03_EDA

June 8, 2025

1 EDA

1.1 EDA Setup: Visualization Style and Histogram Function

The first block of the Exploratory Data Analysis chapter defines essential configurations for reproducible and aesthetically consistent plotting. It establishes standardized paths for accessing processed data (`EXPORT_PATH`) and for saving visualization assets (`ASSETS_PATH`), following the principle of separation between raw computation and derived outputs.

The plotting environment is configured with `seaborn`'s `"whitegrid"` style, which facilitates readability in scientific plots. `matplotlib`'s global figure size is adjusted to ensure uniform layout across different visualizations.

Custom Histogram Plot Function: The function `plot_histogram()` is designed to produce high-quality histograms enriched with kernel density estimation (`kde`) by default. It allows for flexible customization of:

- **Binning** (`bins`)
- **Labels and titles**
- **Figure size**
- **Automatic file saving** (controlled by the `save_path` argument)

This modular utility function enhances code reusability and encourages consistent formatting throughout the EDA chapter, which is particularly valuable in a thesis-level project that emphasizes clarity and visual insight.

[ ]:
```
EXPORT_PATH = "../data/processed/"
ASSETS_PATH = "../assets/plots/eda/"

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import os

# === Plot Style ===
sns.set(style="whitegrid")
plt.rcParams["figure.figsize"] = (10, 6)
def plot_histogram(
    data, column, bins=30, kde=True, figsize=(10, 4),
```

```
    title=None, xlabel=None, ylabel="Number of Patients",
    save_path=None
):
    plt.figure(figsize=figsize)
    sns.histplot(data[column], bins=bins, kde=kde)
    plt.title(title if title else f"{column} Distribution")
    plt.xlabel(xlabel if xlabel else column)
    plt.ylabel(ylabel)
    plt.tight_layout()
    if save_path:
        plt.savefig(save_path)
    plt.show()
```

1.1.1 Dataset Loading and Structural Sanity Check

Before any statistical or visual exploration can be performed, the dataset `df_final_static.csv` is reloaded from disk to ensure isolation between the data preparation and EDA stages. This practice enhances modularity, reproducibility, and minimizes memory footprint across different execution environments (e.g., Jupyter kernels, pipelines).

The following diagnostics are executed to validate the structure and integrity of the dataset:

- `df_final.shape`: Displays the overall dimensions of the dataset, serving as a sanity check that no rows were inadvertently filtered or added since export.
- `df_final.head()`: A visual preview of the first few rows, useful for verifying column types, expected values, and possible categorical encodings.
- `df_final.columns`: Lists all column names, offering a quick overview of the available features for downstream analysis.
- `df_final.isnull().sum().sort_values(ascending=False)/len(df_final)`: Computes the proportion of missing values for each column. This step is essential for evaluating data quality and for guiding imputation strategies or exclusion decisions.

These steps collectively ensure that the dataset is in a clean and analyzable state, aligning with rigorous scientific standards for empirical research.

[ ]:
```
# === Load dataset ===
df_final = pd.read_csv(os.path.join(EXPORT_PATH, "df_final_static.csv"))

# === Confirm structure ===
print(df_final.shape)
df_final.head()
df_final.columns
df_final.isnull().sum().sort_values(ascending=False)/len(df_final)
```

```
(3685, 16)
```

[ ]:
```
SUBJECT_ID              0.0
HADM_ID                 0.0
ICUSTAY_ID              0.0
```

```
AGE                     0.0
GENDER                  0.0
ADMISSION_TYPE          0.0
ADMISSION_LOCATION      0.0
INSURANCE               0.0
FIRST_CAREUNIT          0.0
LOS                     0.0
HOSPITAL_EXPIRE_FLAG    0.0
INTIME_HOUR             0.0
INTIME_WEEKDAY          0.0
ADMITTIME_HOUR          0.0
ADMITTIME_WEEKDAY       0.0
INTIME                  0.0
dtype: float64
```

1.1.2 Descriptive Summary of Numeric Variables

This step provides a statistical snapshot of the numeric features within the dataset through the `describe().T` method, which transposes the default output to a column-wise orientation for enhanced readability.

For each numeric variable, the following summary statistics are computed:

- **Count**: Number of non-missing entries
- **Mean** and **Standard Deviation**: Indicators of central tendency and dispersion
- **Min, 25th, 50th (Median), 75th, and Max**: Useful for detecting skewness, spread, and potential outliers

This profiling phase is particularly valuable in clinical datasets like MIMIC-III, where variables such as age or ICU Length of Stay (LOS) often display right-skewed distributions, long tails, or discretized value spikes due to hospital policies (e.g., fixed discharge times).

By scanning these metrics, one can anticipate the need for transformations (e.g., log-scaling for LOS), outlier mitigation, and scaling adjustments in downstream modeling.

[ ]:
```
print("\n[INFO] Summary statistics for numeric variables:")
display(df_final.describe().T)
```

```
[INFO] Summary statistics for numeric variables:
                       count           mean           std          min  \
SUBJECT_ID            3685.0   38042.643691  29519.241245       3.0000
HADM_ID               3685.0  149043.439077  29176.674824  100074.0000
ICUSTAY_ID            3685.0  250221.804885  28861.797019  200003.0000
AGE                   3685.0      68.258887     15.991439       0.0000
LOS                   3685.0       5.744356      7.677370       0.0079
HOSPITAL_EXPIRE_FLAG  3685.0       0.287110      0.452475       0.0000
INTIME_HOUR           3685.0      13.721574      7.062893       0.0000
INTIME_WEEKDAY        3685.0       3.023338      1.988703       0.0000
```

```
ADMITTIME_HOUR        3685.0      13.995387      7.084968        0.0000
ADMITTIME_WEEKDAY     3685.0       3.016554      2.011907        0.0000

                             25%          50%          75%           max
SUBJECT_ID             13934.0000   27748.0000   62871.000    99985.0000
HADM_ID               123675.0000  148651.0000  175213.000   199943.0000
ICUSTAY_ID            225602.0000  250364.0000  275615.000   299950.0000
AGE                       58.0000      70.0000      81.000       91.0000
LOS                        1.7219       3.0194       6.602       97.2972
HOSPITAL_EXPIRE_FLAG       0.0000       0.0000       1.000        1.0000
INTIME_HOUR                8.0000      16.0000      20.000       23.0000
INTIME_WEEKDAY             1.0000       3.0000       5.000        6.0000
ADMITTIME_HOUR             9.0000      16.0000      20.000       23.0000
ADMITTIME_WEEKDAY          1.0000       3.0000       5.000        6.0000
```

1.2 Analysis of feature distributions

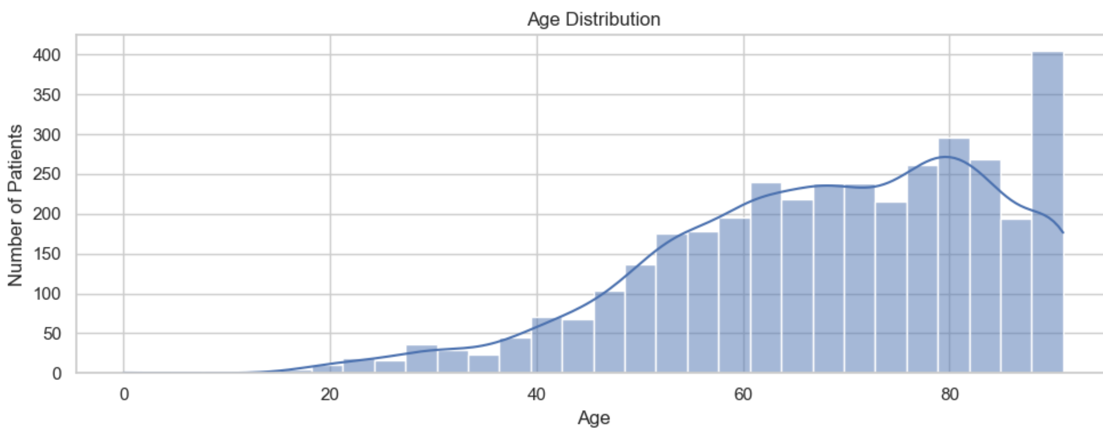
1.2.1 Age Distribution in ICU Sepsis Cohort

The histogram above illustrates the age distribution of patients admitted to the ICU with a diagnosis of sepsis. The distribution is right-skewed, with the majority of patients concentrated between the ages of 60 and 90. A significant spike is observed at age 91, which corresponds to the upper censoring limit imposed by the MIMIC-III dataset to preserve patient anonymity for individuals aged 89 and above.

This pattern is consistent with clinical expectations: elderly patients are more vulnerable to severe sepsis and are more frequently admitted to intensive care. The presence of a density tail in the lower age brackets (under 40) indicates that younger patients are present but far less frequent, likely representing cases of acute or atypical infections.

The distribution supports the decision to include **age as a primary predictor** in modeling ICU Length of Stay (LOS), as both biological resilience and comorbidity burden are age-dependent. Additionally, the sharp censoring at 91 must be taken into account to avoid bias or misinterpretation in models that assume a continuous age range.

[ ]:
```
plot_histogram(
    data=df_final,
    column="AGE",
    bins=30,
    title="Age Distribution",
    xlabel="Age",
    save_path=ASSETS_PATH + "age_distribution.png"
)
```



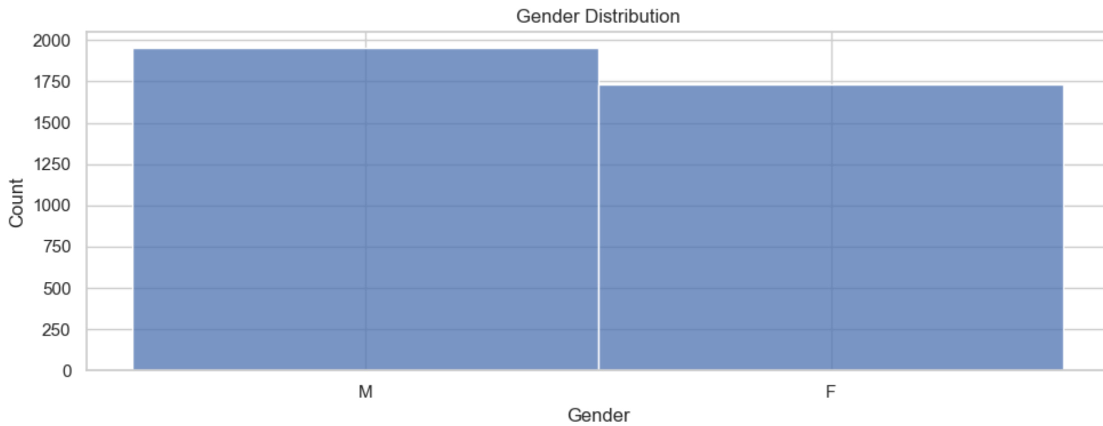
1.2.2 Gender Distribution in the ICU Sepsis Cohort

The bar chart illustrates the distribution of biological sex among ICU patients diagnosed with sepsis. The population consists of a slightly higher number of males (`M`) compared to females (`F`), with approximately 1950 male patients versus 1750 females.

This modest male predominance aligns with clinical literature suggesting that males are more frequently affected by sepsis, potentially due to differences in immune response, comorbidities, and healthcare access. However, the distribution remains reasonably balanced, implying that gender-specific bias is unlikely to be a major concern in the downstream modeling process.

It is important to note that while gender may not have a strong predictive signal on its own, it can interact with other variables (e.g., age, admission type, comorbidities) in non-linear ways. As such, it remains a useful covariate to retain in the model, especially when exploring explainability or fairness.

[ ]:
```
plot_histogram(
    data=df_final,
    column="GENDER",
    bins=len(df_final["GENDER"].unique()),
    kde=False,
    title="Gender Distribution",
    xlabel="Gender",
    ylabel="Count",
    save_path=ASSETS_PATH + "gender_distribution.png"
)
```



1.2.3 Distribution of ICU Length of Stay (LOS)

The histogram visualizes the empirical distribution of ICU Length of Stay (LOS), measured in days, for patients diagnosed with sepsis. As anticipated in clinical datasets, the distribution is **heavily right-skewed**, with the majority of stays concentrated in the 0–10 day range.

The tail extends considerably, with some extreme cases reaching up to ~100 days. Quantile statistics confirm this long-tail behavior:

- The **90th percentile** is at approximately **13.3 days**, indicating that 90% of patients are discharged within two weeks.
- The **99th percentile** is at **31.8 days**, suggesting that extreme long stays are rare but present.
- A total of **41 outliers** have a LOS exceeding **60 days**, representing clinically exceptional cases that may reflect complications, comorbidities, or institutional delays.

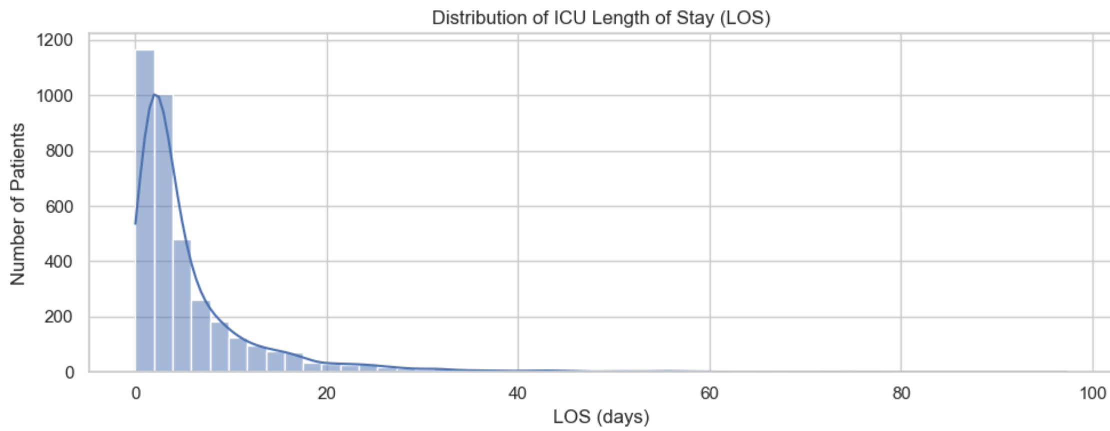
This pronounced asymmetry suggests that **log transformation** of LOS may be beneficial to stabilize variance and improve model performance. Furthermore, the presence of extreme outliers necessitates careful validation and may call for **robust modeling techniques** or outlier handling strategies during training.

[ ]:
```
plot_histogram(
    data=df_final,
    column="LOS",
    bins=50,
    title="Distribution of ICU Length of Stay (LOS)",
    xlabel="LOS (days)",
    save_path=ASSETS_PATH + "los_distribution.png"
)

q90 = df_final['LOS'].quantile(0.90)
q99 = df_final['LOS'].quantile(0.99)
max_los = df_final['LOS'].max()

print(f'Outliers: {df_final[(df_final.LOS>60)].shape[0]}')
```

```
print(f"90° percentile: {q90:.2f} days")
print(f"99° percentile: {q99:.2f} days")
print(f"Max LOS: {max_los:.2f} days")
```




```
Outliers: 7
90° percentile: 13.71 days
99° percentile: 37.41 days
Max LOS: 97.30 days
```

[ ]:
```
# Remove outliers
df_final = df_final[df_final['LOS'] <= 60]
```

[ ]:
```
# Install needed packages
!apt-get install texlive texlive-xetex texlive-latex-extra pandoc &> /dev/null
!pip install pypandoc &> /dev/null

# Mount your google drive to get access to your ipynb files

from google.colab import drive
drive.mount('/content/drive')
# and copy your notebook to this colab machine. Note that I am using *MY*␣
↪notebook filename

!cp "/content/drive/MyDrive/Colab Notebooks/03_EDA.ipynb" ./ &> /dev/null

# Then you can run the converter.

!jupyter nbconvert --to PDF "03_EDA.ipynb" &> /dev/null
```