Emergency Room Visits Dashboard Documentation

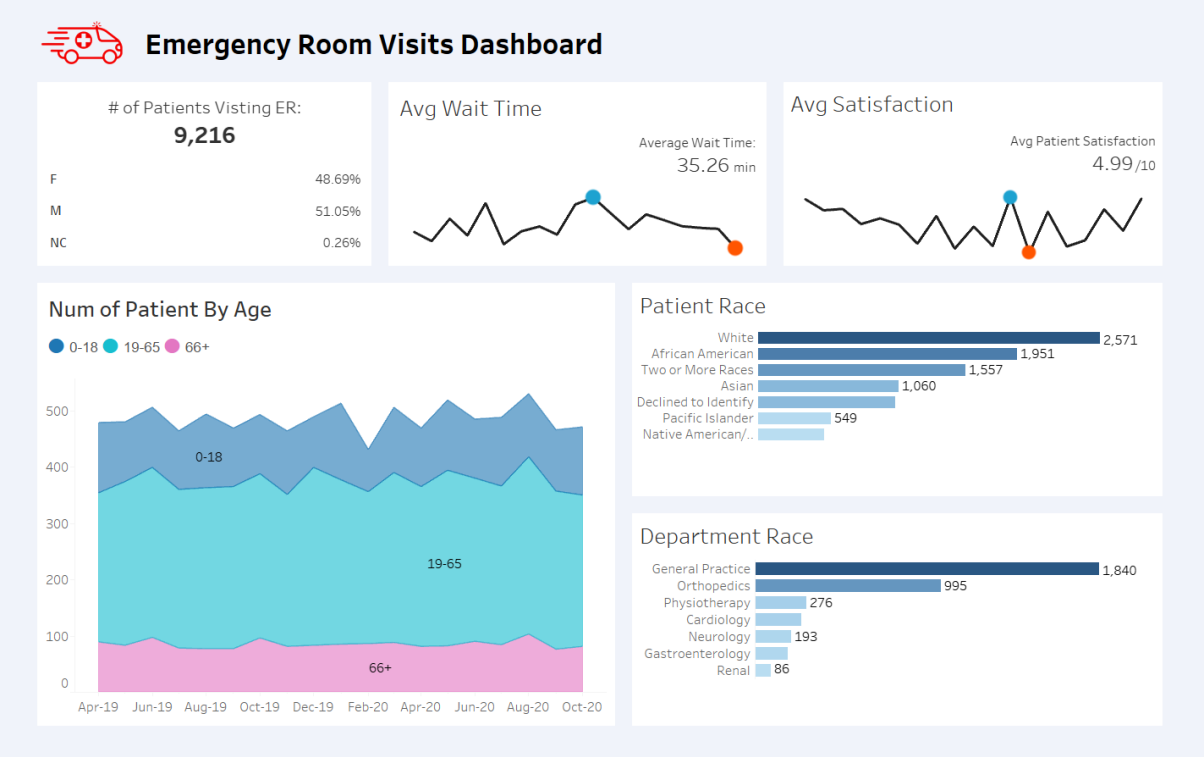
1.0 Introduction

Title: Emergency Room Dashboard

Purpose: To provide insights on wait times, ratings score and other parameters like race.

Audience: Hospital Managers, Hospital Higher Officials

2.0 Dashboard Overview



3.0 Data Sources

Origin: CSV File downloaded from data world

Format: CSV

4.0 Data Fields and Calculations

Dimensions:

1. Date
2. Department
3. Patient Gender
4. Patient Id
5. Patient Last Name
6. Patient First Initial
7. Patient Race
8. Age Grouping (calculated field)

Measures:

1. Patient Age
2. Patient Stats Score
3. Patient Wait Time
4. Min/Max Sat Score (calculated field)
5. Min/Max Wait Time (calculated field)

Calculated Fields:

1. Age Grouping

   For calculating Age grouping based of requirement.

```
Age Groupings

IF [Patient Age] >= 0 AND [Patient Age] <= 18 THEN '0-18'
ELSEIF [Patient Age] >= 19 AND [Patient Age] <= 65 THEN '19-65'
ELSEIF [Patient Age] >= 66  THEN '66+'
END
```

2. Min/Max Sat Score

   For Calculating Min and Max point in line chart using dual axis for sat score i.e. satisfaction score.

```
Min/Max Sat Score

Results are computed along Table (across).
IF AVG([Patient Sat Score]) = WINDOW_MIN(AVG([Patient Sat Score]))
THEN AVG([Patient Sat Score])
ELSEIF AVG([Patient Sat Score]) = WINDOW_MAX(AVG([Patient Sat Score]))
THEN AVG([Patient Sat Score])
END
```

3. Min/Max Wait Time

   For calculating Min and Max points on line chart for wait time using dual axis to indicate using circles.

```
Min/Max Wait Time

Results are computed along Table (across).
IF AVG([Patient Waittime]) = WINDOW_MIN(AVG([Patient Waittime]))
THEN AVG([Patient Waittime])
ELSEIF AVG([Patient Waittime]) = WINDOW_MAX(AVG([Patient Waittime]))
THEN AVG([Patient Waittime])
END
```

5.0 Filters and Actions

In this dashboard there are no filters yet.

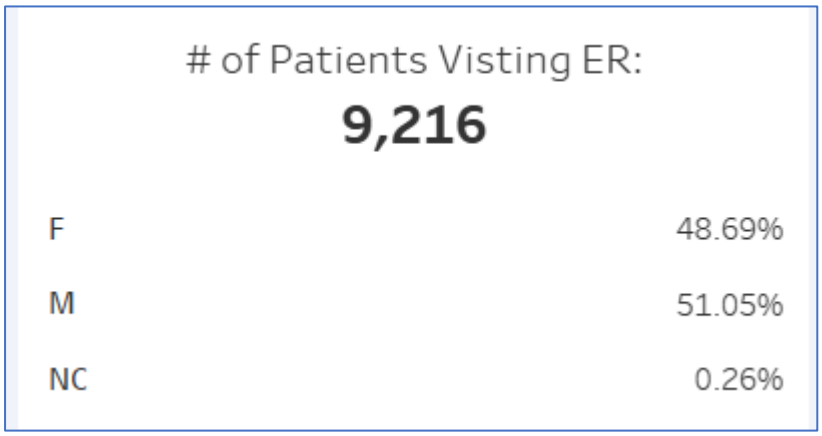
Uses can filter clicking on Visualization:

1. Gender
2. Patient Race
3. Department
4. Line Chart i.e. (month-year) combination

6.0 Visualizations

1. KPIs

a. Breakdown of Total Patients by Gender

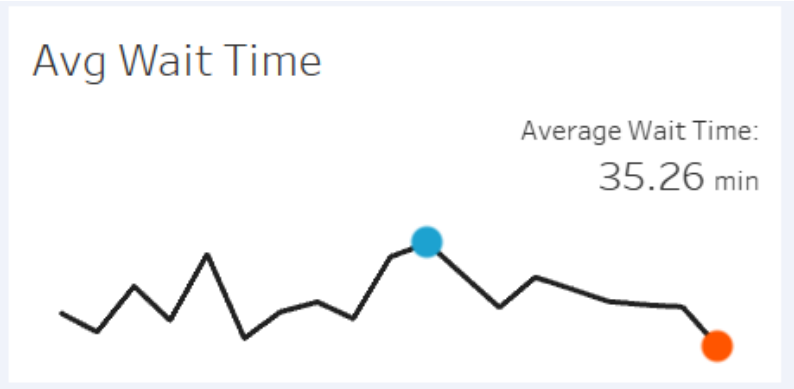


SQL Validation for this:

```
WITH total_patient AS
(
  select count(*) as total_patient
  from hospital_emergencey
),
gender_patient_grp as
(
  select patient_gender,
    count(*) as gender_count
  from hospital_emergencey
  group by patient_gender
)

SELECT
  gp.patient_gender,
  gp.gender_count,
  tp.total_patient,
  ROUND((CAST(gp.gender_count AS float) / tp.total_patient) * 100,2) as percentage_total
FROM total_patient as tp
CROSS JOIN gender_patient_grp as gp
```

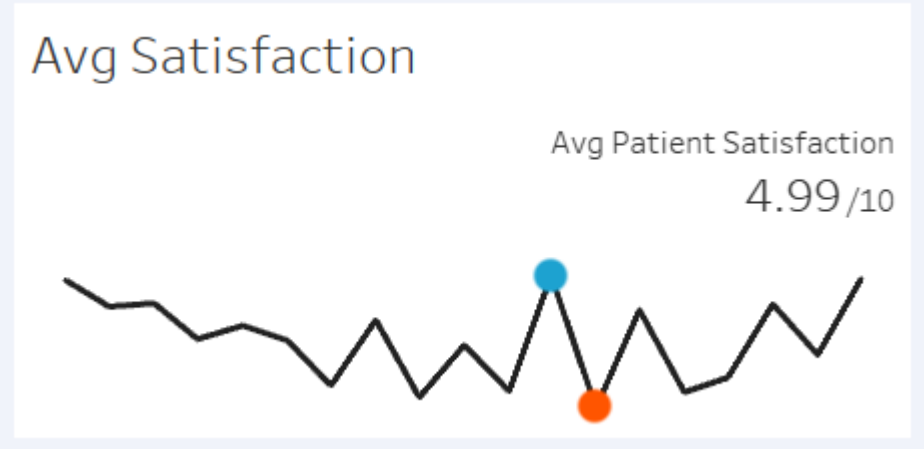
b. Average Waiting Time with Line chart ranging all dates available.



SQL Validation:

```
select
MONTH(date) as month,
YEAR(date) as year,
round(AVG(cast(patient_waittime as float)),2) as avg_time
from hospital_emergencey
group by MONTH(date), YEAR(date)
order by 1,2 desc
```

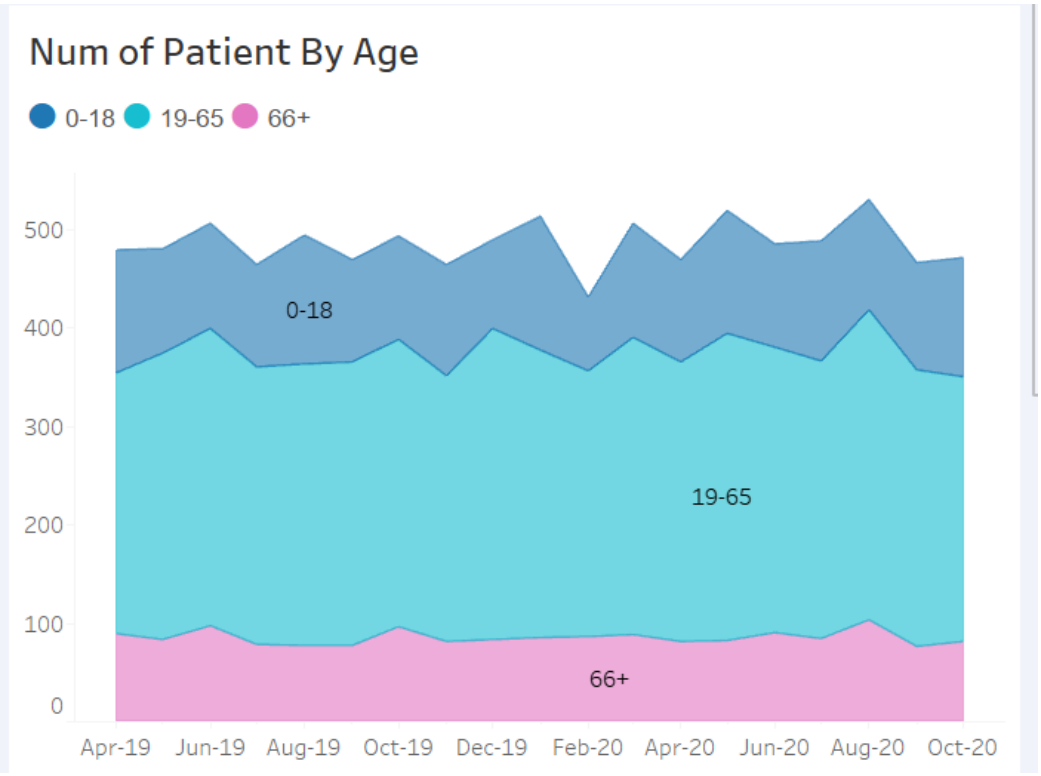
c. Average Satisfaction Score with Line chart ranging all dates available.



SQL Validation

```
select
MONTH(date) as month,
YEAR(date) as year,
round(AVG(cast(patient_sat_score as float)),2) as avg_time
from hospital_emergencey
group by MONTH(date), YEAR(date)
order by 1,2 desc
```

2. Number of Patient by Age Groups



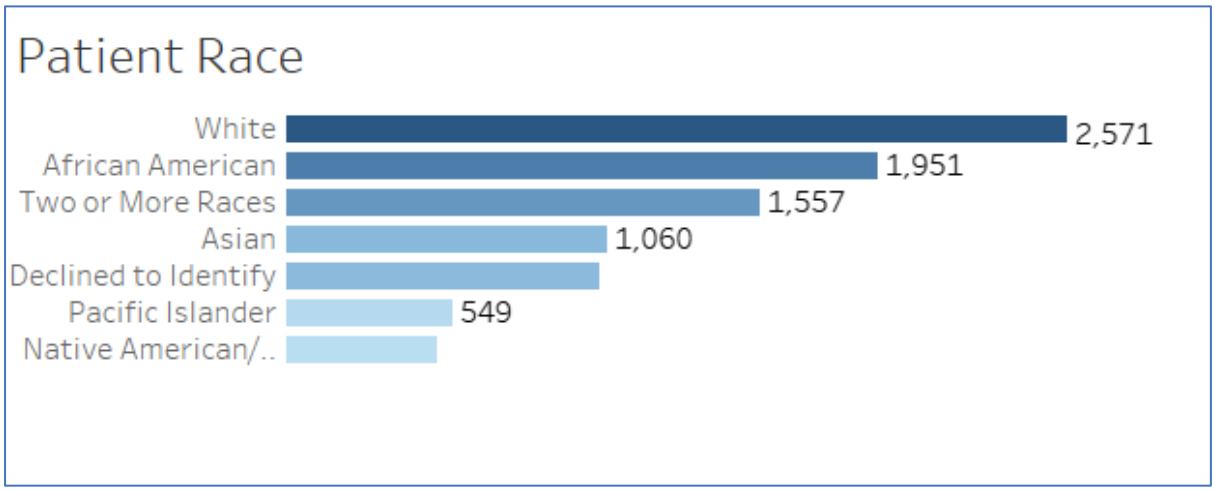
SQL Validation for Area Chart

```sql
WITH age_group AS
(
select
  patient_age,
  patient_id,
  CASE
    WHEN patient_age >=0 AND patient_age <=18 THEN '0-18'
    WHEN patient_age >=19 AND patient_age <= 65 THEN '18-65'
    WHEN patient_age >= 66 THEN '66+'
  END as age_group,
  date

FROM hospital_emergencey
)

select
  MONTH(date) as month,
  YEAR(date) as year,
  age_group,
  count(patient_id) as patient
FROM age_group
GROUP BY MONTH(date), YEAR(date), age_group
order by 1,2 desc
```

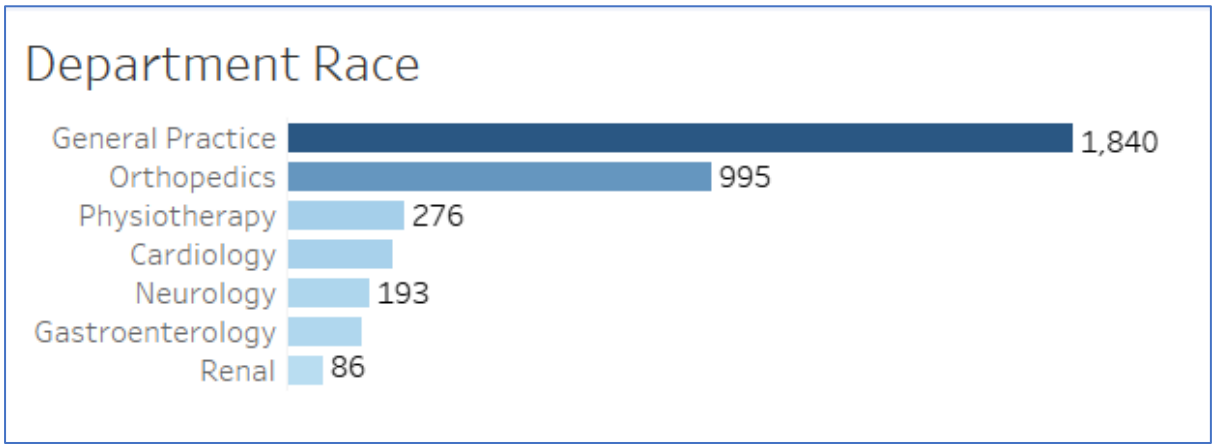
3. Number of Patient-by-Patient Race



SQL Validation

```sql
select patient_race,
  count(patient_id) as cnt
from hospital_emergencey
group by patient_race
order by cnt desc
```

4. Number of Patient-by-Department



SQL Validation for Department Chart

```
select department_referral,
  count(patient_id) as cnt
from hospital_emergencey
group by department_referral
having department_referral <> 'None'
order by cnt desc
```