Contents:

Overview

The dataset given for this project consists of 7043 rows and 10 rows. After applying data preprocessing methods on the given dataset, the new dataset, called scaled dataset, has 6940 rows and 12 columns, with the 12th column (Churn) being the target variable. This scaled dataset was preprocessed; therefore, the final feature set and target variable were separated before performing the train-test split.

Splitting the Dataset

To evaluate the model performance effectively, the dataset was split into two subsets:

- Training Set (80%)
- Testing Set (20%)

The train_test_split function from the scikit-learn library was used for this task, with the following key configurations:

- Test size: 20% of the dataset
- Random seed (random_state): 42 was used to ensure reproducibility
- Stratification: The target variable (Churn) was used for stratification. This ensures that both training and testing sets have the same distribution of the target class.
    - This step is crucial when dealing with imbalanced data, ensuring the proportion of the target classes (Churn vs. Non-Churn) remains consistent across both training and testing sets.

Composition of the Data Splits

Training Set:

- Number of Samples (Rows): 5552 - 80% of the scaled dataset
- Number of Features (Columns): 11 - excluding the target variable “Churn”
- Class distribution (Target Churn):
    - The distribution of the target variable in the training set remains consistent due to stratification.
    - Classes: Class 0 (Non-Churn) and Class 1 (Churn)

Testing Set:

- Number of Samples (Rows): 1388 - 20% of the scaled dataset
- Number of Features (Columns): 11 - excluding the target variable “Churn”
- Class distribution (Target Churn):
    - Similar distribution of the Churn target variable due to stratification
    - Classes: Class 0 (Non-Churn) and Class 1 (Churn)

Code Implementation

```
# splitting the dataset into distinct training and testing sets
# saving the training and testing sets into their own folders

# scaled_df_final
print('Number of Instances =', scaled_df_final.shape[0])

# drop the target column('Churn') from the scaled dataset
x = scaled_df_final.drop(columns=['Churn']) #features
y = scaled_df_final['Churn'] #target
```

```
Number of Instances = 6940
```

```
# randomising data split
# splitting the data into training and testing sets (80% for training, 20% for testing)
X_train, X_test, y_train, y_test = train_test_split(x, y, test_size=0.2, random_state=42, stratify=y)
```

```
# saving the training data and testing data into their respective folder
X_train.to_csv('./TrainingSet/X_train.csv', index=False)
X_test.to_csv('./TestSet/X_test.csv', index=False)
y_train.to_csv('./TrainingSet/y_train.csv', index=False)
y_test.to_csv('./TestSet/y_test.csv', index=False)

# the following can be used to visualise the index of the dataset(index shows randomise dataset)
# X_train.to_csv('./TrainingSet/X_train.csv', index=True)
```

Conclusion

The dataset was successfully split into an 80-20 ratio for training and testing purposes. The stratification ensured that the distribution of the target variable remained similar across both sets, which is crucial for model evaluation, especially in cases where the dataset is imbalanced.

The training and testing datasets were saved into separate CSV files in the directories called TrainingSet and TestSet, respectively:

- Training Features: X_train.csv
- Testing Features: X_test.csv
- Training Target: y_train.csv
- Testing Target: y_test.csv