Vladimir_Cardenas_HAP823_Spring_2024_Final_Project_Data_Preparati

April 13, 2024

1 HAP 464 Spring 2024 Project - Response to Antidepressants

1.1 Data import from All of Us

[1]:
```
import pandas
import os

# This query represents dataset "HAP464 Demographics" for domain "person" and␣
 ↪was generated for All of Us Registered Tier Dataset v7
dataset_34828527_person_sql = """
    SELECT
        person.person_id,
        person.gender_concept_id,
        p_gender_concept.concept_name as gender,
        person.birth_datetime as date_of_birth,
        person.ethnicity_concept_id,
        p_ethnicity_concept.concept_name as ethnicity,
        person.sex_at_birth_concept_id,
        p_sex_at_birth_concept.concept_name as sex_at_birth
    FROM
        `""" + os.environ["WORKSPACE_CDR"] + """.person` person
    LEFT JOIN
        `""" + os.environ["WORKSPACE_CDR"] + """.concept` p_gender_concept
            ON person.gender_concept_id = p_gender_concept.concept_id
    LEFT JOIN
        `""" + os.environ["WORKSPACE_CDR"] + """.concept` p_ethnicity_concept
            ON person.ethnicity_concept_id = p_ethnicity_concept.concept_id
    LEFT JOIN
        `""" + os.environ["WORKSPACE_CDR"] + """.concept`␣
 ↪p_sex_at_birth_concept
            ON person.sex_at_birth_concept_id = p_sex_at_birth_concept.
 ↪concept_id
    WHERE
        person.PERSON_ID IN (
            SELECT
                distinct person_id
```

```
            FROM
                `""" + os.environ["WORKSPACE_CDR"] + """.cb_search_person`␣
↪cb_search_person
            WHERE
                cb_search_person.person_id IN (
                    SELECT
                        person_id
                    FROM
                        `""" + os.environ["WORKSPACE_CDR"] + """.person` p
                    WHERE
                        race_concept_id IN (8516)
                )
                AND cb_search_person.person_id IN (
                    SELECT
                        criteria.person_id
                    FROM
                        (SELECT
                            DISTINCT person_id,
                            entry_date,
                            concept_id
                        FROM
                            `""" + os.environ["WORKSPACE_CDR"] + """.
↪cb_search_all_events`
                        WHERE
                            (
                                concept_id IN (
                                    SELECT
                                        DISTINCT c.concept_id
                                    FROM
                                        `""" + os.environ["WORKSPACE_CDR"] +␣
↪"""".cb_criteria` c
                                    JOIN
                                        (
                                            select
                                                cast(cr.id as string) as id
                                            FROM
                                                `""" + os.
↪environ["WORKSPACE_CDR"] + """.cb_criteria` cr
                                            WHERE
                                                concept_id IN (4152280)
                                                AND full_text LIKE '%_rank1]%'
                                        ) a
                                            ON (
                                                c.path LIKE CONCAT('%.',
                                            a.id,
                                            '.%')
                                            OR c.path LIKE CONCAT('%.',
```

```
                                        a.id)
                                        OR c.path LIKE CONCAT(a.id,
                                        '.%')
                                        OR c.path = a.id)
                                    WHERE
                                        is_standard = 1
                                        AND is_selectable = 1
                                    )
                                    AND is_standard = 1
                                )
                            ) criteria
                        )
                        AND cb_search_person.person_id NOT IN (
                            SELECT
                                criteria.person_id
                            FROM
                                (SELECT
                                    DISTINCT person_id,
                                    entry_date,
                                    concept_id
                                FROM
                                    `""" + os.environ["WORKSPACE_CDR"] + """.
↪cb_search_all_events`
                                WHERE
                                    (
                                        concept_id IN (
                                            SELECT
                                                DISTINCT c.concept_id
                                            FROM
                                                `""" + os.
↪environ["WORKSPACE_CDR"] + """.cb_criteria` c
                                            JOIN
                                                (
                                                    select
                                                        cast(cr.id as string)␣
↪as id
                                                    FROM
                                                        `""" + os.
↪environ["WORKSPACE_CDR"] + """.cb_criteria` cr
                                                    WHERE
                                                        concept_id IN (436665)
                                                        AND full_text LIKE␣
↪'%_rank1]%'
                                                ) a
                                                ON (
                                                    c.path LIKE CONCAT('%.',
                                                    a.id,
```

```
                                                    '.%')
                                                    OR c.path LIKE CONCAT('%.',
                                                    a.id)
                                                    OR c.path LIKE CONCAT(a.id,
                                                    '.%')
                                                    OR c.path = a.id)
                                                WHERE
                                                    is_standard = 1
                                                    AND is_selectable = 1
                                                )
                                                AND is_standard = 1
                                            )
                                        ) criteria
                                    ) )"""

dataset_34828527_person_df = pandas.read_gbq(
    dataset_34828527_person_sql,
    dialect="standard",
    use_bqstorage_api=("BIGQUERY_STORAGE_API_ENABLED" in os.environ),
    progress_bar_type="tqdm_notebook")

dataset_34828527_person_df.head(5)
```

```
Downloading:   0%|          | 0/9408 [00:00<?, ?rows/s]
```

[1]:

	person_id	gender_concept_id	gender	date_of_birth	ethnicity_concept_id	ethnicity	sex_at_birth_concept_id	sex_at_birth
0	1096517	45878463	Female	1950-01-19 00:00:00+00:00	38003563	Hispanic or Latino		
1	2606002	2000000002	Not man only, not woman only, prefer not to an…	1955-09-13 00:00:00+00:00	38003564	Not Hispanic or Latino		
2	1242988	2000000002	Not man only, not woman only, prefer not to an…	1955-10-17 00:00:00+00:00	38003564	Not Hispanic or Latino		
3	2873521	45878463	Female	1955-09-19 00:00:00+00:00	38003564	Not Hispanic or Latino		
4	1175585	45878463	Female	1957-08-28 00:00:00+00:00	38003564	Not Hispanic or Latino		

```
0                       0  No matching concept
1                       0  No matching concept
2                       0  No matching concept
3                       0  No matching concept
4                       0  No matching concept
```

[2]:
```
import pandas
import os

# This query represents dataset "HAP464 Survival" for domain "observation" and␣
 ↪was generated for All of Us Registered Tier Dataset v7
dataset_19455778_observation_sql = """
    SELECT
        observation.person_id,
        observation.observation_datetime
    FROM
        ( SELECT
            *
        FROM
            `""" + os.environ["WORKSPACE_CDR"] + """.observation` observation
        WHERE
            (
                observation_concept_id IN (
                    4051267, 4061268, 4062254, 4083743, 4132309, 4177807,␣
 ↪4195755, 4302017, 4306655, 441139, 442605, 443882
                )
            )
            AND (
                observation.PERSON_ID IN (
                    SELECT
                        distinct person_id
                    FROM
                        `""" + os.environ["WORKSPACE_CDR"] + """.
 ↪cb_search_person` cb_search_person
                    WHERE
                        cb_search_person.person_id IN (
                            SELECT
                                person_id
                            FROM
                                `""" + os.environ["WORKSPACE_CDR"] + """.
 ↪person` p
                            WHERE
                                race_concept_id IN (8516)
                        )
                        AND cb_search_person.person_id IN (
                            SELECT
                                criteria.person_id
```

```
                            FROM
                                (SELECT
                                    DISTINCT person_id,
                                    entry_date,
                                    concept_id
                                FROM
                                    `""" + os.environ["WORKSPACE_CDR"] + """.
↪cb_search_all_events`
                                WHERE
                                    (
                                        concept_id IN (
                                            SELECT
                                                DISTINCT c.concept_id
                                            FROM
                                                `""" + os.
↪environ["WORKSPACE_CDR"] + """.cb_criteria` c
                                            JOIN
                                                (
                                                    select
                                                        cast(cr.id as string)␣
↪as id
                                                    FROM
                                                        `""" + os.
↪environ["WORKSPACE_CDR"] + """.cb_criteria` cr
                                                    WHERE
                                                        concept_id IN (4152280)
                                                        AND full_text LIKE␣
↪'%_rank1]%'
                                                ) a
                                                    ON (
                                                        c.path LIKE CONCAT('%.',
                                                    a.id,
                                                    '.%')
                                                    OR c.path LIKE CONCAT('%.',
                                                    a.id)
                                                    OR c.path LIKE CONCAT(a.id,
                                                    '.%')
                                                    OR c.path = a.id)
                                            WHERE
                                                is_standard = 1
                                                AND is_selectable = 1
                                            )
                                            AND is_standard = 1
                                        )
                                    ) criteria
                                )
                                AND cb_search_person.person_id NOT IN (
```

```
                                    SELECT
                                        criteria.person_id
                                    FROM
                                        (SELECT
                                            DISTINCT person_id,
                                            entry_date,
                                            concept_id
                                        FROM
                                            `""" + os.environ["WORKSPACE_CDR"]␣
↪+ """.cb_search_all_events`
                                        WHERE
                                            (
                                                concept_id IN (
                                                    SELECT
                                                        DISTINCT c.concept_id
                                                    FROM
                                                        `""" + os.
↪environ["WORKSPACE_CDR"] + """.cb_criteria` c
                                                    JOIN
                                                        (
                                                            select
                                                                cast(cr.id as␣
↪string) as id
                                                            FROM
                                                                `""" + os.
↪environ["WORKSPACE_CDR"] + """.cb_criteria` cr
                                                            WHERE
                                                                concept_id IN␣
↪(436665)
                                                                AND full_text␣
↪LIKE '%_rank1]%'
                                                        ) a
                                                            ON (
                                                                c.path LIKE␣
↪CONCAT('%.',
                                                            a.id,
                                                            '.%')
                                                            OR c.path LIKE␣
↪CONCAT('%.',
                                                            a.id)
                                                            OR c.path LIKE␣
↪CONCAT(a.id,
                                                            '.%')
                                                            OR c.path = a.id)
                                                    WHERE
                                                        is_standard = 1
```

```
                                                                    AND is_selectable =␣
 ↪1
                                                                )
                                                                AND is_standard = 1
                                                            )
                                                        ) criteria
                                                    ) ))
                                            ) observation"""

dataset_19455778_observation_df = pandas.read_gbq(
    dataset_19455778_observation_sql,
    dialect="standard",
    use_bqstorage_api=("BIGQUERY_STORAGE_API_ENABLED" in os.environ),
    progress_bar_type="tqdm_notebook")

dataset_19455778_observation_df.head(5)
```

```
Downloading:   0%|          | 0/6 [00:00<?, ?rows/s]
```

[2]:

	person_id	observation_datetime
0	3203537	2020-02-05 00:23:00+00:00
1	3008288	2020-11-27 06:00:00+00:00
2	1067278	2013-10-30 20:20:00+00:00
3	1168453	2018-12-14 01:23:00+00:00
4	1935512	2013-10-01 01:07:00+00:00

[3]:
```
import pandas
import os

# This query represents dataset "HAP464 Diseases" for domain "condition" and␣
 ↪was generated for All of Us Registered Tier Dataset v7
dataset_68803172_condition_sql = """
    SELECT
        c_occurrence.person_id,
        c_standard_concept.concept_name as standard_concept_name,
        c_standard_concept.concept_code as standard_concept_code,
        c_occurrence.condition_start_datetime,
        c_occurrence.condition_end_datetime
    FROM
        ( SELECT
            *
        FROM
            `""" + os.environ["WORKSPACE_CDR"] + """.condition_occurrence`␣
 ↪c_occurrence
        WHERE
            (
                condition_concept_id IN  (
```

```
                    SELECT
                        DISTINCT c.concept_id
                    FROM
                        `""" + os.environ["WORKSPACE_CDR"] + """.cb_criteria` c
                    JOIN
                        (
                            select
                                cast(cr.id as string) as id
                            FROM
                                `""" + os.environ["WORKSPACE_CDR"] + """.
↪cb_criteria` cr
                            WHERE
                                concept_id IN (
                                    4274025
                                )
                                AND full_text LIKE '%_rank1]%'
                        ) a
                            ON (
                                c.path LIKE CONCAT('%.',
                            a.id,
                            '.%')
                            OR c.path LIKE CONCAT('%.',
                            a.id)
                            OR c.path LIKE CONCAT(a.id,
                            '.%')
                            OR c.path = a.id)
                        WHERE
                            is_standard = 1
                            AND is_selectable = 1
                        )
                )
                AND (
                    c_occurrence.PERSON_ID IN (
                        SELECT
                            distinct person_id
                        FROM
                            `""" + os.environ["WORKSPACE_CDR"] + """.
↪cb_search_person` cb_search_person
                        WHERE
                            cb_search_person.person_id IN (
                                SELECT
                                    person_id
                                FROM
                                    `""" + os.environ["WORKSPACE_CDR"] + """.
↪person` p
                                WHERE
                                    race_concept_id IN (8516)
```

```
                            )
                            AND cb_search_person.person_id IN (
                                SELECT
                                    criteria.person_id
                                FROM
                                    (SELECT
                                        DISTINCT person_id,
                                        entry_date,
                                        concept_id
                                    FROM
                                        `""" + os.environ["WORKSPACE_CDR"] +␣
↪"""".cb_search_all_events`
                                    WHERE
                                        (
                                            concept_id IN (
                                                SELECT
                                                    DISTINCT c.concept_id
                                                FROM
                                                    `""" + os.
↪environ["WORKSPACE_CDR"] + """.cb_criteria` c
                                                JOIN
                                                    (
                                                        select
                                                            cast(cr.id as␣
↪string) as id
                                                        FROM
                                                            `""" + os.
↪environ["WORKSPACE_CDR"] + """.cb_criteria` cr
                                                        WHERE
                                                            concept_id IN␣
↪(4152280)
                                                            AND full_text LIKE␣
↪'%_rank1]%'
                                                    ) a
                                                        ON (
                                                            c.path LIKE␣
↪CONCAT('%.',
                                                        a.id,
                                                        '.%')
                                                        OR c.path LIKE␣
↪CONCAT('%.',
                                                        a.id)
                                                        OR c.path LIKE CONCAT(a.
↪id,
                                                        '.%')
                                                        OR c.path = a.id)
```

```
                                                WHERE
                                                    is_standard = 1
                                                    AND is_selectable = 1
                                                )
                                                AND is_standard = 1
                                            )
                                        ) criteria
                                    )
                                    AND cb_search_person.person_id NOT IN (
                                        SELECT
                                            criteria.person_id
                                        FROM
                                            (SELECT
                                                DISTINCT person_id,
                                                entry_date,
                                                concept_id
                                            FROM
                                                `""" + os.
↪environ["WORKSPACE_CDR"] + """.cb_search_all_events`
                                            WHERE
                                                (
                                                    concept_id IN (
                                                        SELECT
                                                            DISTINCT c.
↪concept_id
                                                        FROM
                                                            `""" + os.
↪environ["WORKSPACE_CDR"] + """.cb_criteria` c
                                                        JOIN
                                                            (
                                                                select
                                                                    cast(cr.id␣
↪as string) as id
                                                                FROM
                                                                    `""" + os.
↪environ["WORKSPACE_CDR"] + """.cb_criteria` cr
                                                                WHERE
                                                                    concept_id␣
↪IN (436665)
                                                                    AND␣
↪full_text LIKE '%_rank1]%'
                                                            ) a
                                                                ON (
                                                                    c.path LIKE␣
↪CONCAT('%.',
                                                                    a.id,
```

```
                                                                    '.%')
                                                                    OR c.path LIKE␣
↪CONCAT('%.',
                                                                    a.id)
                                                                    OR c.path LIKE␣
↪CONCAT(a.id,
                                                                    '.%')
                                                                    OR c.path = a.
↪id)
                                                                WHERE
                                                                    is_standard = 1
                                                                    AND␣
↪is_selectable = 1
                                                                )
                                                                AND is_standard = 1
                                                            )
                                                        ) criteria
                                                    ) ))
                                        ) c_occurrence
                                    LEFT JOIN
                                        `""" + os.environ["WORKSPACE_CDR"] + """.
↪concept` c_standard_concept
                                            ON c_occurrence.condition_concept_id =␣
↪c_standard_concept.concept_id"""

dataset_68803172_condition_df = pandas.read_gbq(
    dataset_68803172_condition_sql,
    dialect="standard",
    use_bqstorage_api=("BIGQUERY_STORAGE_API_ENABLED" in os.environ),
    progress_bar_type="tqdm_notebook")

dataset_68803172_condition_df.head(5)
```

```
Downloading:   0%|          | 0/4727885 [00:00<?, ?rows/s]
```

[3]:

	person_id	standard_concept_name	standard_concept_code	condition_start_datetime	condition_end_datetime
0	1736879	Cystocele	252005008	2022-02-12 03:41:17+00:00	2022-02-12 03:41:17+00:00
1	2111039	Circadian rhythm sleep disorder of shift work …	713498009	2018-07-15 00:00:00+00:00	2018-07-15 00:00:00+00:00
2	1731801	Circadian rhythm sleep disorder of shift work …	713498009	2020-09-23 00:00:00+00:00	2020-09-23 00:00:00+00:00
3	1556378	Circadian rhythm sleep disorder of shift work …	713498009	2018-06-05 00:00:00+00:00	2018-06-05 00:00:00+00:00
4	3201836	Tinea corporis			

```
4                84849002 2021-03-25 18:15:00+00:00 2021-03-25 18:15:00+00:00
```

[4]:
```
import pandas
import os

# This query represents dataset "HAP464 Antidepressants" for domain "drug" and␣
 ↪was generated for All of Us Registered Tier Dataset v7
dataset_31628560_drug_sql = """
    SELECT
        d_exposure.person_id,
        d_standard_concept.concept_name as standard_concept_name,
        d_standard_concept.concept_code as standard_concept_code,
        d_exposure.drug_exposure_start_datetime,
        d_exposure.drug_exposure_end_datetime,
        d_exposure.verbatim_end_date,
        d_exposure.refills,
        d_exposure.quantity,
        d_exposure.days_supply
    FROM
        ( SELECT
            *
        FROM
            `""" + os.environ["WORKSPACE_CDR"] + """.drug_exposure` d_exposure
        WHERE
            (
                drug_concept_id IN  (
                    SELECT
                        DISTINCT ca.descendant_id
                    FROM
                        `""" + os.environ["WORKSPACE_CDR"] + """.
 ↪cb_criteria_ancestor` ca
                    JOIN
                        (
                            select
                                distinct c.concept_id
                            FROM
                                `""" + os.environ["WORKSPACE_CDR"] + """.
 ↪cb_criteria` c
                            JOIN
                                (
                                    select
                                        cast(cr.id as string) as id
                                    FROM
                                        `""" + os.environ["WORKSPACE_CDR"] +␣
 ↪"""".cb_criteria` cr
                                    WHERE
                                        concept_id IN (
```

```
                                        21604686, 21604729, 21604788
                                    )
                                    AND full_text LIKE '%_rank1]%'
                            ) a
                                ON (
                                    c.path LIKE CONCAT('%.',
                                a.id,
                                '.%')
                                OR c.path LIKE CONCAT('%.',
                                a.id)
                                OR c.path LIKE CONCAT(a.id,
                                '.%')
                                OR c.path = a.id)
                            WHERE
                                is_standard = 1
                                AND is_selectable = 1
                            ) b
                                ON (
                                    ca.ancestor_id = b.concept_id
                                )
                        )
                    )
                    AND (
                        d_exposure.PERSON_ID IN (
                            SELECT
                                distinct person_id
                            FROM
                                `""" + os.environ["WORKSPACE_CDR"] + """.
↪cb_search_person` cb_search_person
                            WHERE
                                cb_search_person.person_id IN (
                                    SELECT
                                        person_id
                                    FROM
                                        `""" + os.environ["WORKSPACE_CDR"] +␣
↪""".person` p
                                    WHERE
                                        race_concept_id IN (8516)
                                )
                                AND cb_search_person.person_id IN (
                                    SELECT
                                        criteria.person_id
                                    FROM
                                        (SELECT
                                            DISTINCT person_id,
                                            entry_date,
                                            concept_id
```

```
                                    FROM
                                        `""" + os.environ["WORKSPACE_CDR"]␣
↪+ """.cb_search_all_events`
                                    WHERE
                                        (
                                            concept_id IN (
                                                SELECT
                                                    DISTINCT c.concept_id
                                                FROM
                                                    `""" + os.
↪environ["WORKSPACE_CDR"] + """.cb_criteria` c
                                                JOIN
                                                    (
                                                        select
                                                            cast(cr.id as␣
↪string) as id
                                                        FROM
                                                            `""" + os.
↪environ["WORKSPACE_CDR"] + """.cb_criteria` cr
                                                        WHERE
                                                            concept_id IN␣
↪(4152280)
                                                            AND full_text␣
↪LIKE '%_rank1]%'
                                                    ) a
                                                        ON (
                                                            c.path LIKE␣
↪CONCAT('%.',
                                                            a.id,
                                                            '.%')
                                                            OR c.path LIKE␣
↪CONCAT('%.',
                                                            a.id)
                                                            OR c.path LIKE␣
↪CONCAT(a.id,
                                                            '.%')
                                                            OR c.path = a.id)
                                                    WHERE
                                                        is_standard = 1
                                                        AND is_selectable =␣
↪1
                                                    )
                                                    AND is_standard = 1
                                            )
                                        ) criteria
                                    )
```

```
                                    AND cb_search_person.person_id NOT IN (
                                        SELECT
                                            criteria.person_id
                                        FROM
                                            (SELECT
                                                DISTINCT person_id,
                                                entry_date,
                                                concept_id
                                            FROM
                                                `""" + os.
↪environ["WORKSPACE_CDR"] + """.cb_search_all_events`
                                            WHERE
                                                (
                                                    concept_id IN (
                                                        SELECT
                                                            DISTINCT c.
↪concept_id
                                                        FROM
                                                            `""" + os.
↪environ["WORKSPACE_CDR"] + """.cb_criteria` c
                                                        JOIN
                                                            (
                                                                select
                                                                    cast(cr.
↪id as string) as id
                                                                FROM
                                                                    `""" +␣
↪os.environ["WORKSPACE_CDR"] + """.cb_criteria` cr
                                                                WHERE
                                                                   ␣
↪concept_id IN (436665)
                                                                    AND␣
↪full_text LIKE '%_rank1]%'
                                                            ) a
                                                                ON (
                                                                    c.path␣
↪LIKE CONCAT('%.',
                                                                    a.id,
                                                                    '.%')
                                                                    OR c.path␣
↪LIKE CONCAT('%.',
                                                                    a.id)
                                                                    OR c.path␣
↪LIKE CONCAT(a.id,
                                                                    '.%')
```

```
                                                                    OR c.path =␣
↪a.id)
                                                                WHERE
                                                                    is_standard␣
↪= 1
                                                                    AND␣
↪is_selectable = 1
                                                                )
                                                                AND is_standard␣
↪= 1
                                                        )
                                                    ) criteria
                                                ) ))
                                    ) d_exposure
                                LEFT JOIN
                                    `""" + os.environ["WORKSPACE_CDR"] + """.
↪concept` d_standard_concept
                                        ON d_exposure.drug_concept_id =␣
↪d_standard_concept.concept_id"""

dataset_31628560_drug_df = pandas.read_gbq(
    dataset_31628560_drug_sql,
    dialect="standard",
    use_bqstorage_api=("BIGQUERY_STORAGE_API_ENABLED" in os.environ),
    progress_bar_type="tqdm_notebook")

dataset_31628560_drug_df.head(5)
```

```
Downloading:   0%|          | 0/213283 [00:00<?, ?rows/s]
```

[4]:

	person_id	standard_concept_name	standard_concept_code	drug_exposure_start_datetime	drug_exposure_end_datetime	verbatim_end_date	refills	quantity	days_supply
0	3521063	24 HR venlafaxine 150 MG Extended Release Oral…	729931	2016-05-13 05:00:00+00:00	2016-05-19 05:00:00+00:00	NaT	<NA>	NaN	<NA>
1	1306203	24 HR venlafaxine 150 MG Extended Release Oral…	729931	2018-08-06 05:00:00+00:00	2018-08-31 05:00:00+00:00	NaT	0	28.0	<NA>
2	2700224	24 HR venlafaxine 150 MG Extended Release Oral…	729931	2020-07-10 05:00:00+00:00					
3	1117650	24 HR venlafaxine 150 MG Extended Release Oral…	729931	2012-05-22 17:58:00+00:00					
4	1795485	24 HR venlafaxine 150 MG Extended Release Oral…	729931	2007-10-31 05:00:00+00:00					

```
2  2020-08-17 05:00:00+00:00              NaT        0      30.0         <NA>
3  2012-07-17 17:10:00+00:00              NaT        0      30.0         <NA>
4  2007-12-24 05:00:00+00:00              NaT        0      30.0         <NA>
```

1.2 Data Checkpoint

[5]:
```
# Import libraries with short names
import pandas as pd
import numpy as np
import os
```

[6]:
```
demographics_df = dataset_34828527_person_df  # participant demographics
dead_df = dataset_19455778_observation_df     # dead participants
diseases_df = dataset_68803172_condition_df   # all diseases
antidepressant_df = dataset_31628560_drug_df  # all antidepressants
```

1.3 Data Preparation

1.3.1 Prepare antidepressants dataframe

Group antidepressants using the the grouping file to create grouped_ad_df

[7]:
```
# Read the antidepressant grouping file
ad_grouping_df = pd.read_csv("./data/unique_ad_grouped.csv")
ad_grouping_df.head()
```

[7]:
```
   standard_concept_code                              standard_concept_name  \
0                1000048               doxepin hydrochloride 10 MG Oral Capsule
1                1000052  doxepin hydrochloride 10 MG Oral Capsule [Sine…
2                1000054          doxepin hydrochloride 10 MG/ML Oral Solution
3                1000058             doxepin hydrochloride 100 MG Oral Capsule
4                1000062  doxepin hydrochloride 100 MG Oral Capsule [Sin…

  ad_grouping
0     Doxepin
1     Doxepin
2     Doxepin
3     Doxepin
4     Doxepin
```

[8]:
```
# Convert standard_concept_code to Int64
antidepressant_df['standard_concept_code'] =␣
 ↪antidepressant_df['standard_concept_code'].astype('Int64')

# Left join to group the antidepressants based on standard_concept_code
grouped_ad_df = pd.merge(antidepressant_df,␣
 ↪ad_grouping_df[['standard_concept_code', 'ad_grouping']],
                      on='standard_concept_code', how='left')
grouped_ad_df.head()
```

[8]:
```
   person_id                              standard_concept_name  \
0    3521063  24 HR venlafaxine 150 MG Extended Release Oral…
1    1306203  24 HR venlafaxine 150 MG Extended Release Oral…
2    2700224  24 HR venlafaxine 150 MG Extended Release Oral…
3    1117650  24 HR venlafaxine 150 MG Extended Release Oral…
4    1795485  24 HR venlafaxine 150 MG Extended Release Oral…

   standard_concept_code drug_exposure_start_datetime  \
0                 729931    2016-05-13 05:00:00+00:00
1                 729931    2018-08-06 05:00:00+00:00
2                 729931    2020-07-10 05:00:00+00:00
3                 729931    2012-05-22 17:58:00+00:00
4                 729931    2007-10-31 05:00:00+00:00

  drug_exposure_end_datetime verbatim_end_date  refills  quantity  \
0  2016-05-19 05:00:00+00:00               NaT     <NA>       NaN
1  2018-08-31 05:00:00+00:00               NaT        0      28.0
2  2020-08-17 05:00:00+00:00               NaT        0      30.0
3  2012-07-17 17:10:00+00:00               NaT        0      30.0
4  2007-12-24 05:00:00+00:00               NaT        0      30.0

   days_supply  ad_grouping
0         <NA>  Venlafaxine
1         <NA>  Venlafaxine
2         <NA>  Venlafaxine
3         <NA>  Venlafaxine
4         <NA>  Venlafaxine
```

[9]:
```
# Number of participants prior to dates processing
len(grouped_ad_df['person_id'].unique().tolist())
```

[9]:
```
7081
```

[10]:
```
# Drop unneeded columns will use AD grouping moving forward
grouped_ad_df = grouped_ad_df.drop(columns=['standard_concept_name',␣
 ↪'standard_concept_code'])
grouped_ad_df.head()
```

[10]:
```
   person_id drug_exposure_start_datetime drug_exposure_end_datetime  \
0    3521063    2016-05-13 05:00:00+00:00  2016-05-19 05:00:00+00:00
1    1306203    2018-08-06 05:00:00+00:00  2018-08-31 05:00:00+00:00
2    2700224    2020-07-10 05:00:00+00:00  2020-08-17 05:00:00+00:00
3    1117650    2012-05-22 17:58:00+00:00  2012-07-17 17:10:00+00:00
4    1795485    2007-10-31 05:00:00+00:00  2007-12-24 05:00:00+00:00

  verbatim_end_date  refills  quantity  days_supply  ad_grouping
0               NaT     <NA>       NaN         <NA>  Venlafaxine
```

```
1                NaT         0       28.0          <NA>  Venlafaxine
2                NaT         0       30.0          <NA>  Venlafaxine
3                NaT         0       30.0          <NA>  Venlafaxine
4                NaT         0       30.0          <NA>  Venlafaxine
```

Fill out data for antidepressant dates

- Start date (first 3 trials)
- End date (first 3 trials)
- Dosage days (first 3 trials)
- Up to 4 weeks is just random non adherence

Note: Process followed only allows for the dates for the first 3 trials of the antidepressant. If the antidepressant is prescribed again after the first 3 trials, these succeeding trial periods will not be included. This means that the antidepressant only has 3 chances to succeed or fail.

[12]:
```
grouped_ad_df.shape[0] # number of rows with duplicates
```

[12]:
```
213283
```

[13]:
```
# Remove duplicate rows
grouped_ad_df = grouped_ad_df.drop_duplicates(
    subset=['person_id',
            'drug_exposure_start_datetime',
            'drug_exposure_end_datetime',
            'days_supply',
            'ad_grouping'])
```

[14]:
```
grouped_ad_df.shape[0] # number of rows without duplicates
```

[14]:
```
177939
```

[15]:
```
grouped_ad_df.head()
```

[15]:
```
   person_id drug_exposure_start_datetime drug_exposure_end_datetime  \
0    3521063    2016-05-13 05:00:00+00:00  2016-05-19 05:00:00+00:00
1    1306203    2018-08-06 05:00:00+00:00  2018-08-31 05:00:00+00:00
2    2700224    2020-07-10 05:00:00+00:00  2020-08-17 05:00:00+00:00
3    1117650    2012-05-22 17:58:00+00:00  2012-07-17 17:10:00+00:00
4    1795485    2007-10-31 05:00:00+00:00  2007-12-24 05:00:00+00:00

  verbatim_end_date  refills  quantity  days_supply  ad_grouping
0               NaT     <NA>       NaN          <NA>  Venlafaxine
1               NaT        0      28.0          <NA>  Venlafaxine
2               NaT        0      30.0          <NA>  Venlafaxine
3               NaT        0      30.0          <NA>  Venlafaxine
4               NaT        0      30.0          <NA>  Venlafaxine
```

[16]:
```
# import library
from datetime import datetime, timedelta
```

[17]:
```
# Initialize ad_dates_df to store dosage dates and days for each antidepressant␣
 ↪group per person
# Define a dataframe with the desired structure
ad_dates_df = pd.DataFrame(columns=[
    "person_id",
    "ad_grouping",
    "trial_number",
    "earliest_ad_start",
    "earliest_ad_end",
    "dosage_days"
])

# Check datatypes for match with required processing
print(ad_dates_df.dtypes)
```

```
person_id            object
ad_grouping          object
trial_number         object
earliest_ad_start    object
earliest_ad_end      object
dosage_days          object
dtype: object
```

[18]:
```
# Change integers to nullable ints
ad_dates_df['person_id'] = ad_dates_df['person_id'].astype('Int64')
ad_dates_df['trial_number'] = ad_dates_df['trial_number'].astype('Int64')
ad_dates_df['dosage_days'] = ad_dates_df['dosage_days'].astype('Int64')

# Change dates to proper dates data types
ad_dates_df['earliest_ad_start'] = pd.
 ↪to_datetime(ad_dates_df['earliest_ad_start'])
ad_dates_df['earliest_ad_end'] = pd.to_datetime(ad_dates_df['earliest_ad_end'])

# Print the DataFrame with updated datatypes
print(ad_dates_df.dtypes)

# Retain data type for ad_grouping as object / string
```

```
person_id                     Int64
ad_grouping                  object
trial_number                  Int64
earliest_ad_start    datetime64[ns]
earliest_ad_end      datetime64[ns]
dosage_days                   Int64
dtype: object
```

[19]:
```
ad_dates_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 0 entries
Data columns (total 6 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   person_id          0 non-null      Int64
 1   ad_grouping        0 non-null      object
 2   trial_number       0 non-null      Int64
 3   earliest_ad_start  0 non-null      datetime64[ns]
 4   earliest_ad_end    0 non-null      datetime64[ns]
 5   dosage_days        0 non-null      Int64
dtypes: Int64(3), datetime64[ns](2), object(1)
memory usage: 124.0+ bytes
```

[20]:
```
# define function to calculate end dates and dosage days given the values of␣
 ↪the end date and days supplies columns
def calculate_dosage_days(row, trial_end, trial_dosage):

    # end date and days supply do not have values
    if (pd.isna(row['drug_exposure_end_datetime']) and
        (pd.isna(row['days_supply']) or row['days_supply'] == 0)):

        # take start date as end date and increment dosage (one time dose)
        trial_end = row['drug_exposure_start_datetime']
        trial_dosage += 1

    # end date has value but days supply does not
    elif (pd.isna(row['days_supply']) or row['days_supply'] == 0):
        # take current end date and assume same dosage days as time elapse
        trial_end = row['drug_exposure_end_datetime']
        date_diff = row['drug_exposure_end_datetime'] -␣
 ↪row['drug_exposure_start_datetime']
        trial_dosage += date_diff.days + 1

    # end date has no value but days supply has
    elif pd.isna(row['drug_exposure_end_datetime']):
        # increment dosage days by days supply and assume same number of days
        # elapsed from start date to get end date
        trial_end = row['drug_exposure_start_datetime'] +␣
 ↪timedelta(days=row['days_supply']) - timedelta(days=1)
        trial_dosage += row['days_supply']

    # end date and days supply have values
    else:
        # get current end date and increment dosage days by days supply
```

```
        trial_end = row['drug_exposure_end_datetime']
        trial_dosage += row['days_supply']

    return trial_end, trial_dosage
```

[21]:
```
# define function to process antidepressant dates for each person
def process_person(person, person_subset):
    ad_dates_list = []

    # loop through every antidepressant the person is taking
    for ad in person_subset['ad_grouping'].unique():
        # create a subset for the antidepressant for the person
        person_ad_subset = person_subset[person_subset['ad_grouping'] == ad]

        # Get the earliest antidepressant start date
        earliest_ad_start = person_ad_subset['drug_exposure_start_datetime'].
 ↪min()

        # Sort dosage data for the person and antidepressant by start date
        person_ad_subset_sorted = person_ad_subset.
 ↪sort_values(by='drug_exposure_start_datetime')

        # Initialize values for end date and dosage trackers
        earliest_ad_end = earliest_ad_start
        dosage_days = 0

        ## Initialize trial trackers, there will be a maximum of 3 trials
        trial, trial_end, trial_dosage = 1, earliest_ad_start, 0

        # Loop sorted dosage data and find values for end date amd dosage days
        for index, row in person_ad_subset_sorted.iterrows():

            # if start date is more than 4 weeks from current end date or the␣
 ↪last row  then process trackers
            if (((row['drug_exposure_start_datetime'] - trial_end) >␣
 ↪timedelta(days=28)) or
                (index == person_ad_subset_sorted.index[-1])):

                # take the current trial end and trial days to write on the␣
 ↪table
                earliest_ad_end, dosage_days = trial_end, trial_dosage

                # append to the dates being tracked
                ad_dates_list.append({
                    'person_id': person,
                    'ad_grouping': ad,
                    'trial_number': trial,
```

```
                    'earliest_ad_start': earliest_ad_start,
                    'earliest_ad_end': earliest_ad_end,
                    'dosage_days': dosage_days
                })

                # if the last trial or last row then break the loop
                if (trial > 2) or (index == person_ad_subset_sorted.index[-1]):
                    break

                # if not the last trial or last row, initialize the next dosage␣
↪dates
                else:
                    earliest_ad_start = row['drug_exposure_start_datetime']
                    earliest_ad_end = row['drug_exposure_start_datetime']

                # update trial trackers to start tracking next trial
                trial_end = row['drug_exposure_start_datetime']
                trial += 1
                trial_dosage = 0

            # current end date in tracker is greater than the dates being␣
↪processed then skip
            if ((trial_end > row['drug_exposure_end_datetime']) or
                ((pd.isna(row['drug_exposure_end_datetime']) and (trial_end >␣
↪row['drug_exposure_start_datetime'])))):
                continue

            # calculate trial end date and dosage for the row
            trial_end, trial_dosage = calculate_dosage_days(row, trial_end,␣
↪trial_dosage)


    return ad_dates_list
```

[22]:
```
# Iterate over unique person_ids in grouped_ad_df
for person in grouped_ad_df['person_id'].unique():
    # create a subset for the person
    person_subset = grouped_ad_df[grouped_ad_df['person_id'] == person]

    # get the relevant antidepressant trial dates and dosage days for the person
    ad_dates_list = process_person(person, person_subset)

    # append to the dataframe
    ad_dates_df = pd.concat([ad_dates_df, pd.DataFrame(ad_dates_list)],␣
↪ignore_index=True)

ad_dates_df.head(5)
```

[22]:
```
   person_id  ad_grouping  trial_number          earliest_ad_start  \
0    3521063  Venlafaxine             1  2014-07-30 05:00:00+00:00
1    3521063  Venlafaxine             2  2015-08-25 05:00:00+00:00
2    3521063  Venlafaxine             3  2015-10-27 05:00:00+00:00
3    3521063   Sertraline             1  2012-10-10 05:00:00+00:00
4    3521063   Sertraline             2  2019-05-02 05:00:00+00:00

              earliest_ad_end  dosage_days
0  2014-10-23 05:00:00+00:00           60
1  2015-09-23 05:00:00+00:00           30
2  2015-11-25 05:00:00+00:00           30
3  2013-04-24 05:00:00+00:00          170
4  2019-05-31 05:00:00+00:00           30
```

Fill out antidepressant remission details for each trial

- Check if dosage days for the trial is more than 70 days (10 weeks), dose_70p (1 or 0)
- Check if there are other antidepressants start within the trial, other_ad (1 or 0)
- If dose_70p is 1 and other_ad is 0 for the trial, then Remission is 1 otherwise 0

[23]:
```
# Create the new columns for dose_70p
ad_dates_df['dose_70p'] = np.where(ad_dates_df['dosage_days'] >= 70, 1, 0)

ad_dates_df.head(5)
```

[23]:
```
   person_id  ad_grouping  trial_number          earliest_ad_start  \
0    3521063  Venlafaxine             1  2014-07-30 05:00:00+00:00
1    3521063  Venlafaxine             2  2015-08-25 05:00:00+00:00
2    3521063  Venlafaxine             3  2015-10-27 05:00:00+00:00
3    3521063   Sertraline             1  2012-10-10 05:00:00+00:00
4    3521063   Sertraline             2  2019-05-02 05:00:00+00:00

              earliest_ad_end  dosage_days  dose_70p
0  2014-10-23 05:00:00+00:00           60         0
1  2015-09-23 05:00:00+00:00           30         0
2  2015-11-25 05:00:00+00:00           30         0
3  2013-04-24 05:00:00+00:00          170         1
4  2019-05-31 05:00:00+00:00           30         0
```

[24]:
```
# Create functions to calculate starting other antidepressants within the trial␣
 ↪period

# Function to compute oth_ad columns
def compute_other_ad (row, person_counts):
    # if only one row then there is no history of other antidepressants
    if person_counts[row['person_id']] == 1:
        return 0
```

```
    else: # check if another antidepressant was started within 10 weeks or 70␣
↪days
        trial_start = row['earliest_ad_start']
        trial_end = trial_start + pd.Timedelta(days=70)

        # Check other rows for the same person_id for other antidepressants
        other_rows = ad_dates_df[(ad_dates_df['person_id'] == row['person_id'])␣
↪&
                                 (ad_dates_df['ad_grouping'] !=␣
↪row['ad_grouping'])]

        # find row where antidepressant started within trial period return 1 if␣
↪found
        for _, other_row in other_rows.iterrows():
            ad_start = other_row['earliest_ad_start']
            if pd.notna(ad_start) and trial_start <= ad_start <= trial_end:
                return 1

        # return 0 if not found
        return 0
```

[25]:
```
# Count the occurrences of each person_id
person_counts = ad_dates_df['person_id'].value_counts()

# Fill values for the use of other antidepressants for the time period
ad_dates_df['other_ad'] = ad_dates_df.apply(compute_other_ad, axis=1,␣
↪args=(person_counts,))

ad_dates_df.head()
```

[25]:

	person_id	ad_grouping	trial_number	earliest_ad_start	earliest_ad_end	dosage_days	dose_70p	other_ad
0	3521063	Venlafaxine	1	2014-07-30 05:00:00+00:00	2014-10-23 05:00:00+00:00	60	0	0
1	3521063	Venlafaxine	2	2015-08-25 05:00:00+00:00	2015-09-23 05:00:00+00:00	30	0	0
2	3521063	Venlafaxine	3	2015-10-27 05:00:00+00:00	2015-11-25 05:00:00+00:00	30	0	0
3	3521063	Sertraline	1	2012-10-10 05:00:00+00:00	2013-04-24 05:00:00+00:00	170	1	0
4	3521063	Sertraline	2	2019-05-02 05:00:00+00:00	2019-05-31 05:00:00+00:00	30	0	0

[26]:
```
# If dose_70p is 1 and other_ad is 0 for the trial, then Remission is 1␣
↪otherwise 0
```

```
ad_dates_df['Remission'] = np.where((ad_dates_df["dose_70p"] == 1) &␣
 ↪(ad_dates_df["other_ad"] == 0), 1, 0)

ad_dates_df.head()
```

[26]:

	person_id	ad_grouping	trial_number	earliest_ad_start	earliest_ad_end	dosage_days	dose_70p	other_ad	Remission
0	3521063	Venlafaxine	1	2014-07-30 05:00:00+00:00	2014-10-23 05:00:00+00:00	60	0	0	0
1	3521063	Venlafaxine	2	2015-08-25 05:00:00+00:00	2015-09-23 05:00:00+00:00	30	0	0	0
2	3521063	Venlafaxine	3	2015-10-27 05:00:00+00:00	2015-11-25 05:00:00+00:00	30	0	0	0
3	3521063	Sertraline	1	2012-10-10 05:00:00+00:00	2013-04-24 05:00:00+00:00	170	1	0	1
4	3521063	Sertraline	2	2019-05-02 05:00:00+00:00	2019-05-31 05:00:00+00:00	30	0	0	0

1.3.2 Create analysis dataframe

From the flattended antidepressant dataframe, we can use this as a base for our analysis dataframe and combine with person the person dataframe

[27]:
```
# Create analysis dataframe by left joining with demographics dataframe and␣
 ↪then the dead dataframe
analysis_df = pd.merge(ad_dates_df, demographics_df, on='person_id', how='left')

# Rename column to date_of_death in dead_df
dead_df = dead_df.rename(columns={'observation_datetime': 'date_of_death'})

# left join with analysis_df on person_id
analysis_df = pd.merge(analysis_df, dead_df, on='person_id', how='left')

analysis_df.head()
```

[27]:

	person_id	ad_grouping	trial_number	earliest_ad_start	earliest_ad_end	dosage_days	dose_70p	other_ad	Remission
0	3521063	Venlafaxine	1	2014-07-30 05:00:00+00:00	2014-10-23 05:00:00+00:00	60	0	0	0
1	3521063	Venlafaxine	2	2015-08-25 05:00:00+00:00	2015-09-23 05:00:00+00:00	30	0	0	0
2	3521063	Venlafaxine	3	2015-10-27 05:00:00+00:00	2015-11-25 05:00:00+00:00	30	0	0	0
3	3521063	Sertraline	1	2012-10-10 05:00:00+00:00	2013-04-24 05:00:00+00:00	170	1	0	1
4	3521063	Sertraline	2	2019-05-02 05:00:00+00:00	2019-05-31 05:00:00+00:00	30	0	0	0

```
3  2013-04-24 05:00:00+00:00          170         1         0          1
4  2019-05-31 05:00:00+00:00           30         0         0          0

   gender_concept_id  gender             date_of_birth  ethnicity_concept_id  \
0           45878463  Female 1973-09-11 00:00:00+00:00              38003564
1           45878463  Female 1973-09-11 00:00:00+00:00              38003564
2           45878463  Female 1973-09-11 00:00:00+00:00              38003564
3           45878463  Female 1973-09-11 00:00:00+00:00              38003564
4           45878463  Female 1973-09-11 00:00:00+00:00              38003564

                ethnicity  sex_at_birth_concept_id sex_at_birth date_of_death
0  Not Hispanic or Latino                 45878463       Female           NaT
1  Not Hispanic or Latino                 45878463       Female           NaT
2  Not Hispanic or Latino                 45878463       Female           NaT
3  Not Hispanic or Latino                 45878463       Female           NaT
4  Not Hispanic or Latino                 45878463       Female           NaT
```

[28]:
```
analysis_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 28310 entries, 0 to 28309
Data columns (total 17 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   person_id                28310 non-null  Int64
 1   ad_grouping              28310 non-null  object
 2   trial_number             28310 non-null  Int64
 3   earliest_ad_start        28310 non-null  object
 4   earliest_ad_end          28310 non-null  object
 5   dosage_days              28310 non-null  Int64
 6   dose_70p                 28310 non-null  int64
 7   other_ad                 28310 non-null  int64
 8   Remission                28310 non-null  int64
 9   gender_concept_id        28310 non-null  Int64
 10  gender                   28310 non-null  object
 11  date_of_birth            28310 non-null  datetime64[ns, UTC]
 12  ethnicity_concept_id     28310 non-null  Int64
 13  ethnicity                28310 non-null  object
 14  sex_at_birth_concept_id  28310 non-null  Int64
 15  sex_at_birth             28310 non-null  object
 16  date_of_death            22 non-null     datetime64[ns, UTC]
dtypes: Int64(6), datetime64[ns, UTC](2), int64(3), object(6)
memory usage: 3.8+ MB
```

[29]:
```
# Set data types that are not appropriate after creation of new dataframe

# Dates should be correctly typed
```

```
analysis_df['earliest_ad_start'] = pd.
 ↪to_datetime(analysis_df['earliest_ad_start'])
analysis_df['earliest_ad_end'] = pd.to_datetime(analysis_df['earliest_ad_end'])

# Print the DataFrame with updated datatypes
print(analysis_df.dtypes)
```

```
person_id                                Int64
ad_grouping                             object
trial_number                             Int64
earliest_ad_start          datetime64[ns, UTC]
earliest_ad_end            datetime64[ns, UTC]
dosage_days                              Int64
dose_70p                                 int64
other_ad                                 int64
Remission                                int64
gender_concept_id                        Int64
gender                                  object
date_of_birth              datetime64[ns, UTC]
ethnicity_concept_id                     Int64
ethnicity                               object
sex_at_birth_concept_id                  Int64
sex_at_birth                            object
date_of_death              datetime64[ns, UTC]
dtype: object
```

Calculate age at antidepressant start and age at death

[30]:
```
# Calculate age in years at adantidepressant start
analysis_df['age_at_ad'] = (analysis_df['earliest_ad_start'] -␣
 ↪analysis_df['date_of_birth']).dt.days / 365.25

# For 'age_at_death', only calculate where 'date_of_death' is not null
# This ensures we don't get NaN values for people who are still alive
analysis_df.loc[analysis_df['date_of_death'].notnull(), 'age_at_death'] = (
    analysis_df['date_of_death'] - analysis_df['date_of_birth']).dt.days / 365.
 ↪25

analysis_df.head()
```

[30]:

	person_id	ad_grouping	trial_number	earliest_ad_start	\
0	3521063	Venlafaxine	1	2014-07-30 05:00:00+00:00	
1	3521063	Venlafaxine	2	2015-08-25 05:00:00+00:00	
2	3521063	Venlafaxine	3	2015-10-27 05:00:00+00:00	
3	3521063	Sertraline	1	2012-10-10 05:00:00+00:00	
4	3521063	Sertraline	2	2019-05-02 05:00:00+00:00	

	earliest_ad_end	dosage_days	dose_70p	other_ad	Remission	\

```
0 2014-10-23 05:00:00+00:00           60         0         0          0
1 2015-09-23 05:00:00+00:00           30         0         0          0
2 2015-11-25 05:00:00+00:00           30         0         0          0
3 2013-04-24 05:00:00+00:00          170         1         0          1
4 2019-05-31 05:00:00+00:00           30         0         0          0

   gender_concept_id  gender             date_of_birth  ethnicity_concept_id  \
0           45878463  Female 1973-09-11 00:00:00+00:00              38003564
1           45878463  Female 1973-09-11 00:00:00+00:00              38003564
2           45878463  Female 1973-09-11 00:00:00+00:00              38003564
3           45878463  Female 1973-09-11 00:00:00+00:00              38003564
4           45878463  Female 1973-09-11 00:00:00+00:00              38003564

                ethnicity  sex_at_birth_concept_id sex_at_birth date_of_death  \
0  Not Hispanic or Latino                 45878463       Female           NaT
1  Not Hispanic or Latino                 45878463       Female           NaT
2  Not Hispanic or Latino                 45878463       Female           NaT
3  Not Hispanic or Latino                 45878463       Female           NaT
4  Not Hispanic or Latino                 45878463       Female           NaT

   age_at_ad  age_at_death
0  40.881588           NaN
1  41.952088           NaN
2  42.124572           NaN
3  39.080082           NaN
4  45.637235           NaN
```

Create age groups Age groups will be as follows: - 0 to 12 - 13 to 19 - 20 to 40 - 41 to 64 - 65 to 79 - 80 to 89

[31]:
```
# Get the age range in the data
# Find the minimum age at ad
min_age_at_ad = analysis_df['age_at_ad'].min()

# Find the maximum age at ad
max_age_at_ad = analysis_df['age_at_ad'].max()

print(f"Minimum age at ad: {min_age_at_ad}")
print(f"Maximum age at ad: {max_age_at_ad}")
```

```
Minimum age at ad: 7.709787816563997
Maximum age at ad: 88.09034907597535
```

[32]:
```
# Define bins for age ranges
bins = [0, 12, 19, 40, 64, 79, 89]

# Define labels for age groups
```

```
labels = ['age 0-12', 'age 13-19', 'age 20-40', 'age 41-64', 'age 65-79', 'age␣
 ↪80-89']

# Create a categorical column for age ranges
analysis_df['age_range'] = pd.cut(analysis_df['age_at_ad'], bins=bins,␣
 ↪labels=labels, right=True, include_lowest=True)

# Create dummy variables for the age ranges
age_dummies = pd.get_dummies(analysis_df['age_range']).astype(int)

# Concatenate the dummy variables with the original DataFrame
analysis_df = pd.concat([analysis_df, age_dummies], axis=1)

analysis_df.head()
```

[32]:

	person_id	ad_grouping	trial_number	earliest_ad_start	earliest_ad_end	dosage_days	dose_70p	other_ad	Remission	gender_concept_id	…	date_of_death	age_at_ad	age_at_death	age_range	age 0-12	age 13-19	age 20-40	age 41-64	age 65-79	age 80-89
0	3521063	Venlafaxine	1	2014-07-30 05:00:00+00:00	2014-10-23 05:00:00+00:00	60	0	0	0	45878463	…	NaT	40.881588	NaN	age 41-64	0	0	0	1	0	0
1	3521063	Venlafaxine	2	2015-08-25 05:00:00+00:00	2015-09-23 05:00:00+00:00	30	0	0	0	45878463	…	NaT	41.952088	NaN	age 41-64	0	0	0	1	0	0
2	3521063	Venlafaxine	3	2015-10-27 05:00:00+00:00	2015-11-25 05:00:00+00:00	30	0	0	0	45878463	…	NaT	42.124572	NaN	age 41-64	0	0	0	1	0	0
3	3521063	Sertraline	1	2012-10-10 05:00:00+00:00	2013-04-24 05:00:00+00:00	170	1	0	1	45878463	…	NaT	39.080082	NaN	age 20-40	0	0	1	0	0	0
4	3521063	Sertraline	2	2019-05-02 05:00:00+00:00	2019-05-31 05:00:00+00:00	30	0	0	0	45878463	…	NaT	45.637235	NaN	age 41-64	0	0	0	1	0	0

[5 rows x 26 columns]

Indicator for female sex at birth This is the indicator used for the reference study. We will need to use it too to validate the results.

[33]:
```
analysis_df['Female'] = np.where(analysis_df['gender'] == 'Female', 1, 0)

analysis_df.head()
```

[33]:
```
   person_id  ad_grouping  trial_number         earliest_ad_start  \
0    3521063  Venlafaxine             1 2014-07-30 05:00:00+00:00
1    3521063  Venlafaxine             2 2015-08-25 05:00:00+00:00
2    3521063  Venlafaxine             3 2015-10-27 05:00:00+00:00
3    3521063   Sertraline             1 2012-10-10 05:00:00+00:00
4    3521063   Sertraline             2 2019-05-02 05:00:00+00:00

             earliest_ad_end  dosage_days  dose_70p  other_ad  Remission  \
0 2014-10-23 05:00:00+00:00           60         0         0          0
1 2015-09-23 05:00:00+00:00           30         0         0          0
2 2015-11-25 05:00:00+00:00           30         0         0          0
3 2013-04-24 05:00:00+00:00          170         1         0          1
4 2019-05-31 05:00:00+00:00           30         0         0          0

   gender_concept_id  …  age_at_ad age_at_death  age_range age 0-12  \
0           45878463  …  40.881588          NaN  age 41-64        0
1           45878463  …  41.952088          NaN  age 41-64        0
2           45878463  …  42.124572          NaN  age 41-64        0
3           45878463  …  39.080082          NaN  age 20-40        0
4           45878463  …  45.637235          NaN  age 41-64        0

   age 13-19 age 20-40 age 41-64  age 65-79  age 80-89 Female
0          0         0         1          0          0      1
1          0         0         1          0          0      1
2          0         0         1          0          0      1
3          0         1         0          0          0      1
4          0         0         1          0          0      1

[5 rows x 27 columns]
```

1.3.3 Prepare diseases dataframe

Remove duplicates

[34]:
```
diseases_df.shape[0] # number of rows with duplicates
```

[34]:
```
4727885
```

[35]:
```
# Remove duplicate rows
diseases_df = diseases_df.drop_duplicates(
    subset=['person_id',
            'standard_concept_code',
            'condition_start_datetime',
            'condition_end_datetime'])
```

[36]:
```
diseases_df.shape[0] # number of rows without duplicates
```

[36]:
```
3813477
```

[37]:
```
diseases_df.head()
```

[37]:
```
   person_id                                standard_concept_name  \
0    1736879                                             Cystocele
1    2111039  Circadian rhythm sleep disorder of shift work …
2    1731801  Circadian rhythm sleep disorder of shift work …
3    1556378  Circadian rhythm sleep disorder of shift work …
4    3201836                                      Tinea corporis

  standard_concept_code  condition_start_datetime    condition_end_datetime
0             252005008 2022-02-12 03:41:17+00:00 2022-02-12 03:41:17+00:00
1             713498009 2018-07-15 00:00:00+00:00 2018-07-15 00:00:00+00:00
2             713498009 2020-09-23 00:00:00+00:00 2020-09-23 00:00:00+00:00
3             713498009 2018-06-05 00:00:00+00:00 2018-06-05 00:00:00+00:00
4              84849002 2021-03-25 18:15:00+00:00 2021-03-25 18:15:00+00:00
```

Combine with analysis dataframe (analysis_df on the left)

[38]:
```
analysis_df = pd.merge(analysis_df, diseases_df, on='person_id', how='left')

analysis_df.head()
```

[38]:
```
   person_id  ad_grouping  trial_number         earliest_ad_start  \
0    3521063  Venlafaxine             1 2014-07-30 05:00:00+00:00
1    3521063  Venlafaxine             1 2014-07-30 05:00:00+00:00
2    3521063  Venlafaxine             1 2014-07-30 05:00:00+00:00
3    3521063  Venlafaxine             1 2014-07-30 05:00:00+00:00
4    3521063  Venlafaxine             1 2014-07-30 05:00:00+00:00

            earliest_ad_end  dosage_days  dose_70p  other_ad  Remission  \
0 2014-10-23 05:00:00+00:00           60         0         0          0
1 2014-10-23 05:00:00+00:00           60         0         0          0
2 2014-10-23 05:00:00+00:00           60         0         0          0
3 2014-10-23 05:00:00+00:00           60         0         0          0
4 2014-10-23 05:00:00+00:00           60         0         0          0

   gender_concept_id  … age 13-19 age 20-40  age 41-64 age 65-79  age 80-89  \
0           45878463  …         0         0          1         0          0
1           45878463  …         0         0          1         0          0
2           45878463  …         0         0          1         0          0
3           45878463  …         0         0          1         0          0
4           45878463  …         0         0          1         0          0

  Female standard_concept_name  standard_concept_code  \
```

```
0      1       Acute sinusitis                15805002
1      1       Acute sinusitis                15805002
2      1       Acute sinusitis                15805002
3      1       Acute sinusitis                15805002
4      1       Acute sinusitis                15805002

   condition_start_datetime    condition_end_datetime
0 2008-09-17 05:00:00+00:00 2008-09-17 11:59:59+00:00
1 2008-11-28 05:00:00+00:00 2008-11-28 11:59:59+00:00
2 2008-10-14 05:00:00+00:00 2008-10-14 11:59:59+00:00
3 2008-10-14 05:00:00+00:00 2009-07-30 05:00:00+00:00
4 2010-02-18 05:00:00+00:00 2010-02-18 11:59:59+00:00

[5 rows x 31 columns]
```

Select only diseases that started prior to starting the antidepressant for each trial

[39]:
```
# number of rows prior to clean up
analysis_df.shape[0]
```

[39]:
```
18191201
```

[40]:
```
# Retain only rows where condition_start_datetime is less than earliest_ad_start
analysis_df = analysis_df[analysis_df['condition_start_datetime'] <␣
 ↪analysis_df['earliest_ad_start']]

# Reset index
analysis_df.reset_index(drop=True, inplace=True)
```

[41]:
```
# number of rows after to clean up
analysis_df.shape[0]
```

[41]:
```
6677782
```

[42]:
```
analysis_df.head()
```

[42]:
```
   person_id  ad_grouping  trial_number         earliest_ad_start  \
0    3521063  Venlafaxine             1 2014-07-30 05:00:00+00:00
1    3521063  Venlafaxine             1 2014-07-30 05:00:00+00:00
2    3521063  Venlafaxine             1 2014-07-30 05:00:00+00:00
3    3521063  Venlafaxine             1 2014-07-30 05:00:00+00:00
4    3521063  Venlafaxine             1 2014-07-30 05:00:00+00:00

             earliest_ad_end  dosage_days  dose_70p  other_ad  Remission  \
0 2014-10-23 05:00:00+00:00           60         0         0          0
1 2014-10-23 05:00:00+00:00           60         0         0          0
2 2014-10-23 05:00:00+00:00           60         0         0          0
3 2014-10-23 05:00:00+00:00           60         0         0          0
```

```
4 2014-10-23 05:00:00+00:00           60         0         0          0

   gender_concept_id  … age 13-19 age 20-40  age 41-64 age 65-79  age 80-89  \
0           45878463  …         0         0          1         0          0
1           45878463  …         0         0          1         0          0
2           45878463  …         0         0          1         0          0
3           45878463  …         0         0          1         0          0
4           45878463  …         0         0          1         0          0

  Female standard_concept_name  standard_concept_code  \
0      1       Acute sinusitis               15805002
1      1       Acute sinusitis               15805002
2      1       Acute sinusitis               15805002
3      1       Acute sinusitis               15805002
4      1       Acute sinusitis               15805002

   condition_start_datetime    condition_end_datetime
0 2008-09-17 05:00:00+00:00 2008-09-17 11:59:59+00:00
1 2008-11-28 05:00:00+00:00 2008-11-28 11:59:59+00:00
2 2008-10-14 05:00:00+00:00 2008-10-14 11:59:59+00:00
3 2008-10-14 05:00:00+00:00 2009-07-30 05:00:00+00:00
4 2010-02-18 05:00:00+00:00 2010-02-18 11:59:59+00:00

[5 rows x 31 columns]
```

2 Data Checkpoint

2.1 Store the analysis dataframe in a file

[43]:
```
# Check data types prior to writing file for reference after reading
analysis_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6677782 entries, 0 to 6677781
Data columns (total 31 columns):
 #   Column                    Dtype
---  ------                    -----
 0   person_id                 Int64
 1   ad_grouping               object
 2   trial_number              Int64
 3   earliest_ad_start         datetime64[ns, UTC]
 4   earliest_ad_end           datetime64[ns, UTC]
 5   dosage_days               Int64
 6   dose_70p                  int64
 7   other_ad                  int64
 8   Remission                 int64
 9   gender_concept_id         Int64
```

```
 10  gender                     object
 11  date_of_birth              datetime64[ns, UTC]
 12  ethnicity_concept_id       Int64
 13  ethnicity                  object
 14  sex_at_birth_concept_id    Int64
 15  sex_at_birth               object
 16  date_of_death              datetime64[ns, UTC]
 17  age_at_ad                  float64
 18  age_at_death               float64
 19  age_range                  category
 20  age 0-12                   int64
 21  age 13-19                  int64
 22  age 20-40                  int64
 23  age 41-64                  int64
 24  age 65-79                  int64
 25  age 80-89                  int64
 26  Female                     int64
 27  standard_concept_name      object
 28  standard_concept_code      object
 29  condition_start_datetime   datetime64[ns, UTC]
 30  condition_end_datetime     datetime64[ns, UTC]
dtypes: Int64(6), category(1), datetime64[ns, UTC](6), float64(2), int64(10),
object(6)
memory usage: 1.5+ GB
```

[44]:
```
analysis_df.to_csv('./checkpoint/hap823_analysis_df.csv', index = False)
```

2.2 Succeeding runs can start from here by reading the file and changing the datatypes

[45]:
```
# Import libraries with short names
import pandas as pd
import numpy as np
import os
```

[46]:
```
analysis_df = pd.read_csv('./checkpoint/hap823_analysis_df.csv', low_memory =␣
 ↪False)
```

[47]:
```
# Check data types after loading from file for reference on which datatypes to␣
 ↪change
analysis_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6677782 entries, 0 to 6677781
Data columns (total 31 columns):
 #   Column                     Dtype
---  ------                     -----
 0   person_id                  int64
```

```
 1   ad_grouping               object
 2   trial_number              int64
 3   earliest_ad_start         object
 4   earliest_ad_end           object
 5   dosage_days               int64
 6   dose_70p                  int64
 7   other_ad                  int64
 8   Remission                 int64
 9   gender_concept_id         int64
 10  gender                    object
 11  date_of_birth             object
 12  ethnicity_concept_id      int64
 13  ethnicity                 object
 14  sex_at_birth_concept_id   int64
 15  sex_at_birth              object
 16  date_of_death             object
 17  age_at_ad                 float64
 18  age_at_death              float64
 19  age_range                 object
 20  age 0-12                  int64
 21  age 13-19                 int64
 22  age 20-40                 int64
 23  age 41-64                 int64
 24  age 65-79                 int64
 25  age 80-89                 int64
 26  Female                    int64
 27  standard_concept_name     object
 28  standard_concept_code     int64
 29  condition_start_datetime  object
 30  condition_end_datetime    object
dtypes: float64(2), int64(17), object(12)
memory usage: 1.5+ GB
```

[48]:
```
# Change datatypes to aid processing
# Integer columns (imported correctly)
#['person_id ', 'trial_number', 'dosage_days', 'dose_70p', 'other_ad',␣
 ↪'Remission', 'gender_concept_id',
# 'ethnicity_concept_id', 'sex_at_birth_concept_id', 'age 0-12', 'age 13-19',␣
 ↪'age 20-40', 'age 41-64', 'age 65-79',
# 'age 80-89', 'Female', 'disease_group']

# Float columns (imported correctly)
#['age_at_ad', 'age_at_death']

# Date columns
dates_col = ['earliest_ad_start', 'earliest_ad_end', 'date_of_birth',␣
 ↪'date_of_death',
```

```
            'condition_start_datetime', 'condition_end_datetime']

for col in dates_col:
    analysis_df[col] = pd.to_datetime(analysis_df[col], errors='coerce')

#Rename columns for diseases
analysis_df = analysis_df.rename(columns={'standard_concept_name':␣
 ↪'disease_name'})
analysis_df = analysis_df.rename(columns={'standard_concept_code':␣
 ↪'disease_code'})
```

[49]:
```
analysis_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6677782 entries, 0 to 6677781
Data columns (total 31 columns):
 #   Column                    Dtype
---  ------                    -----
 0   person_id                 int64
 1   ad_grouping               object
 2   trial_number              int64
 3   earliest_ad_start         datetime64[ns, UTC]
 4   earliest_ad_end           datetime64[ns, UTC]
 5   dosage_days               int64
 6   dose_70p                  int64
 7   other_ad                  int64
 8   Remission                 int64
 9   gender_concept_id         int64
 10  gender                    object
 11  date_of_birth             datetime64[ns, UTC]
 12  ethnicity_concept_id      int64
 13  ethnicity                 object
 14  sex_at_birth_concept_id   int64
 15  sex_at_birth              object
 16  date_of_death             datetime64[ns, UTC]
 17  age_at_ad                 float64
 18  age_at_death              float64
 19  age_range                 object
 20  age 0-12                  int64
 21  age 13-19                 int64
 22  age 20-40                 int64
 23  age 41-64                 int64
 24  age 65-79                 int64
 25  age 80-89                 int64
 26  Female                    int64
 27  disease_name              object
 28  disease_code              int64
 29  condition_start_datetime  datetime64[ns, UTC]
```

```
 30  condition_end_datetime    datetime64[ns, UTC]
dtypes: datetime64[ns, UTC](6), float64(2), int64(17), object(6)
memory usage: 1.5+ GB
```

[50]:
```
analysis_df.head()
```

[50]:
```
   person_id  ad_grouping  trial_number         earliest_ad_start  \
0    3521063  Venlafaxine             1 2014-07-30 05:00:00+00:00
1    3521063  Venlafaxine             1 2014-07-30 05:00:00+00:00
2    3521063  Venlafaxine             1 2014-07-30 05:00:00+00:00
3    3521063  Venlafaxine             1 2014-07-30 05:00:00+00:00
4    3521063  Venlafaxine             1 2014-07-30 05:00:00+00:00

            earliest_ad_end  dosage_days  dose_70p  other_ad  Remission  \
0 2014-10-23 05:00:00+00:00           60         0         0          0
1 2014-10-23 05:00:00+00:00           60         0         0          0
2 2014-10-23 05:00:00+00:00           60         0         0          0
3 2014-10-23 05:00:00+00:00           60         0         0          0
4 2014-10-23 05:00:00+00:00           60         0         0          0

   gender_concept_id  … age 13-19 age 20-40  age 41-64 age 65-79  age 80-89  \
0           45878463  …         0         0          1         0          0
1           45878463  …         0         0          1         0          0
2           45878463  …         0         0          1         0          0
3           45878463  …         0         0          1         0          0
4           45878463  …         0         0          1         0          0

  Female     disease_name  disease_code  condition_start_datetime  \
0      1  Acute sinusitis      15805002 2008-09-17 05:00:00+00:00
1      1  Acute sinusitis      15805002 2008-11-28 05:00:00+00:00
2      1  Acute sinusitis      15805002 2008-10-14 05:00:00+00:00
3      1  Acute sinusitis      15805002 2008-10-14 05:00:00+00:00
4      1  Acute sinusitis      15805002 2010-02-18 05:00:00+00:00

     condition_end_datetime
0 2008-09-17 11:59:59+00:00
1 2008-11-28 11:59:59+00:00
2 2008-10-14 11:59:59+00:00
3 2009-07-30 05:00:00+00:00
4 2010-02-18 11:59:59+00:00

[5 rows x 31 columns]
```

2.2.1 Reduce dimensionality

- Drop unneeded columns

[51]:
```
# Columns to keep
cols_to_keep = ['person_id',
                'ad_grouping',
                'trial_number',
                'earliest_ad_start',
                'Remission',
                'age 13-19', #dropped 'age 0-12' to prevent dummy variable trap
                'age 20-40',
                'age 41-64',
                'age 65-79',
                'age 80-89',
                'Female',
                'disease_code'
               ]

# Select only the columns in cols_to_keep
analysis_df_reduced = analysis_df[cols_to_keep]

analysis_df_reduced.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6677782 entries, 0 to 6677781
Data columns (total 12 columns):
 #   Column             Dtype
---  ------             -----
 0   person_id          int64
 1   ad_grouping        object
 2   trial_number       int64
 3   earliest_ad_start  datetime64[ns, UTC]
 4   Remission          int64
 5   age 13-19          int64
 6   age 20-40          int64
 7   age 41-64          int64
 8   age 65-79          int64
 9   age 80-89          int64
 10  Female             int64
 11  disease_code       int64
dtypes: datetime64[ns, UTC](1), int64(10), object(1)
memory usage: 611.4+ MB
```

2.2.2 Convert disease_codes into dummy variables in store in a new copy of the analysis dataframe

[52]:
```
# Check if disease_code has missing values (place 0 as default if it exists)
analysis_df_reduced['disease_code'].isnull().sum()
```

[52]:
```
0
```

[53]:
```
# Loop through each unique antidepressant to create an analysis dataframe for␣
 ↪each
for ad in analysis_df_reduced['ad_grouping'].unique():
    # Create a subset for the antidepressant
    analysis_subset = analysis_df_reduced[analysis_df_reduced['ad_grouping'] ==␣
 ↪ad]

    # Create a copy containing only primary key columns and disease code to␣
 ↪create dummy variables
    disease_per_trial = analysis_subset[['person_id', 'ad_grouping',␣
 ↪'trial_number', 'disease_code']].copy()

    # Create dummy variables for the disease codes
    disease_dummies = pd.get_dummies(disease_per_trial['disease_code'],␣
 ↪prefix='disease').astype(int)

    # Concatenate the dummy variables with the disease per trial dataframe
    disease_per_trial = pd.concat([disease_per_trial, disease_dummies], axis=1)

    # Drop disease_code column in disease per trial
    disease_per_trial = disease_per_trial.drop(columns=['disease_code'])

    # Drop the disease_code column from the analyis subset dataframe
    analysis_subset = analysis_subset.drop(columns=['disease_code'])

    # Drop columns with less than 30 samples as they do not have enough data to␣
 ↪properly identify distributions
    # Take column sums
    column_sums = disease_per_trial.sum()

    # Identify columns to drop
    columns_to_drop = [col for col in disease_per_trial.columns if col.
 ↪startswith('disease_') and column_sums[col] < 30]

    # Drop columns with less than 30 samples
    disease_per_trial = disease_per_trial.drop(columns=columns_to_drop)

    # Aggregate analysis_df_reduced based on primary keys, use first value for␣
 ↪aggregation
    aggregation_dict_1 = {col: 'first'
                          for col in analysis_subset.columns if col not in␣
 ↪['person_id', 'ad_grouping', 'trial_number']}
    # Group by the primary key columns and aggregate
    analysis_subset = analysis_subset.groupby(['person_id', 'ad_grouping',␣
 ↪'trial_number'],
```

```
                                                        as_index=False).
↪agg(aggregation_dict_1)

   # Aggregate disease per trial based on primary keys, use max value for␣
↪aggregation
   aggregation_dict_2 = {col: 'max'
                         for col in disease_per_trial.columns if col not in␣
↪['person_id', 'ad_grouping', 'trial_number']}

   # Group by the primary key columns and aggregate
   disease_per_trial = disease_per_trial.groupby(['person_id', 'ad_grouping',␣
↪'trial_number'],
                                                 as_index=False).
↪agg(aggregation_dict_2)

   # Left analysis_df_reduced and disease_per_trial using primary keys␣
↪(analysis_df_reduced on the left)
   analysis_subset = pd.merge(analysis_subset, disease_per_trial,␣
↪on=['person_id', 'ad_grouping', 'trial_number'],
                              how='left')

   # create checkpoint files for each subset
   filename = f'./checkpoint/hap823_analysis_subset_{ad}.csv'
   analysis_subset.to_csv(filename, index = False)
```

3 Data Checkpoint

3.1 Restore needed files from the data checkpoint

[3]:
```
# Import libraries with short names
import pandas as pd
import numpy as np
import os
```

[4]:
```
# Open each file and store in a dictionary of dataframes
ad_list = ['Amitriptyline',
           'Bupropion',
           'Citalopram',
           'Desvenlafaxine',
           'Doxepin',
           'Duloxetine',
           'Escitalopram',
           'Fluoxetine',
           'Mirtazapine',
           'Nortriptyline',
           'Paroxetine',
```

```
            'Sertraline',
            'Trazodone',
            'Venlafaxine',
            'Other']

analysis_df_dict = dict()
for ad in ad_list:
    filename = f'./checkpoint/hap823_analysis_subset_{ad}.csv'
    analysis_df_dict[ad] = pd.read_csv(filename, low_memory = False)
```

[6]:
```
# Check the info and head of each file
for ad in ad_list:
    print(f"\n\nAntidepressant: {ad}")
    print(analysis_df_dict[ad].info())
    print(analysis_df_dict[ad].head())
```

```
Antidepressant: Amitriptyline
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1984 entries, 0 to 1983
Columns: 1609 entries, person_id to disease_15960141000119102
dtypes: int64(1607), object(2)
memory usage: 24.4+ MB
None
   person_id    ad_grouping  trial_number          earliest_ad_start  \
0    1000039  Amitriptyline             1  2019-12-18 16:13:00+00:00
1    1000370  Amitriptyline             1  2019-11-01 00:00:00+00:00
2    1004308  Amitriptyline             1  2007-08-06 14:27:00+00:00
3    1005542  Amitriptyline             1  2016-11-08 00:00:00+00:00
4    1010384  Amitriptyline             1  2020-04-06 16:54:00+00:00

   Remission  age 13-19  age 20-40  age 41-64  age 65-79  age 80-89  …  \
0          0          0          0          1          0          0  …
1          0          0          0          0          1          0  …
2          1          0          0          1          0          0  …
3          0          0          0          1          0          0  …
4          0          0          0          1          0          0  …

   disease_1085811000119109  disease_1085911000119103  \
0                         0                         0
1                         0                         0
2                         0                         0
3                         0                         0
4                         0                         0

   disease_1089981000119106  disease_10633351000119109  \
0                         0                          0
```

```
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_10633391000119104  disease_10641391000119104  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_10685111000119102  disease_10743881000119104  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_12246311000119109  disease_15960141000119102
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          1                          0

[5 rows x 1609 columns]


Antidepressant: Bupropion
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2658 entries, 0 to 2657
Columns: 1685 entries, person_id to disease_15960141000119102
dtypes: int64(1683), object(2)
memory usage: 34.2+ MB
None
   person_id ad_grouping  trial_number          earliest_ad_start  Remission  \
0    1000042   Bupropion             1  2019-01-03 00:00:00+00:00          0
1    1000151   Bupropion             1  2005-09-05 07:43:01+00:00          0
2    1000151   Bupropion             2  2014-04-30 20:39:04+00:00          1
3    1000151   Bupropion             3  2014-08-25 22:14:33+00:00          1
4    1001865   Bupropion             1  2012-07-26 00:00:00+00:00          0

   age 13-19  age 20-40  age 41-64  age 65-79  age 80-89  …  \
0          0          0          1          0          0  …
1          0          0          1          0          0  …
2          0          0          1          0          0  …
3          0          0          1          0          0  …
```

```
4          0          1          0          0          0  …

   disease_1092851000119103  disease_10633351000119109  \
0                         0                          0
1                         0                          0
2                         0                          0
3                         0                          0
4                         0                          0

   disease_10633391000119104  disease_10641391000119104  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_10672271000119100  disease_10685111000119102  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_10742471000119108  disease_10743881000119104  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_12246311000119109  disease_15960141000119102
0                          1                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

[5 rows x 1685 columns]


Antidepressant: Citalopram
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2393 entries, 0 to 2392
Columns: 1287 entries, person_id to disease_16837681000119104
dtypes: int64(1285), object(2)
memory usage: 23.5+ MB
None
   person_id ad_grouping  trial_number          earliest_ad_start  Remission  \
```

```
0    1000151  Citalopram            1  2004-08-23 11:52:32+00:00          1
1    1000151  Citalopram            2  2005-05-16 23:17:01+00:00          1
2    1001002  Citalopram            1  2018-12-22 00:05:00+00:00          0
3    1001002  Citalopram            2  2019-01-31 15:14:00+00:00          0
4    1001663  Citalopram            2  2016-08-25 06:00:00+00:00          0

   age 13-19  age 20-40  age 41-64  age 65-79  age 80-89  …  \
0          0          0          1          0          0  …
1          0          0          1          0          0  …
2          0          0          1          0          0  …
3          0          0          1          0          0  …
4          0          0          1          0          0  …

   disease_451041000124103  disease_1079821000119106  \
0                        0                         0
1                        0                         0
2                        0                         0
3                        0                         0
4                        0                         0

   disease_1089981000119106  disease_10633391000119104  \
0                         0                          0
1                         0                          0
2                         0                          0
3                         0                          0
4                         0                          0

   disease_10641391000119104  disease_10685111000119102  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_10743881000119104  disease_15960141000119102  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_16055431000119108  disease_16837681000119104
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0
```

```
[5 rows x 1287 columns]


Antidepressant: Desvenlafaxine
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 68 entries, 0 to 67
Data columns (total 68 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   person_id          68 non-null     int64
 1   ad_grouping        68 non-null     object
 2   trial_number       68 non-null     int64
 3   earliest_ad_start  68 non-null     object
 4   Remission          68 non-null     int64
 5   age 13-19          68 non-null     int64
 6   age 20-40          68 non-null     int64
 7   age 41-64          68 non-null     int64
 8   age 65-79          68 non-null     int64
 9   age 80-89          68 non-null     int64
 10  Female             68 non-null     int64
 11  disease_1201005    68 non-null     int64
 12  disease_3972004    68 non-null     int64
 13  disease_11381005   68 non-null     int64
 14  disease_13370002   68 non-null     int64
 15  disease_16932000   68 non-null     int64
 16  disease_18818009   68 non-null     int64
 17  disease_21897009   68 non-null     int64
 18  disease_23056005   68 non-null     int64
 19  disease_34713006   68 non-null     int64
 20  disease_35298007   68 non-null     int64
 21  disease_35489007   68 non-null     int64
 22  disease_36923009   68 non-null     int64
 23  disease_40739000   68 non-null     int64
 24  disease_43339004   68 non-null     int64
 25  disease_44054006   68 non-null     int64
 26  disease_47505003   68 non-null     int64
 27  disease_50127006   68 non-null     int64
 28  disease_55822004   68 non-null     int64
 29  disease_56097005   68 non-null     int64
 30  disease_56294008   68 non-null     int64
 31  disease_57406009   68 non-null     int64
 32  disease_59621000   68 non-null     int64
 33  disease_61582004   68 non-null     int64
 34  disease_63305008   68 non-null     int64
 35  disease_66344007   68 non-null     int64
 36  disease_66590003   68 non-null     int64
 37  disease_73430006   68 non-null     int64
 38  disease_76581006   68 non-null     int64
```

```
 39  disease_78275009   68 non-null     int64
 40  disease_78667006   68 non-null     int64
 41  disease_84757009   68 non-null     int64
 42  disease_86406008   68 non-null     int64
 43  disease_89765005   68 non-null     int64
 44  disease_93796005   68 non-null     int64
 45  disease_93974005   68 non-null     int64
 46  disease_111552007  68 non-null     int64
 47  disease_128192007  68 non-null     int64
 48  disease_129565002  68 non-null     int64
 49  disease_191611001  68 non-null     int64
 50  disease_193462001  68 non-null     int64
 51  disease_195967001  68 non-null     int64
 52  disease_197480006  68 non-null     int64
 53  disease_203082005  68 non-null     int64
 54  disease_235595009  68 non-null     int64
 55  disease_238136002  68 non-null     int64
 56  disease_239732001  68 non-null     int64
 57  disease_239873007  68 non-null     int64
 58  disease_266435005  68 non-null     int64
 59  disease_267432004  68 non-null     int64
 60  disease_275471001  68 non-null     int64
 61  disease_313436004  68 non-null     int64
 62  disease_367475009  68 non-null     int64
 63  disease_373621006  68 non-null     int64
 64  disease_396275006  68 non-null     int64
 65  disease_414916001  68 non-null     int64
 66  disease_428724006  68 non-null     int64
 67  disease_703938007  68 non-null     int64
dtypes: int64(66), object(2)
memory usage: 36.2+ KB
None
   person_id     ad_grouping  trial_number          earliest_ad_start  \
0    1035038  Desvenlafaxine             1  2019-03-17 05:00:00+00:00
1    1035038  Desvenlafaxine             2  2019-06-29 05:00:00+00:00
2    1035038  Desvenlafaxine             3  2019-10-23 05:00:00+00:00
3    1119935  Desvenlafaxine             1  2018-03-17 06:00:00+00:00
4    1119935  Desvenlafaxine             2  2018-08-04 06:00:00+00:00

   Remission  age 13-19  age 20-40  age 41-64  age 65-79  age 80-89  …  \
0          0          0          0          1          0          0  …
1          0          0          0          1          0          0  …
2          0          0          0          1          0          0  …
3          0          0          0          1          0          0  …
4          0          0          0          1          0          0  …

   disease_266435005  disease_267432004  disease_275471001  disease_313436004  \
0                  1                  0                  0                  1
```

```
1                   1                   0                   0                   1
2                   1                   0                   0                   1
3                   0                   0                   0                   1
4                   0                   0                   0                   1

   disease_367475009  disease_373621006  disease_396275006  disease_414916001  \
0                  0                  0                  0                  1
1                  0                  0                  0                  1
2                  0                  0                  0                  1
3                  0                  0                  0                  1
4                  0                  0                  0                  1

   disease_428724006  disease_703938007
0                  0                  0
1                  0                  0
2                  0                  0
3                  0                  0
4                  0                  0

[5 rows x 68 columns]


Antidepressant: Doxepin
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 439 entries, 0 to 438
Columns: 643 entries, person_id to disease_451041000124103
dtypes: int64(641), object(2)
memory usage: 2.2+ MB
None
   person_id ad_grouping  trial_number          earliest_ad_start  Remission  \
0    1001663     Doxepin             2  2016-08-25 06:00:00+00:00          0
1    1001663     Doxepin             3  2017-02-18 06:00:00+00:00          0
2    1004308     Doxepin             1  2010-06-28 00:42:00+00:00          0
3    1008400     Doxepin             1  2017-04-26 05:00:00+00:00          0
4    1010874     Doxepin             1  2019-04-26 18:04:00+00:00          0

   age 13-19  age 20-40  age 41-64  age 65-79  age 80-89  …  \
0          0          0          1          0          0  …
1          0          0          1          0          0  …
2          0          0          1          0          0  …
3          0          0          1          0          0  …
4          0          0          1          0          0  …

   disease_129161000119100  disease_132141000119106  disease_132551000119104  \
0                        0                        0                        0
1                        0                        0                        0
2                        0                        0                        0
3                        0                        0                        0
```

```
4                         0                         0                         0

   disease_132611000119104  disease_138911000119106  disease_147211000119101  \
0                         0                         0                         0
1                         0                         0                         0
2                         0                         0                         0
3                         0                         0                         0
4                         0                         0                         0

   disease_188061000119100  disease_293241000119100  disease_368051000119109  \
0                         0                         0                         1
1                         0                         0                         1
2                         0                         0                         0
3                         0                         0                         0
4                         0                         0                         0

   disease_451041000124103
0                         0
1                         0
2                         0
3                         0
4                         0

[5 rows x 643 columns]


Antidepressant: Duloxetine
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2590 entries, 0 to 2589
Columns: 2089 entries, person_id to disease_16837681000119104
dtypes: int64(2087), object(2)
memory usage: 41.3+ MB
None
   person_id ad_grouping  trial_number          earliest_ad_start  Remission  \
0    1000042  Duloxetine             1  2018-08-07 00:00:00+00:00          1
1    1000042  Duloxetine             2  2019-09-12 09:46:53+00:00          1
2    1000042  Duloxetine             3  2020-02-20 09:24:38+00:00          1
3    1000370  Duloxetine             1  2019-01-22 00:00:00+00:00          0
4    1004947  Duloxetine             1  2017-04-27 13:32:00+00:00          1

   age 13-19  age 20-40  age 41-64  age 65-79  age 80-89  …  \
0          0          0          1          0          0  …
1          0          0          1          0          0  …
2          0          0          1          0          0  …
3          0          0          1          0          0  …
4          0          0          0          1          0  …

   disease_10641391000119104  disease_10672271000119100  \
```

```
0                           0                           0
1                           0                           0
2                           0                           0
3                           0                           0
4                           0                           0

   disease_10676831000119101  disease_10685111000119102  \
0                           0                           0
1                           0                           0
2                           0                           0
3                           0                           0
4                           0                           0

   disease_10692761000119107  disease_12246311000119109  \
0                           0                           1
1                           0                           1
2                           0                           1
3                           0                           0
4                           0                           0

   disease_15960061000119102  disease_15960141000119102  \
0                           0                           0
1                           0                           0
2                           0                           0
3                           0                           0
4                           0                           0

   disease_16276361000119109  disease_16837681000119104
0                           0                           0
1                           0                           0
2                           0                           0
3                           0                           0
4                           0                           0

[5 rows x 2089 columns]


Antidepressant: Escitalopram
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2031 entries, 0 to 2030
Columns: 1582 entries, person_id to disease_16837681000119104
dtypes: int64(1580), object(2)
memory usage: 24.5+ MB
None
   person_id   ad_grouping  trial_number          earliest_ad_start  \
0    1000042  Escitalopram             1  2018-05-14 00:00:00+00:00
1    1000612  Escitalopram             1  2010-12-26 16:12:00+00:00
2    1001917  Escitalopram             1  2016-12-28 19:48:00+00:00
```

```
3    1002826  Escitalopram             1  2019-07-29 09:00:00+00:00
4    1003242  Escitalopram             1  2018-11-19 00:00:00+00:00

   Remission  age 13-19  age 20-40  age 41-64  age 65-79  age 80-89  …  \
0          1          0          0          1          0          0  …
1          1          0          0          1          0          0  …
2          1          0          1          0          0          0  …
3          0          0          1          0          0          0  …
4          0          0          0          1          0          0  …

   disease_10641391000119104  disease_10672271000119100  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_10685111000119102  disease_10743881000119104  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_11859701000119107  disease_15960061000119102  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_15960141000119102  disease_16237381000119100  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_16276361000119109  disease_16837681000119104
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

[5 rows x 1582 columns]
```

```
Antidepressant: Fluoxetine
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1940 entries, 0 to 1939
Columns: 1308 entries, person_id to disease_16837681000119104
dtypes: int64(1306), object(2)
memory usage: 19.4+ MB
None
   person_id ad_grouping  trial_number          earliest_ad_start  Remission  \
0    1000000  Fluoxetine             1  2012-04-15 19:44:00+00:00          1
1    1000039  Fluoxetine             1  2013-09-02 16:17:00+00:00          0
2    1004329  Fluoxetine             1  2015-08-31 00:00:00+00:00          1
3    1005788  Fluoxetine             1  2018-02-13 13:21:00+00:00          0
4    1005788  Fluoxetine             2  2019-03-05 15:21:00+00:00          0

   age 13-19  age 20-40  age 41-64  age 65-79  age 80-89  …  \
0          0          0          1          0          0  …
1          0          0          1          0          0  …
2          0          0          1          0          0  …
3          0          0          1          0          0  …
4          0          0          1          0          0  …

   disease_10633391000119104  disease_10633951000119108  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_10641391000119104  disease_10672271000119100  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          1                          0
4                          1                          0

   disease_10685111000119102  disease_10692761000119107  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_10743881000119104  disease_15960141000119102  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0
```

```
   disease_16002031000119102  disease_16837681000119104
0                          0                          0
1                          0                          0
2                          0                          0
3                          1                          0
4                          1                          0

[5 rows x 1308 columns]


Antidepressant: Mirtazapine
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1724 entries, 0 to 1723
Columns: 1615 entries, person_id to disease_16837681000119104
dtypes: int64(1613), object(2)
memory usage: 21.2+ MB
None
   person_id  ad_grouping  trial_number          earliest_ad_start  Remission  \
0    1003242  Mirtazapine             1  2015-01-23 00:00:00+00:00          1
1    1003242  Mirtazapine             2  2018-10-17 00:00:00+00:00          0
2    1007024  Mirtazapine             1  2012-09-23 05:00:00+00:00          0
3    1007024  Mirtazapine             2  2013-10-26 05:00:00+00:00          0
4    1007024  Mirtazapine             3  2014-06-20 05:00:00+00:00          0

   age 13-19  age 20-40  age 41-64  age 65-79  age 80-89  …  \
0          0          0          1          0          0  …
1          0          0          1          0          0  …
2          0          0          1          0          0  …
3          0          0          1          0          0  …
4          0          0          1          0          0  …

   disease_10633391000119104  disease_10633911000119107  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_10641391000119104  disease_10685111000119102  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_10743881000119104  disease_12246311000119109  \
0                          0                          0
```

```
1                           0                           0
2                           0                           0
3                           0                           0
4                           0                           0

   disease_15960061000119102  disease_15960141000119102  \
0                           0                           0
1                           0                           0
2                           0                           0
3                           0                           0
4                           0                           0

   disease_16237261000119100  disease_16837681000119104
0                           0                           0
1                           0                           0
2                           0                           0
3                           0                           0
4                           0                           0

[5 rows x 1615 columns]


Antidepressant: Nortriptyline
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 803 entries, 0 to 802
Columns: 1057 entries, person_id to disease_16039431000119105
dtypes: int64(1055), object(2)
memory usage: 6.5+ MB
None
   person_id    ad_grouping  trial_number          earliest_ad_start  \
0    1011705  Nortriptyline             1  2016-08-31 00:00:00+00:00
1    1014596  Nortriptyline             1  2006-12-28 00:00:00+00:00
2    1015518  Nortriptyline             1  2006-04-19 05:00:00+00:00
3    1017513  Nortriptyline             1  2012-01-08 20:20:00+00:00
4    1024170  Nortriptyline             1  2021-11-28 16:29:00+00:00

   Remission  age 13-19  age 20-40  age 41-64  age 65-79  age 80-89  …  \
0          0          0          0          1          0          0  …
1          1          0          1          0          0          0  …
2          0          0          0          1          0          0  …
3          1          0          0          1          0          0  …
4          0          0          0          1          0          0  …

   disease_368051000119109  disease_451041000124103  disease_1079821000119106  \
0                        0                        0                         0
1                        0                        0                         0
2                        0                        0                         0
3                        0                        0                         0
```

```
4                        1                         1                           0

   disease_1082601000119104  disease_10633351000119109  \
0                         0                          0
1                         0                          0
2                         0                          0
3                         0                          0
4                         0                          0

   disease_10633391000119104  disease_10641391000119104  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_10685111000119102  disease_10743881000119104  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_16039431000119105
0                          0
1                          0
2                          0
3                          0
4                          0

[5 rows x 1057 columns]


Antidepressant: Paroxetine
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 868 entries, 0 to 867
Columns: 698 entries, person_id to disease_15960061000119102
dtypes: int64(696), object(2)
memory usage: 4.6+ MB
None
   person_id ad_grouping  trial_number          earliest_ad_start  Remission  \
0    1009930  Paroxetine             1  2012-04-09 14:31:00+00:00          0
1    1009954  Paroxetine             1  2016-05-04 05:00:00+00:00          1
2    1009954  Paroxetine             2  2021-03-10 05:00:00+00:00          1
3    1010643  Paroxetine             1  2015-10-29 00:00:00+00:00          1
4    1012031  Paroxetine             1  2016-06-13 05:00:00+00:00          0

   age 13-19  age 20-40  age 41-64  age 65-79  age 80-89  …  \
```

```
0          0          0          1          0          0  …
1          0          0          1          0          0  …
2          0          0          1          0          0  …
3          1          0          0          0          0  …
4          0          0          1          0          0  …

   disease_147211000119101  disease_153941000119100  disease_293241000119100  \
0                        0                        0                        0
1                        0                        0                        0
2                        0                        0                        0
3                        0                        0                        0
4                        0                        0                        0

   disease_368051000119109  disease_451041000124103  disease_1036671000000106  \
0                        0                        0                         0
1                        0                        0                         0
2                        0                        0                         0
3                        0                        0                         0
4                        0                        0                         0

   disease_1092851000119103  disease_1092881000119105  \
0                         0                         0
1                         0                         0
2                         0                         0
3                         0                         0
4                         0                         0

   disease_10685111000119102  disease_15960061000119102
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

[5 rows x 698 columns]


Antidepressant: Sertraline
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3598 entries, 0 to 3597
Columns: 2022 entries, person_id to disease_16837681000119104
dtypes: int64(2020), object(2)
memory usage: 55.5+ MB
None
   person_id ad_grouping  trial_number          earliest_ad_start  Remission  \
0    1000151  Sertraline             1  2008-08-26 23:29:49+00:00          1
1    1000151  Sertraline             2  2009-12-07 07:42:00+00:00          1
2    1000151  Sertraline             3  2010-08-01 16:33:43+00:00          1
```

```
3    1000265  Sertraline             1  2011-05-10 05:00:00+00:00          0
4    1001917  Sertraline             1  2021-06-10 16:46:00+00:00          0

   age 13-19  age 20-40  age 41-64  age 65-79  age 80-89  …  \
0          0          0          1          0          0  …
1          0          0          1          0          0  …
2          0          0          1          0          0  …
3          0          0          1          0          0  …
4          0          0          1          0          0  …

   disease_10742471000119108  disease_10743881000119104  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_10750551000119100  disease_10756101000119107  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_12246311000119109  disease_15649991000119105  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_15960061000119102  disease_15960141000119102  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_16039431000119105  disease_16837681000119104
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

[5 rows x 2022 columns]
```

```
Antidepressant: Trazodone
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4464 entries, 0 to 4463
Columns: 2326 entries, person_id to disease_16837681000119104
dtypes: int64(2324), object(2)
memory usage: 79.2+ MB
None
   person_id ad_grouping  trial_number          earliest_ad_start  Remission  \
0    1000039   Trazodone             1  2013-08-31 02:28:00+00:00          0
1    1000042   Trazodone             1  2013-08-03 23:08:59+00:00          0
2    1000265   Trazodone             1  2011-05-10 05:00:00+00:00          0
3    1002739   Trazodone             1  2016-02-13 00:00:00+00:00          1
4    1003242   Trazodone             1  2014-02-25 00:00:00+00:00          0

   age 13-19  age 20-40  age 41-64  age 65-79  age 80-89  …  \
0          0          0          1          0          0  …
1          0          0          1          0          0  …
2          0          0          1          0          0  …
3          0          0          1          0          0  …
4          0          0          1          0          0  …

   disease_10633911000119107  disease_10633951000119108  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_10641391000119104  disease_10672271000119100  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_10685111000119102  disease_10743881000119104  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_12246311000119109  disease_15960061000119102  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0
```

```
   disease_15960141000119102  disease_16837681000119104
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

[5 rows x 2326 columns]


Antidepressant: Venlafaxine
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1448 entries, 0 to 1447
Columns: 1298 entries, person_id to disease_15960141000119102
dtypes: int64(1296), object(2)
memory usage: 14.3+ MB
None
   person_id  ad_grouping  trial_number          earliest_ad_start  Remission  \
0    1000151  Venlafaxine             1  2012-07-30 09:16:43+00:00          1
1    1003242  Venlafaxine             1  2014-02-25 00:00:00+00:00          0
2    1003242  Venlafaxine             2  2015-07-02 00:00:00+00:00          1
3    1003242  Venlafaxine             3  2016-02-01 00:00:00+00:00          1
4    1009990  Venlafaxine             1  2014-03-06 05:00:00+00:00          0

   age 13-19  age 20-40  age 41-64  age 65-79  age 80-89  …  \
0          0          0          1          0          0  …
1          0          0          1          0          0  …
2          0          0          1          0          0  …
3          0          0          1          0          0  …
4          0          0          1          0          0  …

   disease_10633351000119109  disease_10633391000119104  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_10637831000119101  disease_10641391000119104  \
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

   disease_10685111000119102  disease_10742471000119108  \
0                          0                          0
```

```
1                           0                           0
2                           0                           0
3                           0                           0
4                           0                           0

   disease_10743881000119104  disease_11055151000119108  \
0                           0                           0
1                           0                           0
2                           0                           0
3                           0                           0
4                           0                           0

   disease_12246311000119109  disease_15960141000119102
0                           0                           0
1                           0                           0
2                           0                           0
3                           0                           0
4                           0                           0

[5 rows x 1298 columns]


Antidepressant: Other
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 523 entries, 0 to 522
Columns: 793 entries, person_id to disease_15960141000119102
dtypes: int64(791), object(2)
memory usage: 3.2+ MB
None
   person_id ad_grouping  trial_number          earliest_ad_start  Remission  \
0    1004308       Other             1  2007-07-25 17:21:00+00:00          0
1    1008400       Other             1  2015-07-15 05:00:00+00:00          1
2    1009473       Other             1  2007-11-09 14:00:00+00:00          0
3    1013770       Other             1  2019-07-13 09:32:00+00:00          1
4    1014596       Other             1  2011-04-20 00:00:00+00:00          0

   age 13-19  age 20-40  age 41-64  age 65-79  age 80-89  …  \
0          0          0          1          0          0  …
1          0          0          1          0          0  …
2          0          1          0          0          0  …
3          0          0          1          0          0  …
4          0          0          1          0          0  …

   disease_147211000119101  disease_156051000119109  disease_293241000119100  \
0                        0                        0                        0
1                        0                        0                        0
2                        0                        0                        0
3                        0                        0                        0
```

```
4                         0                         0                         0

   disease_367991000119101  disease_368051000119109  disease_451041000124103  \
0                         0                         0                         0
1                         0                         0                         0
2                         0                         0                         0
3                         0                         0                         0
4                         0                         0                         0

   disease_861371000000102  disease_1079811000119104  \
0                         0                         0
1                         0                         0
2                         0                         0
3                         0                         0
4                         0                         0

   disease_10685111000119102  disease_15960141000119102
0                          0                          0
1                          0                          0
2                          0                          0
3                          0                          0
4                          0                          0

[5 rows x 793 columns]
```

4 Fit the Network Model

[ ]: