

# Nurse staffing strategies for enhanced patient care

Matthew Bain

2024-03-22

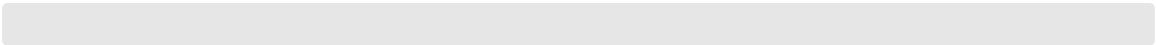
**abstract** I analyze a medical staffing dataset and identify avenues to improve work satisfaction among nurses and the quality of care provided at United States medical institutions.

## Table of contents

- Imports ..... 1
- The dataset ..... 2
  - Load the data ..... 2
  - Inspect the data ..... 2
  - Group the features ..... 4
  - Clean the data ..... 6
- Explore the dataset ..... 6
  - Visualize distributions ..... 6
  - Visualize relationships ..... 6
  - Compare groups ..... 7
- Feature engineer ..... 7
  - Join geographical data ..... 7
  - Join seasonal data ..... 7
- Analyze geography ..... 7
- Analyze seasonality ..... 7
- Model ..... 8
- Extra visualizations ..... 8
  - Sparklines ..... 8
- Archive ..... 8
- Bibliography ..... 8

[...]

## Imports



```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

from great_tables import GT
from pandas.plotting import scatter_matrix

from src.stylesheet import customize_plots
from src.inspection import make_df, display, display2
```

## The dataset

### Load the data

We begin by exploring the data to get to know the features and patterns on which we will base our analysis.

```
if 'data' not in locals():
    data = pd.read_csv(
        "../data/raw/PBJ_Daily_Nurse_Staffing_Q1_2024.zip",
        encoding='ISO-8859-1',
        low_memory=False
    )
else:
    print("data loaded.")
```

### Inspect the data

```
GT(data.sample(10))
```

[illegible]

```
df = data.describe().round(1)
# display(Markdown(df.to_markdown()))
GT(df.reset_index())
```

[illegible]

## Group the features

We note that there are 91 records per provider (`len(data["WorkDate"].unique())`) and 1,330,966 records in the table overall. The following table, which collapses the raw data across providers, thus has 14,626  $\left(\frac{1330966}{91}\right)$  entries.

```
df = data.loc[:, [
    "STATE",
    "COUNTY_NAME", "COUNTY_FIPS",
    "CITY",
    "PROVNAME", "PROVNUM",
    # "MDScensus"
]].value_counts()
# df.to_frame()
GT(df.reset_index().head(n=20))
```

STATE	COUNTY_- NAME	COUNTY_- FIPS	CITY	PROV- NAME	PROVNUM	count
AK	Anchorage	20	ANCHOR- AGE	PRESTIGE CARE & REHAB CENTER OF AN- CHORAGE	025025	91
OH	Allen	3	LIMA	LIMA CONVA- LESCENT HOME	366297	91
OH	Allen	3	LIMA	SHAWNEE MANOR	365361	91
OH	Allen	3	LIMA	SPRINGS OF LIMA THE	366464	91
OH	Allen	3	LIMA	SPRINGVIEW MANOR	366221	91
OH	Allen	3	SPENCERVILLE	ROSELAWN MANOR	365744	91
OH	Ashland	5	ASHLAND	BRETHREN CARE VILLAGE HEALTH CARE CENTER	366166	91
OH	Ashland	5	ASHLAND	CRYS- TAL CARE CENTER OF ASH- LAND	366239	91
OH	Ashland	5	ASHLAND	GOOD SARPE ANNEX UNDER THE STORY	365093	91
OH	Ashland	5	ASHLAND	SHAVE BROWN BROWN	365646	91
OH	Ashtabula	7	Ashtabula	Ashtabula Nursing Center	365000	91

```
GT(data[["CY_Qtr", "WorkDate", "MDScensus"]].head())
```

CY_Qtr	WorkDate	MDScensus
2024Q1	20240101	50
2024Q1	20240102	49
2024Q1	20240103	49
2024Q1	20240104	50
2024Q1	20240105	51

**Clean the data**

**Explore the dataset**

**Visualize distributions**

**Visualize relationships**

```
attributes = ["Hrs_RN", "Hrs_LPN_ctr", "Hrs_CNA", "Hrs_NAtrn", "Hrs_MedAide"]
n = len(attributes)

fig, axs = plt.subplots(n, n, figsize=(8, 8))
scatter_matrix(
    data[attributes].sample(200),
    ax=axs, alpha=.7,
    hist_kwds=dict(bins=15, linewidth=0)
)
fig.align_ylabels(axs[:, 0])
fig.align_xlabels(axs[-1, :])
for ax in axs.flatten():
    ax.tick_params(axis='both', which='both', length=3.5)

# save_fig("scatter_matrix_plot")

plt.show()
```

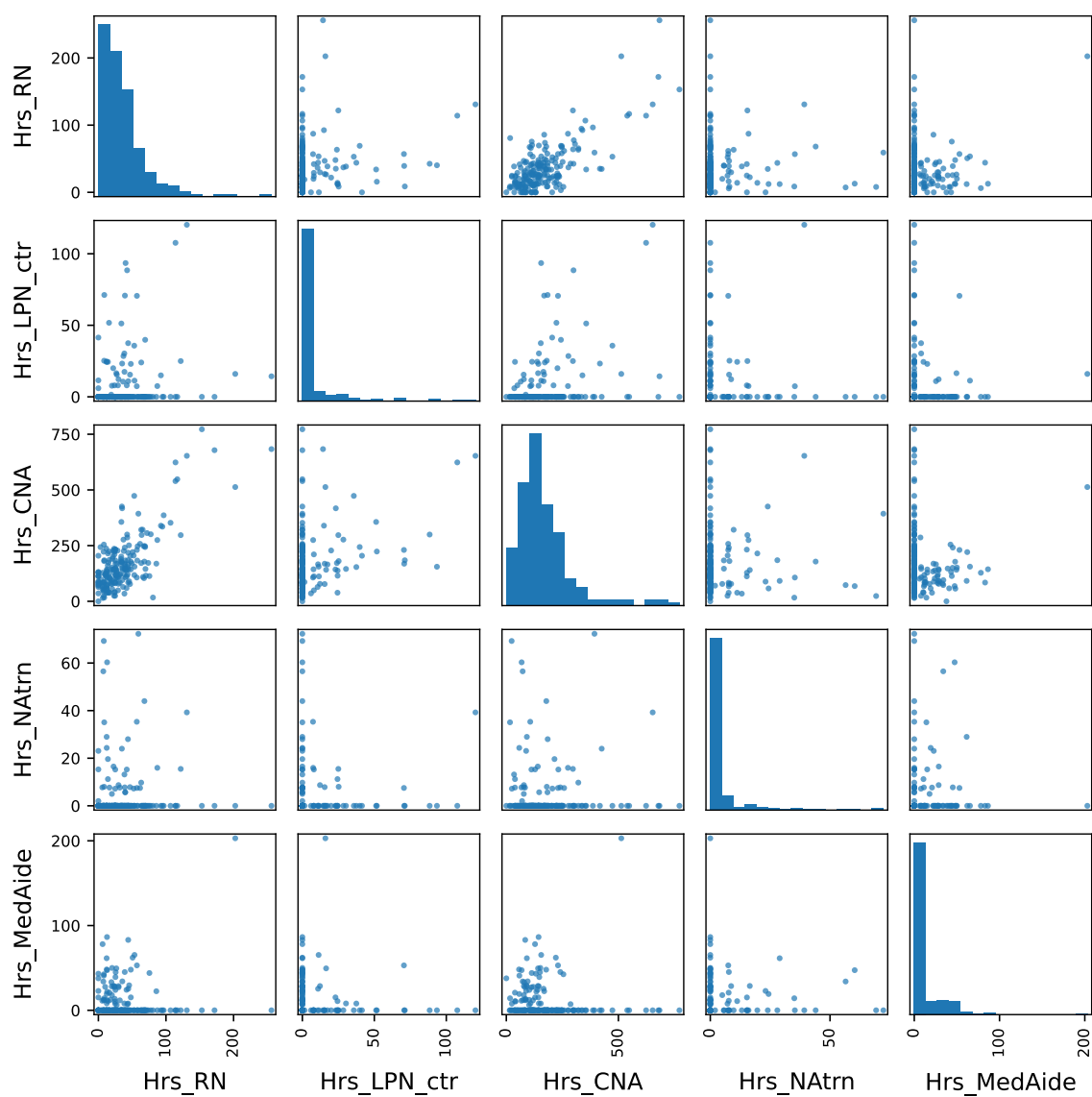


Figure 1: Scatter matrix of nursing worker working hours

**Compare groups**

**Feature engineer**

**Join geographical data**

**Join seasonal data**

**Analyze geography**

**Analyze seasonality**

Model

Extra visualizations

Sparklines

```
# Plot sparklines of average work hours across 91 days by state
(
  GT(gt_df.head(), rowname_col="STATE")
  .fmt_nanoplot(
    columns="lines",
    reference_line="mean",
    reference_area=["min", "q1"]
  )
  .tab_header(
    title="Nurse hours worked in the United States",
    subtitle="The top 5 busiest states",
  )
  .tab_stubhead(label="State")
  .cols_label(
    lines="Total hours worked over 91 days",
  )
)
```

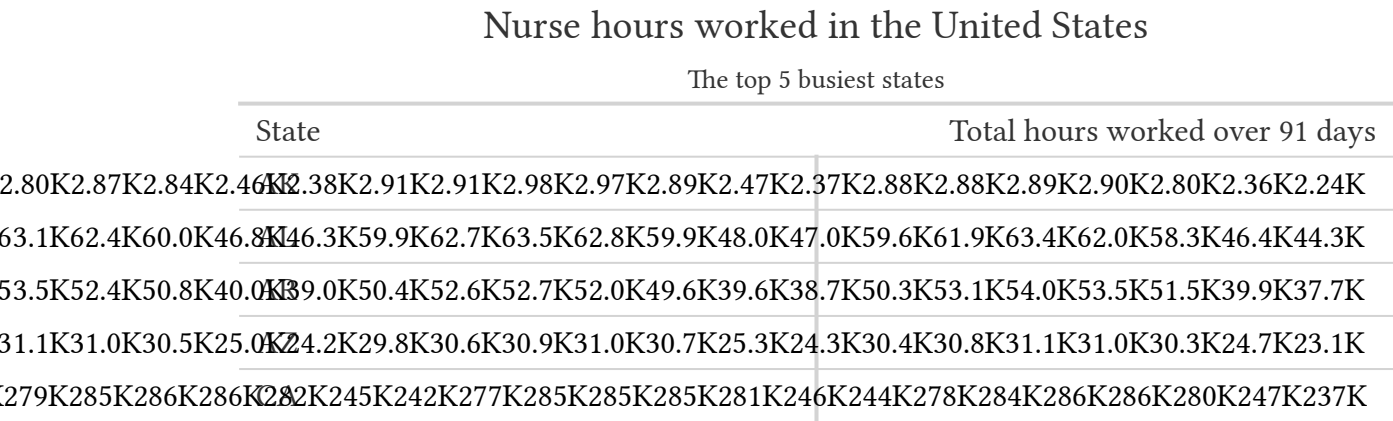


Figure 2: Sparklines of average work hours across 91 days by state.

Archive

Bibliography