Name: Cushing, Bradley
Date: 11/04/24
NYU ID: N10695516
Course Section Number: CSCI-GA-2433-001
Project Part #2 Report

Files included

- cushing_p2_su24_report.pdf
- cushing_p2_su24_logical_schema.sql
- cushing_p2_su24_cb_er_model.sql
- cushing_p2_su24_dol_title_wages.csv
- cushing_p2_su24_warranties_actual.csv
- prediction/model.pkl
- prediction/train-model.py
- prediction/predict-risk.py
- prediction/warranties-actual.csv
- prediction/warranties-random.csv

Logical database schema

The logical database schema has been created in a file cushing_p2_su24_logical_schema.sql. This includes all the necessary SQL code (DDL) to create the database according to the relational schema specified in the first step.

I've also updated the conceptual schema first to include PERSONAL_INFORMATION for users which is described below for those that want to submit it and two new Discount_pct and Price fields in the WARRANTY table which uses the price set by the prediction explained below. I've included a new copy of the conceptual schema in cushing_p2_su24_cb_er_model.sql which also reflects this change. We expect the Discount_pct and Price to be set based on what exists at time of record creation.

Unstructured data collection

I did collect some data that I felt would be useful from the Department of Labor. The idea is that based on wage information for specific job titles I would be able to predict and benchmark salaries paid to employees within the Circuit Blocks Inc. organization. I've included the data set called cushing_p2_su24_dol_title_wages.csv which includes this data but I've decided to instead use some mock data which represents personal information of customers that I can use to predict risk for warranties which is explained below. The reason being this was much easier to use for prediction using Logistic Regression.

I searched quite extensively for good external data that I could use to train a Logistic Regression model but I had a very hard time finding something that would be good for my use case. Therefore, after discussing with the professor I chose to create my own dataset that could be used for training such a model. I hand picked attributes that I thought could be interesting indicators of whether a person who would be purchasing a warranty would be deemed “risky”.

The general idea

The premise is that we would allow customers who have purchased the product to give us additional personal information about themselves and their household and offer them a discount in return. The company Circuit Blocks Inc. sells warranties for the products that they make and sell to customers. Warranties are order specific and are priced at 20% of the original product order price. Anyone who submits additional, personal information will only have to pay 15% of the order price (we’ll use this information to inform marketing campaigns which is why we value it) and those we predict as not risky will only have to pay 10%. When someone uses a warranty they get a full replacement for that order.

You could imagine our prediction model being trained on historical data internally from the company after collecting this information for a few months or years. The company would label each warranty with a 1 if they ended up using the warranty and 0 if they didn’t. This data then serves as a good training set to predict future risk.

Price per warranty

- (submits no information) 20% of original order
- (submits information but risky) 15% of original order
- (submits information but not risky) 20% of original order

Semi-structured data set

Risk categorization based on living situation

- **cushing_p2_su24_warranties_actual.csv**
- prediction/warranties-actual.csv

There are a few copies of the external data used to train the prediction model using logistic regression. Please see cushing_p2_su24_warranties_actual.csv for the actual data. I’ve included a few rows from the sample data below to give an overview.

Example CSV data

Age	Kids_count	Pets_count	Siblings_count	Income	Has_risk
35	5	3	0	97638	1
61	3	5	2	87510	1
56	1	1	4	173967	0

How insights inform EDA created

The first 5 columns of data are used as an indicator for whether the person purchasing a warranty for the product should be considered as “risky” or not. We will process this information submitted by a user who has purchased the product if they choose to also purchase a warranty.

Semi-structured files

We consider unstructured data to be semi-structured in reality as what we want to process from external sources will typically be in CSV, XML, or JSON which we can read, parse, normalize, and reformat into something structured like a table in our relational database. Then we can query our relational database with this structured data to gain insights and inform the business. We won’t consider completely unstructured data like text documents, audio files, etc.

Model training

Training is done in the prediction folder using the data set called “warranties-actual.csv” which is described above. I’ve included this file which has around 1000 fields along with the report. The fields we use for prediction are Age, Kids_count, Pets_count, Siblings_count, and Income amount. We deem these as good predictors for whether or not the person is “risky” which is someone we consider is more likely to use their warranty and request a replacement for their product. The data itself has been sorted by the Has_risk field so it’s easier to read. Note that the intuition is that those with more kids and pets will have higher risk and that those with more siblings and higher income will have lower risk.

Prediction

When trained on the actual data there is 90% accuracy of prediction on actual data. When trained on random data there is 48% accuracy which is to be expected. We used this random data as a sanity check to make sure there was a clear improvement on a realistic looking data set. Training on the real data with distinct patterns that exist in the real world allows us to benefit from making this prediction and charging more for those with more risk.

Data lake and pipeline

Given the above use of historical personal information related to warranty signups from actual customers within the business, we’re able to have a conceptual and logical schema which is mostly relational. We only require CSV files for the training data and generated models which we store in S3 (Cloud Object Storage). The data lake and data pipeline is as follows:

- CUSTOMER registers WARRANTY
- CUSTOMER gives PERSONAL_INFORMATION
- PERSONAL_INFORMATION table contains all records
- All rows are exported for training to CSV file weekly
- This CSV file is stored in an AWS S3 bucket
- The model is trained weekly on the latest CSV file

- Every model is stored in an AWS S3 bucket
- New CUSTOMER gives PERSONAL_INFORMATION
- Latest risk prediction is entered in database with above information
- Risk prediction is update for users on each warranty request

Data lake components

Because our conceptual and logical schemas are for relational databases only we only model the relational parts of the data lake and the above pipeline there. Below we list the 3 components of our data lake for ingesting, training, and inferencing data.

- PostgreSQL (WARRANTIES, PERSONAL_INFO)
- S3 (Cloud Object Storage) for training and model CSV files
- Python scripts for training and prediction inference