## MIMIC III SQL JOE RICE (JR76683)

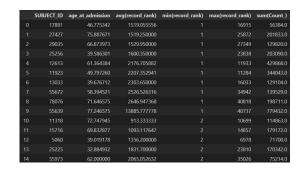


## **METHOD**

- MIMIC III data was loaded locally into Parquet files
- The duckdb Python package was used to query the Parquet files using SQL
  - Some syntax used is specific to this implementation (e.g. RegEx use)
- Queries will be in the presentation notes to preserve formatting and ensure enough room
- In many cases, we limit row counts for readability

### WHAT PATIENTS HAVE THE MOST RECORDS ACROSS ALL TABLES?

- In each table with a Subject ID, we count the records and rank by record count, where a rank of I means the most records in each table
- We union all these results together
- We then make aggregations from the unioned data, grouping by subject ID
- We filter by patients in excess of 30 years old
- We aggregate all of those results and sort by:
  - The minimum ranks observed across all tables, ascending
  - The average ranks across all tables, ascending
  - The maximum age of a subject at a hospital admission, descending



```
select
src.subject id,
datediff('day', subj.dob, cast(adm.admittime as date))/365 as age at admission,
avg(record rank),
min(record rank),
max(record_rank),
sum(Count )
from (
  select
  'Admissions' as Table,
  subject id,
  count(*) Count,
  row number() over (partition by null order by Count desc, subject id)
Record_Rank
  from 'ADMISSIONS*.parquet'
  group by 2
  union all
  select
```

```
'Callout' as Table,
  subject id,
  count(*) Count,
  row number() over (partition by null order by Count desc, subject id)
Record Rank
  from 'CALLOUT*.parquet'
  group by 2
  union all
  select
  'Chartevents' as Table,
  subject id,
  count(*) Count,
  row number() over (partition by null order by Count desc, subject id)
Record Rank
  from 'CHARTEVENTS*.parquet'
  group by 2
  union all
  select
  'Cptevents' as Table,
  subject id,
  count(*) Count,
  row number() over (partition by null order by Count desc, subject id)
Record Rank
  from 'CPTEVENTS*.parquet'
  group by 2
  union all
  select
  'Datetimeevents' as Table,
  subject id,
  count(*) Count,
  row number() over (partition by null order by Count desc, subject id)
Record Rank
  from 'DATETIMEEVENTS*.parquet'
  group by 2
  union all
  select
  'Diagnoses_icd' as Table,
  subject id,
  count(*) Count,
  row number() over (partition by null order by Count desc, subject id)
Record Rank
  from 'DIAGNOSES_ICD*.parquet'
```

```
group by 2
  union all
  select
  'Drgcodes' as Table,
  subject_id,
  count(*) Count,
  row number() over (partition by null order by Count desc, subject id)
Record Rank
  from 'DRGCODES*.parquet'
  group by 2
  union all
  select
  'Icustays' as Table,
  subject id,
  count(*) Count,
  row_number() over (partition by null order by Count desc, subject id)
Record Rank
  from 'ICUSTAYS*.parquet'
  group by 2
  union all
  select
  'Inputevents_cv' as Table,
  subject id,
  count(*) Count,
  row number() over (partition by null order by Count desc, subject id)
Record Rank
  from 'INPUTEVENTS CV*.parquet'
  group by 2
  union all
  select
  'Inputevents_mv' as Table,
  subject id,
  count(*) Count,
  row number() over (partition by null order by Count desc, subject id)
Record Rank
  from 'INPUTEVENTS MV*.parquet'
  group by 2
  union all
  select
  'Labevents' as Table,
  subject id,
  count(*) Count,
```

```
row number() over (partition by null order by Count desc, subject id)
Record Rank
  from 'LABEVENTS*.parquet'
  group by 2
  union all
  select
  'Microbiologyevents' as Table,
  subject id,
  count(*) Count,
  row number() over (partition by null order by Count desc, subject id)
Record Rank
  from 'MICROBIOLOGYEVENTS*.parquet'
  group by 2
  union all
  select
  'Noteevents' as Table,
  subject id,
  count(*) Count,
  row number() over (partition by null order by Count desc, subject id)
Record Rank
  from 'NOTEEVENTS*.parquet'
  group by 2
  union all
  select
  'Outputevents' as Table,
  subject id,
  count(*) Count,
  row number() over (partition by null order by Count desc, subject id)
Record Rank
  from 'OUTPUTEVENTS*.parquet'
  group by 2
  union all
  select
  'Patients' as Table,
  subject id,
  count(*) Count,
  row_number() over (partition by null order by Count_ desc, subject_id)
Record Rank
  from 'PATIENTS*.parquet'
  group by 2
  union all
  select
```

```
'Prescriptions' as Table,
  subject id,
  count(*) Count,
  row number() over (partition by null order by Count desc, subject id)
Record Rank
  from 'PRESCRIPTIONS*.parquet'
  group by 2
  union all
  select
  'Procedureevents mv' as Table,
  subject id,
  count(*) Count,
  row number() over (partition by null order by Count desc, subject id)
Record Rank
  from 'PROCEDUREEVENTS MV*.parquet'
  group by 2
  union all
  select
  'Procedures icd' as Table,
  subject id,
  count(*) Count,
  row number() over (partition by null order by Count desc, subject id)
Record Rank
  from 'PROCEDURES ICD*.parquet'
  group by 2
  union all
  select
  'Services' as Table,
  subject id,
  count(*) Count,
  row number() over (partition by null order by Count desc, subject id)
Record Rank
  from 'SERVICES*.parquet'
  group by 2
  union all
  select
  'Transfers' as Table,
  subject id,
  count(*) Count,
  row number() over (partition by null order by Count desc, subject id)
Record Rank
  from 'TRANSFERS*.parquet'
```

```
group by 2
) src
inner join 'PATIENTS*.parquet' subj
on src.subject_id = subj.subject_id
inner join (
select
subject_id,
max(admittime) as admittime
from 'ADMISSIONS*.parquet'
group by 1
) adm
on src.subject_id = adm.subject_id
where age_at_admission > 30
group by 1, 2
order by 4,3, 2
```

#### DO THEY HAVE ANY DISEASES IN COMMON?

- Using the top 3 results from the initial query, we now examine
- To do this, we use the DIAGNOSES\_ICD joined to D\_ICD\_DIAGNOSES and intersect results from 3 separate queries, one for each subject



```
select
diag.*
from 'DIAGNOSES_ICD*.parquet' subj_diag
inner join 'D ICD DIAGNOSES*.parquet' diag
on subj diag.ICD9 CODE = diag.ICD9 CODE
where subj diag.SUBJECT ID = 17891
intersect
select
diag.*
from 'DIAGNOSES_ICD*.parquet' subj_diag
inner join 'D ICD DIAGNOSES*.parquet' diag
on subj diag.ICD9 CODE = diag.ICD9 CODE
where subj diag.SUBJECT ID = 27427
intersect
select
diag.*
from 'DIAGNOSES ICD*.parquet' subj diag
```

inner join 'D\_ICD\_DIAGNOSES\*.parquet' diag
on subj\_diag.ICD9\_CODE = diag.ICD9\_CODE

where subj\_diag.SUBJECT\_ID = 29035
order by 1

### WHAT MEDICATIONS HAVE THEY HAD IN COMMON?

- Similar to the last step, we take 3 subqueries, one per subject using the PRESCRIPTIONS table
- However, we union the results and each subquery returns subject ID, drug, and number of admissions drug was used
- We aggregate the previous, filtering to drugs only used by all 3 using a HAVING clause
- We sort by descending admission counts and take the top 10 entries

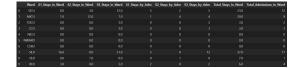


```
select
drug,
sum(count_) as usage
from (
    select
    subject id,
    drug,
    count(distinct hadm_id) as count_
    from 'PRESCRIPTIONS*.parquet'
    where SUBJECT ID = 17891
    group by 1,2
    select
    subject id,
    drug,
    count(distinct hadm_id) as count_
    from 'PRESCRIPTIONS*.parquet'
    where SUBJECT_ID = 27427
    group by 1,2
```

```
union
    select
    subject_id,
    drug,
    count(distinct hadm_id) as count_
    from 'PRESCRIPTIONS*.parquet'
    where SUBJECT_ID = 29035
    group by 1,2
) rx
group by 1
having count(distinct subject_id) = 3
order by 2 desc
limit 10
```

#### WHAT TYPES OF ICU ADMISSIONS HAVE THEY HAD?

- For each subject, we use the TRANSFERS to look at the number of ICU and Ward visits by subject, one column for number of ward stays and number of admissions involving that ward
- We group by the CURR\_CAREUNIT if it's available, otherwise the CURR WARDID
- We sort using a regular expression, where if the grouped by column has all text (i.e. is a care unit), we see those first, then followed by ward ID
  - Secondarily, we sort by descending total stays in each ward/care unit



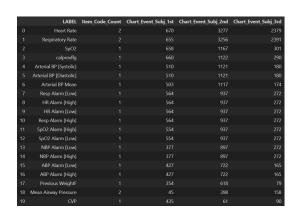
Query in notes below

#### select

coalesce(CURR CAREUNIT, cast(CURR WARDID as varchar(3))) as Ward, sum(case when subject id = 17891 then 1 else 0 end) as S1 Stays in Ward, sum(case when subject id = 27427 then 1 else 0 end) as S2\_Stays\_in\_Ward, sum(case when subject id = 29035 then 1 else 0 end) as S3 Stays in Ward, count(distinct Case when subject\_id = 17891 then hadm\_id end) as S1\_Stays\_by\_Adm, count(distinct Case when subject id = 27427 then hadm id end) as S2 Stays by Adm, count(distinct Case when subject id = 29035 then hadm id end) as S3 Stays by Adm, S1 Stays in Ward + S2 Stays in Ward + S3 Stays in Ward as Total Stays In Ward, S1\_Stays\_by\_Adm + S2\_Stays\_by\_Adm + S3\_Stays\_by\_Adm as Total Admissions In Ward from 'TRANSFERS\*.parquet' where WARD is not null group by 1 order by case when regexp\_matches(WARD, '^[A-Z]+\$') then 1 else 0 end desc, Total Stays In Ward desc

## WHAT CHARTING OCCURS WITH THESE SUBJECTS?

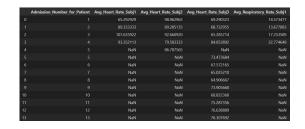
- Using the CHARTEVENTS table, we look at the number of chart events by TYPE
- We then look by each TYPE and for each subject ID, checking to see if the 3 subjects experienced that TYPE
- We filter out using a HAVING clause combined with a LEAST of the counts by subject ID, ensuring the least is greater than 0



```
select
--cv.ITEMID,
items.LABEL,
count(distinct cv.itemid) as Item Code Count,
count(distinct case when subject id = 17891 then charttime else null end)
Chart_Event_Subj_1st,
count(distinct case when subject_id = 27427 then charttime else null end)
Chart Event Subj 2nd,
count(distinct case when subject id = 29035 then charttime else null end)
Chart Event Subj 3rd
from 'CHARTEVENTS*.parquet' cv
inner join 'D ITEMS*.parquet' items
on cv.ITEMID = items.ITEMID
where subject id in (17891,27427,29035)
and cv.value is not null
group by 1
having least(Chart_Event_Subj_1st, Chart_Event_Subj_2nd, Chart_Event_Subj_3rd) >
order by chart event Subj 1st + chart event Subj 2nd + chart event Subj 3rd desc
```

## HOW DO THE VARIOUS CHART READINGS COMPARE TO EACH OTHER?

- We make 3 subqueries, one for each subject
- In those queries, we use ordered analytics and partitioning to identify the admission number any readings are associated with
- We then aggregate the top 6 readings from the previous, by subject and admission
- We do full outer joins by admission number; we do this since we want all data for all admissions and the admission numbers may be different for all
- Since the data are so wide, we only show one example of readings on the right



Query in notes below

#### select

coalesce(subj1.admission number, subj2.admission number, subj3.admission number) as Admission Number for Patient, Subj1.Avg Heart Rate as Avg Heart Rate Subj1, Subj2.Avg Heart Rate as Avg Heart Rate Subj2, Subj3.Avg\_Heart\_Rate as Avg\_Heart\_Rate\_Subj3, Subj1.Avg Respiratory Rate as Avg Respiratory Rate Subj1, Subj2.Avg Respiratory Rate as Avg Respiratory Rate Subj2, Subj3.Avg Respiratory Rate as Avg Respiratory Rate Subj3, Subj1.Avg SpO2 as Avg SpO2 Subj1, Subj2.Avg SpO2 as Avg SpO2 Subj2, Subj3.Avg SpO2 as Avg SpO2 Subj3, Subj1.Avg calprevflg as Avg calprevflg Subj1, Subj2.Avg\_calprevflg as Avg\_calprevflg\_Subj2, Subj3.Avg\_calprevflg as Avg\_calprevflg\_Subj3, Subj1.Avg Arterial BP Diastolic as Avg Arterial BP Diastolic Subj1, Subj2.Avg Arterial BP Diastolic as Avg Arterial BP Diastolic Subj2, Subj3.Avg Arterial BP Diastolic as Avg Arterial BP Diastolic Subj3,

```
Subj1.Avg Arterial BP Systolic as Avg Arterial BP Systolic Subj1,
Subj2.Avg Arterial BP Systolic as Avg Arterial BP Systolic Subj2,
Subj3.Avg Arterial BP Systolic as Avg Arterial BP Systolic Subj3,
from (
 select
 subject id,
 hadm id,
 min(charttime) as First Chart Event,
 avg(case when label = 'Heart Rate' then valuenum end) as Avg Heart Rate,
avg(case when label = 'Respiratory Rate' then valuenum end) as
Avg Respiratory Rate,
avg(case when label = 'SpO2' then valuenum end) as Avg SpO2,
avg(case when label = 'calprevflg' then valuenum end) as Avg calprevflg,
avg(case when label = 'Arterial BP [Diastolic]' then valuenum end) as
Avg Arterial BP Diastolic,
avg(case when label = 'Arterial BP [Systolic]' then valuenum end) as
Avg Arterial BP Systolic,
 Row Number() over (partition by subject id order by min(charttime)) as
Admission Number
 from 'CHARTEVENTS*.parquet' cv
 inner join 'D ITEMS*.parquet' items
  on cv.ITEMID = items.ITEMID
 where items.LABEL in ('Heart Rate', 'Respiratory Rate', 'SpO2', 'calprevflg', 'Arterial BP
[Diastolic]','Arterial BP [Systolic]')
 and subject id = 17891
 group by 1,2
) Subj1
full outer join (
 select
 subject id,
 hadm id,
 min(charttime) as First Chart Event,
 avg(case when label = 'Heart Rate' then valuenum end) as Avg Heart Rate,
avg(case when label = 'Respiratory Rate' then valuenum end) as
Avg Respiratory Rate,
avg(case when label = 'SpO2' then valuenum end) as Avg SpO2,
avg(case when label = 'calprevflg' then valuenum end) as Avg calprevflg,
avg(case when label = 'Arterial BP [Diastolic]' then valuenum end) as
Avg Arterial BP Diastolic,
avg(case when label = 'Arterial BP [Systolic]' then valuenum end) as
```

```
Avg Arterial BP Systolic,
 Row Number() over (partition by subject id order by min(charttime)) as
Admission Number
 from 'CHARTEVENTS*.parquet' cv
 inner join 'D ITEMS*.parquet' items
  on cv.ITEMID = items.ITEMID
 where items.LABEL in ('Heart Rate', 'Respiratory Rate', 'SpO2', 'calprevflg', 'Arterial BP
[Diastolic]','Arterial BP [Systolic]')
 and subject id = 27427
 group by 1,2
) Subj2
on Subj1.admission number = subj2.admission number
full outer join (
 select
 subject id,
 hadm id,
 min(charttime) as First Chart Event,
 avg(case when label = 'Heart Rate' then valuenum end) as Avg Heart Rate,
avg(case when label = 'Respiratory Rate' then valuenum end) as
Avg Respiratory Rate,
avg(case when label = 'SpO2' then valuenum end) as Avg SpO2,
avg(case when label = 'calprevflg' then valuenum end) as Avg calprevflg,
avg(case when label = 'Arterial BP [Diastolic]' then valuenum end) as
Avg Arterial BP Diastolic,
avg(case when label = 'Arterial BP [Systolic]' then valuenum end) as
Avg Arterial BP Systolic,
 Row Number() over (partition by subject id order by min(charttime)) as
Admission Number
 from 'CHARTEVENTS*.parquet' cv
 inner join 'D ITEMS*.parquet' items
  on cv.ITEMID = items.ITEMID
 where items.LABEL in ('Heart Rate', 'Respiratory Rate', 'SpO2', 'calprevflg', 'Arterial BP
[Diastolic]','Arterial BP [Systolic]')
 and subject id = 29035
 group by 1,2
) Subj3
on subj1.admission number = subj3.admission number
and subj2.admission number = subj3.admission number
order by 1
```

### HOW MANY OTHER PEOPLE HAVE THESE DIAGNOSES?

 Using the DIAGNOSES\_ICD table and D\_ICD\_DIAGNOSES, we find the applicable ICD9 codes for our 3 subjects and do a count distinct subject IDs for those codes

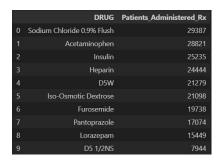


```
select
diag.*
from DIAGNOSES ICD subj diag
inner join D ICD DIAGNOSES diag
on subj_diag.ICD9_CODE = diag.ICD9_CODE
where subj_diag.SUBJECT_ID = 17891
intersect
select
diag.*
from DIAGNOSES_ICD subj_diag
inner join D_ICD_DIAGNOSES diag
on subj diag.ICD9 CODE = diag.ICD9 CODE
where subj_diag.SUBJECT_ID = 27427
intersect
select
diag.*
from DIAGNOSES_ICD subj_diag
```

```
inner join D_ICD_DIAGNOSES diag
on subj_diag.ICD9_CODE = diag.ICD9_CODE
where subj_diag.SUBJECT_ID = 29035
order by 1
```

### HOW MANY OTHER PEOPLE HAVE USED THE SAME MEDS?

Like the previous, but we instead look at drugs instead



```
select
drug,
count(distinct subject_id) as Patients_Administered_Rx
from 'PRESCRIPTIONS*.parquet'
where drug in (
select
  DRUG
  from (
select
drug,
sum(count_) as usage
from (
  select
  subject_id,
  drug,
  count(distinct hadm_id) as count_
  from 'PRESCRIPTIONS*.parquet'
```

```
where SUBJECT ID = 17891
  group by 1,2
  union
  select
  subject_id,
  drug,
  count(distinct hadm_id) as count_
  from 'PRESCRIPTIONS*.parquet'
  where SUBJECT ID = 27427
  group by 1,2
  union
  select
  subject_id,
  drug,
  count(distinct hadm_id) as count_
  from 'PRESCRIPTIONS*.parquet'
  where SUBJECT ID = 29035
  group by 1,2
) rx
group by 1
having count(distinct subject_id) = 3
order by 2 desc
limit 10
)
group by 1
order by 2 desc
limit 10
```

## HOW MANY TIMES HAVE THE SAME MEDS AND DISEASES OCCUR TOGETHER

- We join the DIAGNOSES\_ICD and DRUG table to get the combinations
- We then down select using applicable subqueries for each subject, using queries that we used previously
- We then do a count distinct of the subject IDs to see how often they show up together
- With so many combinations, it's not surprising the value is so high



```
select
count(distinct rx.subject_id)
from (
  select
  distinct
  subject_id,
  drug
  from 'PRESCRIPTIONS*.parquet'
) rx
inner join (
  select
  distinct
  subject id,
  icd9_code
  from 'DIAGNOSES_ICD*.parquet'
) diag
on rx.subject_id = diag.subject_id
where drug in (
```

```
select
  DRUG
  from (
select
drug
from 'PRESCRIPTIONS*.parquet'
where SUBJECT ID = 17891
intersect
select
drug
from 'PRESCRIPTIONS*.parquet'
where SUBJECT_ID = 27427
intersect
select
drug
from 'PRESCRIPTIONS*.parquet'
where SUBJECT ID = 29035
  )
)
and icd9_code in (
select
  ICD9 CODE
 from (
select
diag.*
from 'DIAGNOSES_ICD*.parquet' subj_diag
inner join 'D_ICD_DIAGNOSES*.parquet' diag
on subj_diag.ICD9_CODE = diag.ICD9_CODE
where subj diag.SUBJECT ID = 17891
intersect
select
diag.*
from 'DIAGNOSES_ICD*.parquet' subj_diag
inner join 'D_ICD_DIAGNOSES*.parquet' diag
on subj_diag.ICD9_CODE = diag.ICD9_CODE
where subj diag.SUBJECT ID = 27427
intersect
```

```
select
diag.*
from 'DIAGNOSES_ICD*.parquet' subj_diag
inner join 'D_ICD_DIAGNOSES*.parquet' diag
on subj_diag.ICD9_CODE = diag.ICD9_CODE
where subj_diag.SUBJECT_ID = 29035
order by 1

)
and rx.subject_id not in (17891,27427,29035)
```

## WHAT ARE THE RAREST COMBINATIONS MOST RECENTLY RECORDED?

- We count drug/ICD9 combinations from joining the DIAGNOSES\_ICD and DRUG table, joined by SUBJECT ID and ADMISSION ID
- We then sort by descending discharge date and ascending count
- It should be noted, the combinations do not necessarily mean a drug treat the diagnosis; just that they occur in the same hospital admission
- More analysis would have to required to get that type of information

```
| 2009. C008 | SHORT TITLE | C006, TITLE | C
```

```
select
subj diag.icd9 code,
diag.short title,
diag.long title,
rx.drug,
max(dischtime) as Discharge Date,
count(distinct subj_diag.subject_id) as Patients_Affected
from DIAGNOSES ICD subj diag
inner join D_ICD_DIAGNOSES diag
 on subj diag.ICD9 CODE = diag.ICD9 CODE
inner join PRESCRIPTIONS rx
 on subj diag.subject id = rx.subject id
 and subj diag.hadm id = rx.hadm id
inner join ADMISSIONS adm
 on adm.subject_id = subj_diag.subject id
 and adm.hadm id = subj diag.hadm id
--where subj_diag.subject_id in (3386, 2187, 3417)
group by 1,2,3,4
order by 5 desc, 6
```

# BONUS – WHAT'S THE MOST FREQUENT MED/DIAGNOSIS COMBINATION?

- We join the DIAGNOSES\_ICD and PRESCRIPTIONS table
- We count distinct admissions and patients observing that combination, but sorting by descending admission count
- Perhaps not surprising, many heart-related ailments are present



```
select
diag.icd9_code,
diag_text.short_title,
diag_text.long_title,
rx.drug,
count(distinct diag.hadm_id) as Admissions_with_Combo,
count(distinct diag.subject_id) as Patients_with_Combo
from DIAGNOSES_ICD diag
inner join D_ICD_DIAGNOSES diag_text
    on diag.icd9_code = diag_text.icd9_code
inner join PRESCRIPTIONS rx
    on diag.hadm_id = rx.hadm_id
group by 1,2,3,4
order by 5 desc
```