Repeatable ETL Report

Cory Chitwood, Eddy Doering, Kai Gui, Keith-Jordan Wilkinson

Introduction

In our project, we utilize a dataset of approximately 100,000 anonymized electronic health records (EHR) from hospitalizations of diabetes patients from 1999-2008 compiled by Virginia Commonwealth University researchers from the CERNER Health Facts Database.[1] We First explore and visualize the demographics of diabetes patients represented within the dataset, comparing our findings to 2008 national demographics identified from census data.[2] Next, we apply machine learning models to predict if diabetes patients will experience hospital readmission based on features of their original hospital stay.

Data Sources

1. Strack B, DeShazo JP, Gennings C, Olmo JL, Ventura S, Cios KJ, Clore JN. Impact of HbA1c Measurement on Hospital Readmission Rates: Analysis of 70,000 Clinical Database Patient Records. Biomed Res Int. 2014:781670. DOI: 10.1155/2014/781670
2. U.S. Census Bureau. HIC-9_ACS. Population Without Health Insurance Coverage by Race and Hispanic Origin: 2008 to 2019. Health Insurance Historical Tables - HHI Series. Link

Diabetes EHR Dataset

Simulating Streaming Data

1. The diabetes dataset was downloaded and uploaded into our data lake as a csv file.
2. From a data factory, we ran our Kafka Producer databricks file.
    1. The Kafka Producer imported the diabetes dataset csv as a PySpark dataframe.
    2. The PySpark dataframe was converted into a list of dictionaries, where each row of the csv file is a dictionary.
    3. Each dictionary in the list was converted to JSON and produced as a Kafka message, sleeping for a short time between messages.

Extraction

1. From a data factory, we ran our Kafka Consumer databricks file.
    1. The Kafka Consumer received the JSON messages, converted them into dictionaries, and appended the information to an empty list.
    2. When the list of dictionaries hit a length of 500 items, it was saved as a csv file, and the list was emptied to accommodate additional consumed messages.
2. Each csv file in the data lake was combined into a PySpark dataframe in databricks.

Transformation

1. The PySpark dataframe was converted into a Pandas dataframe, so that missing data represented as "?" strings could be converted into *Null* values.

Load

1. Using PySpark and JDBC, the transformed diabetes dataset was written into appropriate SQL databases.

Census Insurance Dataset

Extraction & Transformation

1. The census table was downloaded and extraneous information was removed.
    1. Only data for the year 2008 and for the US as a whole was kept, since data for individual states and for other years were not relevant for our purposes.

Load

1. Using PySpark and JDBC, the transformed census dataset was written into appropriate SQL databases.

Machine Learning Transformation Steps:

ML Algorithms: Random Forest Classifier, SGDClassifier, KNN Classifier, Ridge Classifier

General Steps

1. CSV data imported using pandas into Jupyter Notebooks.
2. Remove rows with *Null* values from 'race' column.
3. Remove rows with 'Unknown/Invalid' values from 'gender' column.
4. Remove columns: 'encounter_id', 'patient_nbr', 'weight'.
5. Create dummy variables for remaining categorical data.
6. Create 'X' and 'y' dataframes, where the column to predict is 'readmitted'
7. Split X and y into training and testing sets.
8. Fit the model on training data
9. Score the test data.

Optional steps explored in ML optimization:

- Refactor 'readmitted column': Replace values '<30' and '>30' in with 'YES'
- Drop 'medical_specialty' column, or drop *Null* rows in 'medical_specialty', or drop specific values in 'medical_specialty'.
- Drop 'payer_code' column
- Encode medications columns into numeric values, or refactor into binary 'Yes/No"