

"language": "python",

"name": "python3"

},

"language\_info": {

"codemirror\_mode": {

"name": "ipython",

"version": 3

},

"file\_extension": ".py",

"mimetype": "text/x-python",

"name": "python",

"nbconvert\_exporter": "python",

"pygments\_lexer": "ipython3",

"version": "3.9.12"

}

},

"nbformat": 4,

"nbformat\_minor": 5

}