<DATA SCIENCE TOOLBOX>

PROJECT REPORT

Data Insights from Hospital Cost Reports

SUBMITTED BY: ARYAN KUMAR

REG.NO: 12310657

SECTION: K23DW

COURSE CODE: INT375

ROLL NO: 03

LOVELY PROFESSIONAL UNIVERSITY

UNDER THE GUIDANCE OF

Vikas Mangotra UID (31488)

DISCIPLE OF CSE/IT

LOVELY SCHOOL OF COMPUTER SCIENCE

LOVELY POFESSIONAL UNIVERSITY, PHAGWARA

CERTIFICATE

This is to certify that **ARYAN KUMAR** bearing Registration no. **12310657** has completed **INT375** project titled, **"Data Insights from Hospital Cost Reports"** under my guidance and supervision. To the best of my knowledge, the present work is the result of his/her original development, effort and study

Vikas Mangotra
School of Computer Science
Lovely Professional University
Phagwara, Punjab.

DATE:12/04/2025

DECLARATION

I, **Aryan Kumar**, student of **COMPUTER SCIENCE** under CSE/IT Discipline at, Lovely Professional University, Punjab, hereby declare that all the information furnished in this project report is based on my own intensive work and is genuine.

DATE:12/04/2025

REGISTRATION NO: 12310657

TABLE OF CONTENTS

PROJECT OVERVIEW

Data Insights from Hospital Cost Reports

INTRODUCTION

This project analyzes U.S. hospital cost reports to uncover financial patterns, such as revenue trends, state-wise record distribution, and net income behavior over time.

DATASET DESCRIPTION

Title: Cost Report Dataset (Hospitals)
Source: CMS / Government cost reporting
Last Updated: 2023
Key Attributes:

Net Patient Revenue,
Net Income,
Medicare CBSA Number,
Fiscal Year Dates,
State Code,
Type of Control

PROJECT OBJECTIVE

• Clean the dataset
• Explore financial health via statistics
• Visualize revenue trends
• Predict future revenue using regression

SPECIFIC GOALS:

- Identify and remove non-essential financial columns from the dataset to improve clarity and performance.
- Analyze state-wise distribution of hospital cost report records.
- Visualize yearly trends in Net Patient Revenue and Net Income.
- Detect outliers in financial data (Net Revenue, Net Income, Medicare CBSA) using boxplots.
- Understand correlations between numeric financial indicators through a heatmap.
- Explore control types and their frequency in hospital data using donut charts.
- Use linear regression to forecast Net Patient Revenue for the years 2024 and 2025.
- Present key insights for policymakers and financial planners in the healthcare sector.

DATA IMPORT AND LIBRARIES

REQUIRED LIBRARIES

```
import pandas as pd                                  # Deals with tabular data (DataFrames)

import numpy as np                                   # Performs numerical/statistical operations

import matplotlib.pyplot as plt                      # Plots basic graphs

import seaborn as sns                                # Enhances visualization with plots

from sklearn.linear_model import LinearRegression    # to predicting a continuous numeric value
```

LODING DATASET AND INSPECTING

```
df = pd.read_csv('CostReport.csv')

df.shape


df.head()


df.info()

//to see null value

df.isnull().sum()
```

DATA CLEANING AND PREPATION

Getting the description of the dataset

df.describe

Cleaning Data And Removing The NULL Value

```
 drop = [
'Cash on Hand and in Banks',
'Temporary Investments',
'Notes Receivable',
'Prepaid Expenses',
'Inventory',
'Land',
'Land Improvements',
'Buildings',
'Leasehold Improvements',
'Minor Equipment Depreciable',
'Health Information Technology Designated Assets',
'Total Fixed Assets',
'Investments',
'Other Assets',
'Accounts Payable',
'Payroll Taxes Payable',
'Notes and Loans Payable (Short Term)',
'Deferred Income',
'Unsecured Loans',
'Total Other Assets',
'Total Fund Balances',
    'DRG Amounts Before October 1',
    'DRG Amounts After October 1',
    'Outlier Payments For Discharges',
    'Medicaid Charges',
    'Stand-Alone CHIP Charges'
]

clean= df.drop(columns=drop, errors='ignore')
clean.to_csv('CostReport.csv', index=False)
```

EDA and VISUAL DATA EXPLORATION

//Creating a count of records per state

```
state_counts=df['StateCode'].value_counts().sort_values(ascending=Fal
se)
plt.figure(figsize=(12, 6))
sns.barplot(x=state_counts.index, y=state_counts.values)
plt.title('Distribution of Records by State')
plt.xlabel('State')
plt.ylabel('Number of Records')
plt.xticks(rotation=90)
plt.show()
```

//Bar Chart: Net Patient Revenue by Year

```
 print(df.columns[df.columns.str.contains("date", case=False)])
df['Report Date'] = pd.to_datetime(df['Fiscal Year Begin Date'])
df['Year'] = df['Report Date'].dt.year
df.groupby('Year')['Net Patient Revenue'].sum().plot(kind='bar')
plt.title('Net Patient Revenue by Year')
plt.show()
```

// Outliers of Net Patient Revevue , Net Income and Medicare CBSA Number

```
sns.set(style="whitegrid")

cols_to_plot = ['Net Patient Revenue', 'Net Income', 'Medicare CBSA Number']

fig, axes = plt.subplots(1, len(cols_to_plot), figsize=(18, 6), constrained_layout=True)

for i, col in enumerate(cols_to_plot):
    if col in df.columns:
        sns.boxplot(data=df, x=col, ax=axes[i], color='skyblue', width=0.4, fliersize=4)
        axes[i].set_title(f"{col}", fontsize=14, fontweight='bold')
        axes[i].set_xlabel("")
        axes[i].tick_params(axis='x', rotation=30)
plt.suptitle("Outlier Detection Using Boxplots", fontsize=16, fontweight='bold')
 plt.show()
```

// Heatmap correlation

```
        corr = df.corr(numeric_only=True)
        plt.figure(figsize=(12, 10))
        sns.heatmap(corr,cmap='coolwarm',annot=False,linewidths=0.5)
        plt.title('Correlation Heatmap')
        plt.show()
```

// LINEPLOT: Average Net Income by Fiscal Year

```
df['Fiscal Year End Date'] = pd.to_datetime(df['Fiscal Year End Date'], errors='coerce')

if 'Net Income' in df.columns:
    df['Net Income'] = pd.to_numeric(df['Net Income'], errors='coerce')
    trend_df = df.groupby(df['Fiscal Year End Date'].dt.year)['Net Income'].mean().dropna()

    plt.figure(figsize=(10, 5))
    sns.lineplot(x=trend_df.index, y=trend_df.values, marker="o")
    plt.title("Average Net Income by Fiscal Year")
    plt.xlabel("Year")
    plt.ylabel("Average Net Income")
    plt.grid(True)
    plt.tight_layout()
    plt.show()
else:
    trend_df = None

trend_df
```

//Scatterplot: Net Patient Revenue vs Net Income(Scatter plot)

```
# Drop rows with missing values
df_scatter = df[['Net Patient Revenue', 'Net Income']].dropna()

plt.figure(figsize=(10, 6))
sns.scatterplot(data=df_scatter, x='Net Patient Revenue', y='Net
Income', alpha=0.6)
plt.title('Scatter Plot: Net Patient Revenue vs Net Income')
plt.xlabel('Net Patient Revenue')
plt.ylabel('Net Income')
plt.grid(True)
plt.tight_layout()
plt.show()
```

//BarPlot: Record Distribution by State (Top 15)

```
if 'State Code' in df.columns:
  plt.figure(figsize=(12, 6))
  state_counts=df['StateCode'].value_counts().sort_values(ascending
  =False).head(15)
   sns.barplot(x=state_counts.index,y=state_counts.values,
palette="viridis")
  plt.title("Top 15 States by Number of Records")
  plt.xlabel("State")
  plt.ylabel("Record Count")
  plt.xticks(rotation=45)
  plt.tight_layout()
  plt.show()
```

// Pie Chart: Records by State Code of top 10

```
  if 'State Code' in df.columns:
provider_counts = df['State Code'].value_counts().head(10)

plt.figure(figsize=(8, 8))
plt.pie(
    provider_counts.values,
    labels=provider_counts.index,
    autopct='%1.1f%%',
    startangle=140,
    wedgeprops={'edgecolor': 'black'}
)
plt.title('Top 10 Provider Types by Record Count')
plt.axis('equal')
plt.tight_layout()
plt.show()
```

// Donut Chart by 'type of control' of top 10

```
col = 'Type of Control'

if col in df.columns:
    top_counts = df[col].value_counts().head(10)

    plt.figure(figsize=(8, 8))
    wedges, texts, autotexts = plt.pie(
    top_counts.values,
    labels=None,
    autopct='%1.1f%%',
    startangle=140,
    wedgeprops={'width': 0.4, 'edgecolor': 'white'}
)
plt.legend(
    wedges,
    top_counts.index,
    title=col,
    loc='center left',
    bbox_to_anchor=(1, 0.5)
)

plt.title(f'Top 10 {col} Distribution (Donut Chart)')
plt.axis('equal')
plt.tight_layout()
plt.show()
```

// **Prediction for 2024 & 2025**

```
df['Fiscal Year End Date'] = pd.to_datetime(df['Fiscal Year End Date'], errors='coerce')
df['Fiscal Year'] = df['Fiscal Year End Date'].dt.year

revenue_by_year = (
    df.groupby('Fiscal Year')['Net Patient Revenue']
    .sum()
    .dropna()
    .sort_index()
)

revenue_by_year
years = revenue_by_year.index.values.reshape(-1, 1)
revenues = revenue_by_year.values

# Train linear regression model
model = LinearRegression()
model.fit(years, revenues)

future_years = np.array([[2024], [2025]])
future_predictions = model.predict(future_years)

all_years = np.append(years.flatten(), future_years.flatten())
all_revenues = np.append(revenues, future_predictions)

plt.figure(figsize=(10, 6))
plt.plot(all_years, all_revenues / 1e9, marker='o', linestyle='-', color='blue', label='Net Patient Revenue (Billion USD)')

for x, y in zip(all_years, all_revenues):
    plt.text(x, y / 1e9 + 10, f"${y/1e9:.1f}B", ha='center', fontsize=9)
```

```
plt.plot(future_years, future_predictions / 1e9, 'ro', label='Predicted')

plt.title("Net Patient Revenue by Fiscal Year (with 2024 & 2025 Prediction)", fontsize=14)
plt.xlabel("Fiscal Year")
plt.ylabel("Revenue (in Billion USD)")
plt.grid(True, linestyle='--', alpha=0.6)
plt.legend()
plt.tight_layout()
plt.show()
```

KEY FINDINGS

• Net Patient Revenue Trends

Net Patient Revenue has shown a consistent upward trend from earlier years to the most recent. Forecasts for 2024 and 2025 using linear regression indicate continued growth.

• Net Income Variability

Net Income exhibits year-to-year fluctuations, suggesting sensitivity to operational costs and external factors, despite growing revenues.

• State-wise Record Concentration

States like California, Texas, and New York dominate the dataset in terms of record count, which may correlate with hospital density and population size.

• Outliers in Financial Data

Boxplots revealed significant outliers in Net Patient Revenue and Net Income, indicating the presence of large healthcare systems or anomalies requiring deeper analysis.

• Control Type Distribution

Donut charts showed a diverse mix of control types—governmental, non-profit, and for-profit—highlighting variation in hospital ownership structures.

• Correlation Patterns

Correlation heatmaps showed relationships between financial variables, such as Net Patient Revenue being moderately correlated with total charges, providing insight for further modeling.

PROJECT SUMMARY & TAKEAWAYS

📊 Summary of Visualizations & Analysis

- The dataset was thoroughly cleaned by removing irrelevant columns and handling missing values, ensuring accurate analysis.
- State-wise visualizations highlighted where most hospital records originated, giving insight into healthcare density across states.
- Trends in **Net Patient Revenue** showed consistent growth, while **Net Income** fluctuated, emphasizing operational cost impact.
- Outlier detection through boxplots revealed financial extremes that could represent large hospital systems or reporting inconsistencies.
- A heatmap helped identify meaningful correlations between financial metrics.
- Donut and pie charts gave clarity on hospital control types and state-wise record shares.
- Linear regression successfully predicted revenue for 2024 and 2025, indicating a positive financial outlook.

🧠 Major Insights

- **Revenue Growth:** Hospitals are experiencing increased patient revenue over time.
- **Income Instability:** Profitability doesn’t always grow in parallel with revenue, suggesting inefficiencies or cost fluctuations.
- **State-Level Variations:** Certain states dominate the dataset, likely due to size and healthcare infrastructure.
- **Ownership Diversity:** Public, private, and non-profit hospitals contribute differently to the financial landscape.

🏛️ Policy & Infrastructure Implications

- Policymakers and hospital administrators can use these insights to plan funding, staffing, and infrastructure expansions.
- High-revenue growth regions may require increased investment in medical technology and capacity.
- Prediction tools can assist in future budgeting and resource allocation across healthcare networks.

REFERENCES

DATASET SOURCE

Title: Data Insights from Hospital Cost Reports
URL: Data.gov Home - Data.gov
Last Updated: 2023

EXTERNAL TOOLS & LIBRARIES USED

Python Libraries:

- pandas – for data manipulation and preprocessing
- numpy – for numerical and statistical analysis
- matplotlib – for plotting and visualization
- seaborn – for advanced statistical graphics