CMS_Fraud_EDA

November 13, 2025

0.1 CMS Exclusions Dataset Exploratory Data Analysis

[48]:
```
# Import necessary libraries such as numpy and pandas
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import os
```

[47]:
```
# This function will provide basic insights into any data
def quick_summary(df):
    print("=== Missing Values ===")
    print(df.isnull().sum())
    print("\n=== Basic Info ===")
    print(df.info())
    print("\n=== Sample Rows ===")
    print(df.head())
```

0.1.1 OIG Exclusions Data

[86]:
```
# Load the OIG exclusions file into a pandas dataframe
exclusions_file = 'cms_data/leie.csv'
exclusions = pd.read_csv(exclusions_file)
exclusions = exclusions[exclusions.NPI > 0].set_index('NPI')
```

```
/tmp/ipykernel_72/1029917934.py:3: DtypeWarning: Columns (3) have mixed types.
Specify dtype option on import or set low_memory=False.
  exclusions = pd.read_csv(exclusions_file)
```

[87]:
```
quick_summary(exclusions)
```

```
=== Missing Values ===
LASTNAME       524
FIRSTNAME      525
MIDNAME       2475
BUSNAME       7773
GENERAL          0
SPECIALTY      117
UPIN          6897
```

```
DOB            524
ADDRESS          0
CITY             0
STATE            0
ZIP              0
EXCLTYPE         0
EXCLDATE         0
REINDATE         0
WAIVERDATE       0
WVRSTATE      8293
dtype: int64

=== Basic Info ===
<class 'pandas.core.frame.DataFrame'>
Index: 8297 entries, 1972902351 to 1831242650
Data columns (total 17 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   LASTNAME    7773 non-null   object
 1   FIRSTNAME   7772 non-null   object
 2   MIDNAME     5822 non-null   object
 3   BUSNAME     524 non-null    object
 4   GENERAL     8297 non-null   object
 5   SPECIALTY   8180 non-null   object
 6   UPIN        1400 non-null   object
 7   DOB         7773 non-null   float64
 8   ADDRESS     8297 non-null   object
 9   CITY        8297 non-null   object
 10  STATE       8297 non-null   object
 11  ZIP         8297 non-null   int64
 12  EXCLTYPE    8297 non-null   object
 13  EXCLDATE    8297 non-null   int64
 14  REINDATE    8297 non-null   int64
 15  WAIVERDATE  8297 non-null   int64
 16  WVRSTATE    4 non-null      object
dtypes: float64(1), int64(4), object(12)
memory usage: 1.1+ MB
None

=== Sample Rows ===
           LASTNAME FIRSTNAME MIDNAME                      BUSNAME  \
NPI
1972902351      NaN       NaN     NaN  101 FIRST CARE PHARMACY INC
1922348218      NaN       NaN     NaN   184TH STREET PHARMACY CORP
1942476080      NaN       NaN     NaN    A & Y MEDICAL SUPPLY, INC
1275600959      NaN       NaN     NaN    A CARING ALTERNATIVE, INC
1891731758      NaN       NaN     NaN    A FAIR DEAL PHARMACY, INC
```

```
                    GENERAL           SPECIALTY UPIN  DOB  \
NPI
1972902351  OTHER BUSINESS            PHARMACY  NaN  NaN
1922348218  OTHER BUSINESS            PHARMACY  NaN  NaN
1942476080     DME COMPANY       DME - GENERAL  NaN  NaN
1275600959  OTHER BUSINESS  HOME HEALTH AGENCY  NaN  NaN
1891731758  OTHER BUSINESS            PHARMACY  NaN  NaN

                                  ADDRESS          CITY STATE    ZIP  \
NPI
1972902351   C/O 609 W 191ST STREET, APT D      NEW YORK    NY  10040
1922348218                   69 E 184TH ST         BRONX    NY  10468
1942476080       6310 108TH STREET, APT 6J  FOREST HILLS    NY  11375
1275600959        1229 HURON RD E, FLR 6TH     CLEVELAND    OH  44115
1891731758  C/O  P O BOX 329014, #69709-05      BROOKLYN    NY  11232

           EXCLTYPE  EXCLDATE  REINDATE  WAIVERDATE WVRSTATE
NPI
1972902351   1128b8  20220320         0           0      NaN
1922348218   1128a1  20180419         0           0      NaN
1942476080   1128b8  20170518         0           0      NaN
1275600959   1128a1  20130320         0           0      NaN
1891731758   1128b8  20170518         0           0      NaN
```

[88]:
```
# Drops unimportant columns or columns with mostly missing values
exclusions = exclusions.drop(
    columns=['UPIN', 'MIDNAME', 'WVRSTATE', 'BUSNAME', 'REINDATE',␣
 ↪'WAIVERDATE'],
    errors='ignore'
)

exclusions.head()
```

[88]:
```
           LASTNAME FIRSTNAME         GENERAL           SPECIALTY  DOB  \
NPI
1972902351      NaN       NaN  OTHER BUSINESS            PHARMACY  NaN
1922348218      NaN       NaN  OTHER BUSINESS            PHARMACY  NaN
1942476080      NaN       NaN     DME COMPANY       DME - GENERAL  NaN
1275600959      NaN       NaN  OTHER BUSINESS  HOME HEALTH AGENCY  NaN
1891731758      NaN       NaN  OTHER BUSINESS            PHARMACY  NaN

                                  ADDRESS          CITY STATE    ZIP  \
NPI
1972902351   C/O 609 W 191ST STREET, APT D      NEW YORK    NY  10040
1922348218                   69 E 184TH ST         BRONX    NY  10468
1942476080       6310 108TH STREET, APT 6J  FOREST HILLS    NY  11375
1275600959        1229 HURON RD E, FLR 6TH     CLEVELAND    OH  44115
```

```
1891731758  C/O  P O BOX 329014, #69709-05      BROOKLYN    NY  11232

           EXCLTYPE  EXCLDATE
NPI
1972902351   1128b8  20220320
1922348218   1128a1  20180419
1942476080   1128b8  20170518
1275600959   1128a1  20130320
1891731758   1128b8  20170518
```

[89]:
```
# Drops rows with N/A values in specialty and 'dob' column as that is important␣
 ↪for insights
exclusions = exclusions.dropna(subset=['SPECIALTY'])
exclusions = exclusions.dropna(subset=['DOB'])
print(exclusions.shape)
```

```
(7723, 11)
```

[90]:
```
# Changes dates from YYYYMMDD to YYYY-MM-DD
date_cols = ['DOB', 'EXCLDATE']
for col in date_cols:
    exclusions[col] = exclusions[col].astype(str).str.replace(r'\.0$', '',␣
 ↪regex=True)
    exclusions[col] = exclusions[col].replace(['nan', 'NaN', '', '00000000'],␣
 ↪np.nan)
    exclusions[col] = pd.to_datetime(exclusions[col], format='%Y%m%d',␣
 ↪errors='coerce')

exclusions.head()
```

[90]:
```
               LASTNAME FIRSTNAME             GENERAL          SPECIALTY  \
NPI
1760461826  ABAD-SANTOS  CRISELDA  PHYSICIAN (MD, DO)         PSYCHIATRY
1477537496        ABADI  JAMSHEED  PHYSICIAN (MD, DO)  INTERNAL MEDICINE
1124292966   ABARIENTOS   CRISPIN  PHYSICIAN (MD, DO)       RHEUMATOLOGY
1376108431        ABBAS     SHAFI      BUS OWNER/EXEC  DME - PROSTHETICS
1194807255      ABBASSI     JADAN  PHYSICIAN (MD, DO)   GENERAL PRACTICE

                  DOB                  ADDRESS             CITY STATE    ZIP  \
NPI
1760461826 1963-12-20           8506 N ADIR DR        WEST HILLS    CA  91304
1477537496 1939-01-10            89 WEEKS ROAD  E WILLISTON PARK    NY  11596
1124292966 1974-09-19  P O BOX 879, #26401-014             AYER    MA   1432
1376108431 1967-06-06            P O BOX 26020          BEAUMONT    TX  26020
1194807255 1944-09-19         115 NELLIS DRIVE             WAYNE    NJ   7470

           EXCLTYPE   EXCLDATE
```

```
NPI
1760461826   1128b4 2025-01-20
1477537496   1128b4 2014-05-20
1124292966   1128a1 2020-06-18
1376108431   1128a1 2025-10-20
1194807255   1128b4 2018-06-20
```

[94]:
```python
# Creates two new columns: Age of the provider at exclusion, and the year excluded
exclusions['age_at_exclusion'] = (
    (exclusions['EXCLDATE'] - exclusions['DOB']).dt.days / 365.25
)
exclusions['year_excl'] = exclusions['EXCLDATE'].dt.year
```

[96]:
```python
excltype_map = {
    '1128a1': 'Conviction - Medicare Fraud',
    '1128a2': 'Patient Abuse/Neglect',
    '1128a3': 'Felony - Drugs',
    '1128a4': 'Felony - Healthcare Fraud',
    '1128b1': 'Misdemeanor - Fraud',
    '1128b2': 'Default on Student Loan',
    '1128b3': 'License Revocation',
    '1128b4': 'Unlawful Claims',
    '1128b5': 'Kickbacks/Bribery',
    '1128b6': 'False Claims',
    '1128b7': 'Obstruction of Audit',
    '1128b8': 'Controlled Substances Violation',
    '1128b9': 'Insurance Fraud',
    '1128b10': 'Unlawful Billing',
    '1128b11': 'Quality of Care Violation',
    '1128b12': 'Civil Monetary Penalty',
    '1128b13': 'False Statement',
    '1128b14': 'Suspension/Exclusion',
    '1128b15': 'License Suspension',
    '1128b16': 'Federal Program Violation',
}

# Apply mapping
exclusions['EXCLTYPE'] = exclusions['EXCLTYPE'].replace(excltype_map)

# Optional: Fill unmapped types with 'Other' or 'Unknown'
exclusions['EXCLTYPE'] = exclusions['EXCLTYPE'].fillna('Other')

# Check result
print(exclusions['EXCLTYPE'].value_counts())
```

```
EXCLTYPE
Conviction - Medicare Fraud    2921
```

```
Unlawful Claims                2388
Felony - Healthcare Fraud       982
Felony - Drugs                  642
Patient Abuse/Neglect           370
Suspension/Exclusion            197
Obstruction of Audit             84
Kickbacks/Bribery                51
Misdemeanor - Fraud              41
License Revocation               20
False Claims                      7
Default on Student Loan           7
1128Aa                            4
BRCH SA                           4
BRCH CIA                          3
Federal Program Violation         1
1156                              1
Name: count, dtype: int64
```

[91]:
```
quick_summary(exclusions)
```

```
=== Missing Values ===
LASTNAME     0
FIRSTNAME    1
GENERAL      0
SPECIALTY    0
DOB          0
ADDRESS      0
CITY         0
STATE        0
ZIP          0
EXCLTYPE     0
EXCLDATE     0
dtype: int64

=== Basic Info ===
<class 'pandas.core.frame.DataFrame'>
Index: 7723 entries, 1760461826 to 1831242650
Data columns (total 11 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   LASTNAME   7723 non-null   object
 1   FIRSTNAME  7722 non-null   object
 2   GENERAL    7723 non-null   object
 3   SPECIALTY  7723 non-null   object
 4   DOB        7723 non-null   datetime64[ns]
 5   ADDRESS    7723 non-null   object
 6   CITY       7723 non-null   object
 7   STATE      7723 non-null   object
```

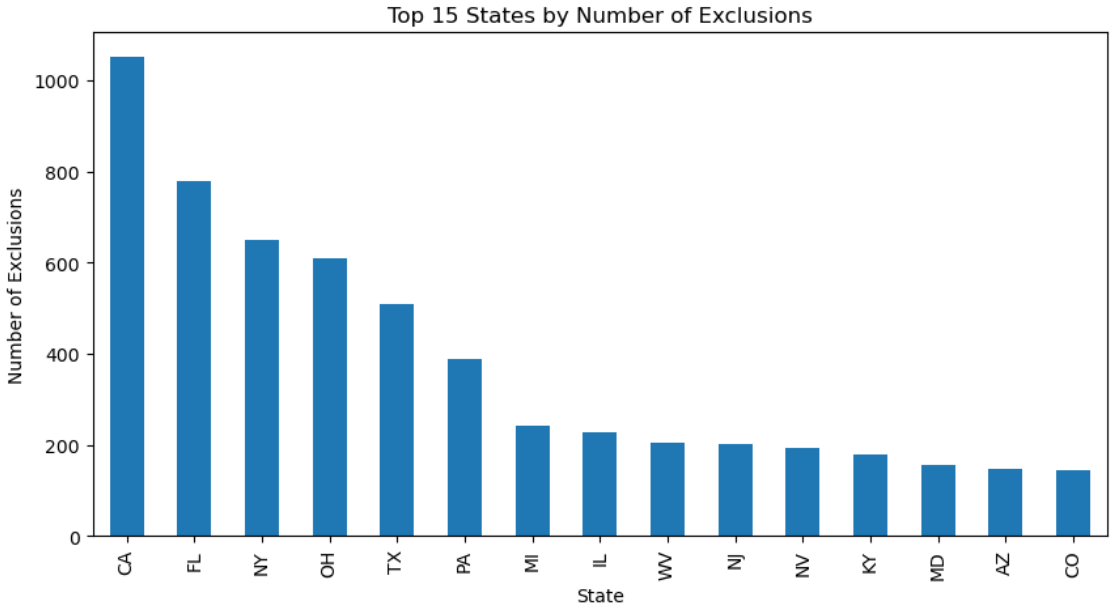
```
 8   ZIP         7723 non-null   int64
 9   EXCLTYPE    7723 non-null   object
 10  EXCLDATE    7723 non-null   datetime64[ns]
dtypes: datetime64[ns](2), int64(1), object(8)
memory usage: 724.0+ KB
None

=== Sample Rows ===
               LASTNAME FIRSTNAME             GENERAL          SPECIALTY  \
NPI
1760461826  ABAD-SANTOS  CRISELDA  PHYSICIAN (MD, DO)         PSYCHIATRY
1477537496        ABADI  JAMSHEED  PHYSICIAN (MD, DO)  INTERNAL MEDICINE
1124292966   ABARIENTOS   CRISPIN  PHYSICIAN (MD, DO)       RHEUMATOLOGY
1376108431        ABBAS     SHAFI      BUS OWNER/EXEC  DME - PROSTHETICS
1194807255      ABBASSI     JADAN  PHYSICIAN (MD, DO)   GENERAL PRACTICE

                  DOB                  ADDRESS              CITY STATE    ZIP  \
NPI
1760461826 1963-12-20           8506 N ADIR DR        WEST HILLS    CA  91304
1477537496 1939-01-10            89 WEEKS ROAD  E WILLISTON PARK    NY  11596
1124292966 1974-09-19  P O BOX 879, #26401-014              AYER    MA   1432
1376108431 1967-06-06            P O BOX 26020          BEAUMONT    TX  26020
1194807255 1944-09-19         115 NELLIS DRIVE             WAYNE    NJ   7470

           EXCLTYPE   EXCLDATE
NPI
1760461826   1128b4 2025-01-20
1477537496   1128b4 2014-05-20
1124292966   1128a1 2020-06-18
1376108431   1128a1 2025-10-20
1194807255   1128b4 2018-06-20
```

[65]:
```
exclusions['SPECIALTY'].value_counts().head(15)
```

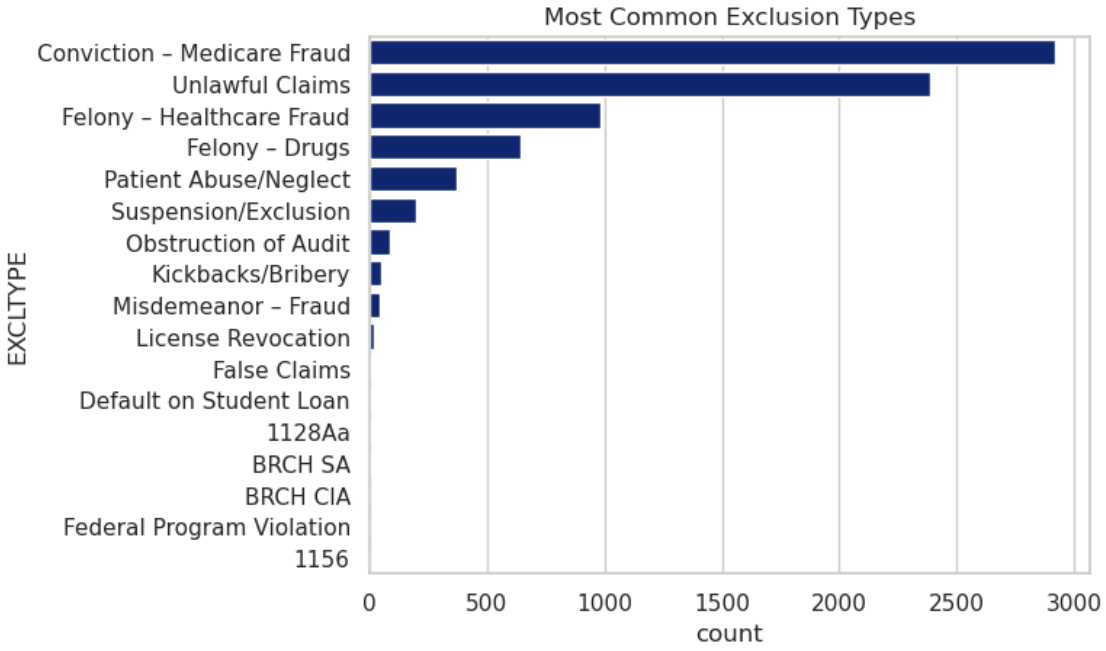
[65]:
```
SPECIALTY
NURSE/NURSES AIDE       826
GENERAL PRACTICE        775
FAMILY PRACTICE         598
INTERNAL MEDICINE       552
CHIROPRACTIC            422
COUNSELOR               397
DENTIST                 334
PHARMACIST              324
PSYCHIATRY              245
NURSE PRACTITIONER (    188
PAIN MANAGEMENT         183
SOCIAL WORKER           172
```

```
PHYSICIAN ASSISTANT     165
THERAPIST               160
PSYCHOLOGY              151
Name: count, dtype: int64
```

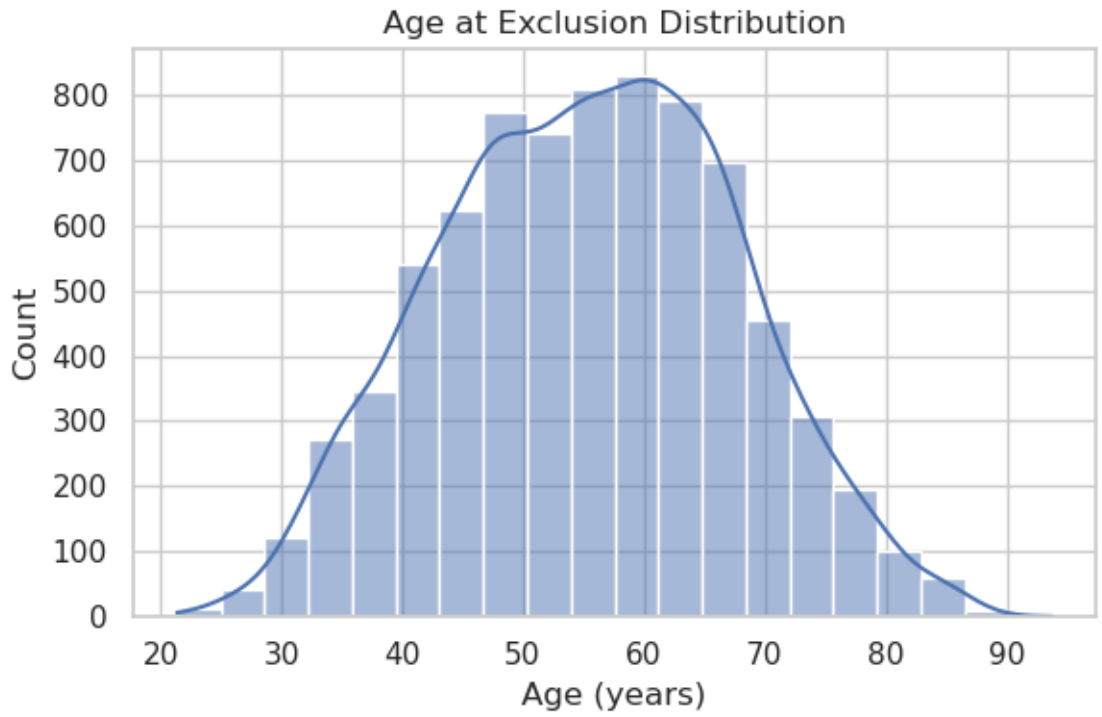
[17]:
```
plt.figure(figsize=(10,5))
exclusions['STATE'].value_counts().head(15).plot(kind='bar')
plt.title('Top 15 States by Number of Exclusions')
plt.xlabel('State')
plt.ylabel('Number of Exclusions')
plt.show()
```

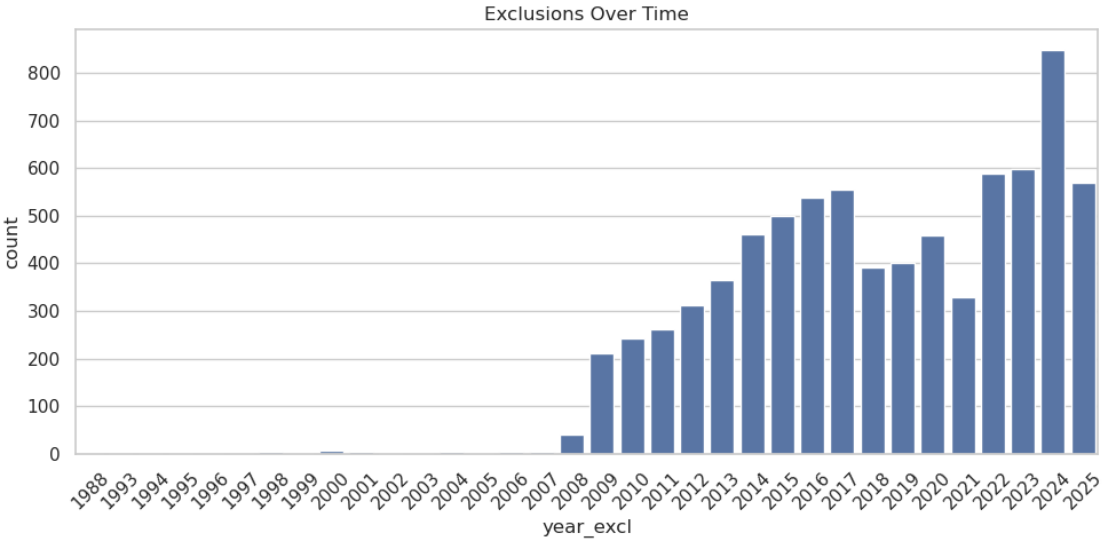


[114]:
```
sns.countplot(y='EXCLTYPE', data=exclusions, order=exclusions['EXCLTYPE'].
  ↪value_counts().index)
plt.title('Most Common Exclusion Types')
plt.show()
```



[121]:
```
sns.set(style="whitegrid")

# ---- Age Distribution ----
plt.figure(figsize=(6, 4))
sns.histplot(exclusions['age_at_exclusion'].dropna(), kde=True, bins=20)
plt.title("Age at Exclusion Distribution")
plt.xlabel("Age (years)")
plt.ylabel("Count")
plt.tight_layout()
plt.show()

# ---- Temporal Trends ----
plt.figure(figsize=(10, 5))
sns.countplot(
    data=exclusions,
    x='year_excl',
    order=sorted(exclusions['year_excl'].dropna().unique())
)
plt.title("Exclusions Over Time")
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

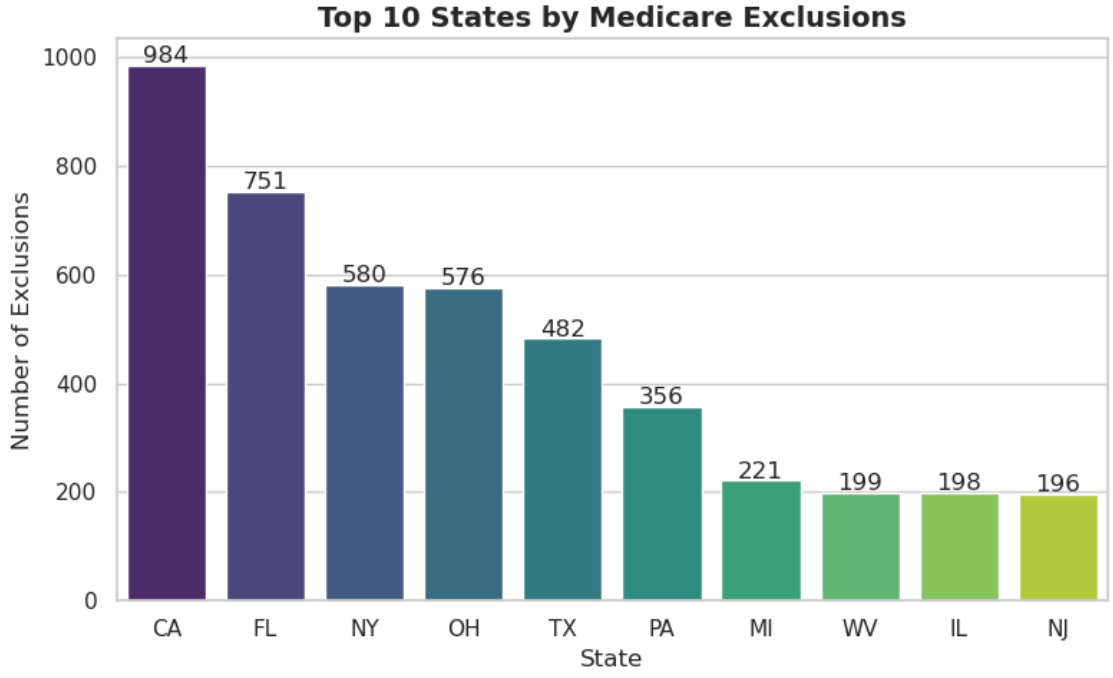




[118]:
```
top_states = (
    exclusions['STATE']
    .value_counts()
    .nlargest(10)
    .reset_index()
```

```
)

# Rename columns clearly
top_states.columns = ['STATE', 'COUNT']

# --- Visualization ---
plt.figure(figsize=(8, 5))
sns.barplot(data=top_states, x='STATE', y='COUNT', palette='viridis')

plt.title("Top 10 States by Medicare Exclusions", fontsize=14, weight='bold')
plt.xlabel("State")
plt.ylabel("Number of Exclusions")

# Add labels above bars
for i, row in top_states.iterrows():
    plt.text(i, row['COUNT'] + 5, int(row['COUNT']), ha='center')

plt.tight_layout()
plt.show()
```

```
/tmp/ipykernel_72/1706301846.py:13: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in
v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same
effect.

  sns.barplot(data=top_states, x='STATE', y='COUNT', palette='viridis')
```



0.1.2 CMS Medicare Provider Data

[100]:
```
files = os.listdir('cms_data')
files
```

[100]:
```
['MUP_PHY_R25_P07_V10_D19_Prov.csv',
 'MUP_PHY_R25_P07_V10_D22_Prov.csv',
 '.ipynb_checkpoints',
 'MUP_PHY_R25_P07_V10_D21_Prov.csv',
 'leie.csv',
 'y2022_prep.csv',
 'y2019_prep.csv',
 'MUP_PHY_R25_P05_V20_D23_Prov.csv',
 'MUP_PHY_R25_P07_V10_D20_Prov.csv',
 'y2023_prep.csv']
```

[111]:
```
fil = files[6]
data = pd.read_csv('cms_data/'+fil)
```

[113]:
```
quick_summary(data)
```

```
=== Missing Values ===
Unnamed: 0                                    0
Rndrng_NPI                                    0
Rndrng_Prvdr_Last_Org_Name                    0
Rndrng_Prvdr_First_Name                     649
Rndrng_Prvdr_MI                            4210
                                           ...
rat_Drug_Mdcr_Alowd_Amt_Drug_Mdcr_Pymt_Amt 9475
rat_Tot_Mdcr_Alowd_Amt_Med_Sbmtd_Chrg      1410
rat_Drug_Tot_Benes_Tot_Benes               1410
excluded                                      0
rand                                          0
Length: 104, dtype: int64

=== Basic Info ===
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 12470 entries, 0 to 12469
Columns: 104 entries, Unnamed: 0 to rand
dtypes: float64(70), int64(7), object(27)
memory usage: 9.9+ MB
None

=== Sample Rows ===
   Unnamed: 0  Rndrng_NPI Rndrng_Prvdr_Last_Org_Name Rndrng_Prvdr_First_Name  \
```

```
0           1  1003000134                    CIBULL                  THOMAS
1          70  1003005315                     SMITH                    ADAM
2         137  1003009861                     BANNA                 MOUSTAFA
3         151  1003010570                      CHOW                    LING
4         321  1003017443                    BLOKAR                  MIRJANA

  Rndrng_Prvdr_MI Rndrng_Prvdr_Crdntls Rndrng_Prvdr_Ent_Cd  \
0               L                 M.D.                   I
1               B                   MD                   I
2             NaN                   MD                   I
3               S                 M.D.                   I
4             NaN                 M.D.                   I

      Rndrng_Prvdr_St1   Rndrng_Prvdr_St2 Rndrng_Prvdr_City  … EXCLTYPE  \
0       2650 RIDGE AVE  EVANSTON HOSPITAL          EVANSTON  …      UNK
1      4977 SKYVIEW CT                NaN     TRAVERSE CITY  …   1128a1
2  5859 W. TALAVI BLVD          SUITE 100          GLENDALE  …      UNK
3   900 E BROADWAY AVE                NaN          BISMARCK  …      UNK
4       65 BLEECKER ST         12TH FLOOR          NEW YORK  …      UNK

     EXCLDATE  REINDATE  WAIVERDATE WVRSTATE  \
0         NaN       NaN         NaN      NaN
1  20221220.0       0.0         0.0      NaN
2         NaN       NaN         NaN      NaN
3         NaN       NaN         NaN      NaN
4         NaN       NaN         NaN      NaN

  rat_Drug_Mdcr_Alowd_Amt_Drug_Mdcr_Pymt_Amt  \
0                                        NaN
1                                        NaN
2                                   1.253709
3                                        NaN
4                                        NaN

  rat_Tot_Mdcr_Alowd_Amt_Med_Sbmtd_Chrg rat_Drug_Tot_Benes_Tot_Benes  \
0                              0.246364                    0.000000
1                                   NaN                         NaN
2                              0.498749                    0.111872
3                              0.485200                    0.000000
4                              0.527974                    0.000000

   excluded      rand
0         0  0.994100
1         1  0.253627
2         0  0.994092
3         0  0.994872
4         0  0.991242
```

[5 rows x 104 columns]

[112]:
```
plt.figure(figsize=(12, 7))

# Taking a sample of the data can make the plot less crowded and faster to␣
 ↪render
sample_df = data.sample(n=5000, random_state=42)

ax = sns.scatterplot(
    data=sample_df,
    x='Bene_Avg_Risk_Scre',
    y='Tot_Mdcr_Pymt_Amt',
    hue='Tot_Srvcs',
    size='Tot_Srvcs', # Vary point size by number of services
    sizes=(20, 200),
    palette='viridis',
    alpha=0.6
)

ax.set_yscale('log')
plt.title('Patient Risk Score vs. Total Medicare Payment', fontsize=16)
plt.xlabel('Beneficiary Average Risk Score', fontsize=12)
plt.ylabel('Total Medicare Payment Amount (Log Scale)', fontsize=12)
plt.legend(title='Total Services')
plt.tight_layout()
plt.style.use("seaborn-v0_8-dark-palette")
plt.show()
```