Findings & Justifications

I created a Structured Query Language (SQL) files of all the Dataset provided in the question.

1) HospitalProfiling.sql was created.
2) Cleaned Data of HospitalProfiling.sql by taking the avg Empoyee count and stored in HospitalProfiling_con_avg.sql

Few Columns were added (Year_Average,Year_Median,Max_Year,Min_Year,First_Quarter_Sum,Second_Quarter_Sum,Third_Quarter_Sum,Fourth_Quarter_Sum,First_Quarter_Avg, Second_Quarter_Avg,Third_Quarter_Avg,Fourth_Quarter_Avg,First_Half_Year_Total,First_Half_Year_Average,Second_Half_Year_Total,Second_Half_Year_Average) to HospitalRevenue.csv to make HospitalRevenue_added_cols.sql

3) Removed "NO District Available" District_Dd rows from HospitalRevenue_added_cols.sql
4) Added Buy_or_not column to ProjectedRevenue.sql
5) Removed "NO District Available" District_id rows from ProjectedRevenue.sql
6) Created table `hospitalprofiling+revenue.sql` which contains the join of Hospital Profile and Hospital Revenue to get the Hospital_Employees with the Revenues table.
7) Checked for duplicates in hospitalprofiling+revenue table, none found :)
8) Created `projectedrevenue_added_cols.sql` by adding Buy_or_not as 1
9) Created `hospitalprofiling_projectedrevenue.sql` by joining `projectedrevenue_added_cols` & `hospitalprofiling+revenue` tables
10) Created `hospitalprofiling_minus_revenue.sql` by subtracting `hospitalprofiling_projectedrevenue` from `hospitalprofiling+revenue` tables
11) Created dataset.csv by combining 2 CSV files `hospitalprofiling_projectedrevenue` & `hospitalprofiling_minus_revenue`

Assuming the `hospitalprofiling_minus_revenue.csv` has the not sold data as they were not found to have the predicted values when we created `hospitalprofiling_projectedrevenue`

This dataset.csv was my Main Dataset.

Revenue Prediction:

1) I first chose Regression to find predicted Revenues.
2) Divided this dataset into 70%, 30% to create the training and testing dataset respectively.
3) Drew the correlation plots, box plots and summary plots to have a quick look at the trainset.
4) Next ran the regression algorithm using the `glm` library available in R.
5) Tested this against the testing set. Got fairly decent results.
6) Then scored the Solution.csv file with the predicted value from my model.

Buy_or_not Prediction:

1) I then chose the ADA Boost algorithm with 50 iterations to predict this target variable.
2) The same 70, 30 ratio was used for training and testing datasets.
3) After the testing gave 100% accuracy I scored the Solution.csv file with my new model.

The final Solution.csv that was generated was uploaded in the SQL table.

Wherever the Buy_or_not attribute was 0, the Predicted revenue value was made 0.

Queries used:

--- To find Duplicate data:

```
SELECT *,count(*) FROM `mytable` GROUP BY Hospital_ID,Region_ID,District_ID,Instrument_ID,Month_1,Month_2,Month_3,Month_4,Month_5,Month_6,Month_7,Month_8,Month_9,Month_10,Month_11,Month_12,Year_Total,Year_Average,Year_Median,Max_Year,Min_Year,First_Quarter_Sum,Second_Quarter_Sum,Third_Quarter_Sum,Fourth_Quarter_Sum,First_Quarter_Avg,Second_Quarter_Avg,Third_Quarter_Avg,Fourth_Quarter_Avg,First_Half_Year_Total,First_Half_Year_Average,Second_Half_Year_Total,Second_Half_Year_Average HAVING count(*)>1 ORDER BY `Month_1` ASC
```

--- Hospital Profile and Hospital Revenue join

```
select hr.Hospital_ID,hr.Region_ID,hr.District_ID,hr.Instrument_ID,Month_1,Month_2,Month_3,Month_4,Month_5,Month_6,Month_7,Month_8,Month_9,Month_10,Month_11,Month_12,Year_Total,Year_Average,Year_Median,Max_Year,Min_Year,First_Quarter_Sum,Second_Quarter_Sum,Third_Quarter_Sum,Fourth_Quarter_Sum,First_Quarter_Avg,Second_Quarter_Avg,Third_Quarter_Avg,Fourth_Quarter_Avg,First_Half_Year_Total,First_Half_Year_Average,Second_Half_Year_Total,Second_Half_Year_Average,hp.Hospital_employees from hospitalprofiling_conv_avg hp,hospitalrevenue_added_cols hr where hr.Hospital_ID=hp.Hospital_ID and hr.District_ID=hp.District_ID
```

--- Hospital Revenue and Projected Revenue join

```
select hpr.Hospital_ID,hpr.Region_ID,hpr.District_ID,hpr.Instrument_ID,Month_1,Month_2,Month_3,Month_4,Month_5,Month_6,Month_7,Month_8,Month_9,Month_10,Month_11,Month_12,Year_Total,Year_Average,Year_Median,Max_Year,Min_Year,First_Quarter_Sum,Second_Quarter_Sum,Third_Quarter_Sum,Fourth_Quarter_Sum,First_Quarter_Avg,Second_Quarter_Avg,Third_Quarter_Avg,Fourth_Quarter_Avg,First_Half_Year_Total,First_Half_Year_Average,Second_Half_Year_Total,Second_Half_Year_Average,Hospital_employees,Annual_Projected_Revenue,Buy_or_not from `hospitalprofiling+revenue` hpr,projectedrevenue_added_cols pr where hpr.Hospital_ID=pr.Hospital_ID and hpr.District_ID=pr.District_ID and hpr.Instrument_ID=pr.Instrument_ID
```

*** `projectedrevenue_added_cols` - `hospitalprofiling_projectedrevenue` [Sold instruments, but not in my inventory

```
select Hospital_ID,District_ID,Instrument_ID from projectedrevenue_added_cols pra where not exists (select Hospital_ID,District_ID,Instrument_ID from hospitalprofiling_projectedrevenue hppr where pra.Hospital_ID=hppr.Hospital_ID and pra.District_ID=hppr.District_ID and pra.Instrument_ID=hppr.Instrument_ID)
```

--- `hospitalprofiling+revenue` - `projectedrevenue_added_cols` [NOT Sold instruments, but in my inventory]

```
select pra.Hospital_ID,pra.District_ID,pra.Instrument_ID,Month_1,Month_2,Month_3,Month_4,Month_5,Month_6,Month_7,Month_8,Month_9,Month_10,Month_11,Month_12,Year_Total,Year_Average,Year_Median,Max_Year,Min_Year,First_Quarter_Sum,Second_Quarter_Sum,Third_Quarter_Sum,Fourth_QuarterS
```

um,First_Quarter_Avg,Second_Quarter_Avg,Third_Quarter_Avg,Fourth_Quarter_Avg,First_Half_Year_Total,First_Half_Year_Average,Second_Half_Year_Total,Second_Half_Year_Average,Hospital_employees from `hospitalprofiling+revenue` pra where not exists (select Hospital_ID,District_ID,Instrument_ID from projectedrevenue_added_cols hppr where pra.Hospital_ID=hppr.Hospital_ID and pra.District_ID=hppr.District_ID and pra.Instrument_ID=hppr.Instrument_ID)

Also,

Based on Correlation plots, removed Min _Year & Hospital_employees Column [Terribly low correlation]; removed First.Quarter.Sum,Second.Quarter.Sum,Third.Quarter.Sum,Fourth.Quarter.Sum,Second.Half.Year.Total, Year.Total Columns since same correlation as their Average counterparts hence, created ds1.csv

Normalized ds1 by recentering.

--Abhishek Karan