ECE9017 Final Report

Group Number: 5

Group Name: Youth With You

Group Member:

Ruoran Liu	251120523
Yan Liu	251127692
Yufei Ding	251124178
Yanhao Huang	251102260

Project idea and Objectives

Our project develops a data warehouse and analysis system for a medium-sized video game retail chain. Suppose there are several retailers for one brand, and each of them uses different database systems such as Oracle, SQL Server or NoSQL DBs. Using the Extract out of the Extract Transform Load (ETL) would help get you data from different data sources into one homogeneous format.

There are also some logic and calculations like percentage profits, total number of selling across stores and so on would be needed to better get the data ready for analysis. We do that on bulk of data using the Transform of ETL. Besides, a lot of data cleaning activities that might be needed to get better quality data, in case there might be erroneous data pushed by the retailers. This can be achieved using ETL.

We will build the ETL process using combinations of T-SQL using stored procedures and SSIS packages. We will also create multi-dimensional cube using analysis Service SSAS, demonstrating the data source view, dimension hierarchies, Cube structure, measures and measure groups, Dimension usage, calculation, KPIs, MDX query. This will help to aggregate (Summarize) the data for high performance.

Business requirements and technical requirements

Operational Database

The database will keep track of all the order related details of the game retail chain. There are 10 relational tables which are Country, City, Address, Store, Staff, Customer, Game, Platform, Inventory, Order, and 2 conjunction tables Inventory_Order and Game_Platfrom used for many-to-many relationships. Each table has their unique surrogate key.

The columns in each table are shown by following:

Country - country_id, name
City - city_id, city_name, country_id
Address - address_id, address, direction, postal_code, city_id
Store - store_id, address_id
Staff - staff_id, manager_staff_id, store_id, first_name, last_name, email
Customer - customer_id, address_id, first_name, last_name, email
Game - game_id, rating, released_data, price, name, players_count, recommendation_count
Platform - platform_id, name
Inventory - game_id, stroe_id, stock_quantity

Order - staff_id, customer_id, order_date
Inventory_Order - inventory_id, order_id, quantity
Game_Platfrom - game_id, platform_id

Each customer has full contact information including first name, last name, address and email address. Each customer has one address, whereas each address can belong to multiple customers.

Each staff has full contact information including first name, last name, email address and store id where they work at. Each staff also has another staff identified as their manager (except of course for the CEO).

Each order is transacted by a customer and a staff along with the transaction date. A customer can order multiple times and a staff can transact multiple orders.

Each store has a unique address which is composed of address, direction, postal code, city and country. Each store has at least one staff, whereas a staff only works in one store. Each store has multiple inventory records.

Each inventory has full stock information including game id and store id, and the stock quantity. Each order can order from multiple games, at the same time, each game can be ordered by multiple orders.

Each game has full game information including the game name, rating, released date, unit price, the number of players for this game, the number of players who recommend this game. Besides, each game also gives the information which platform the game can be played on. Each game can be played on multiple platforms and each platform can play multiple games.

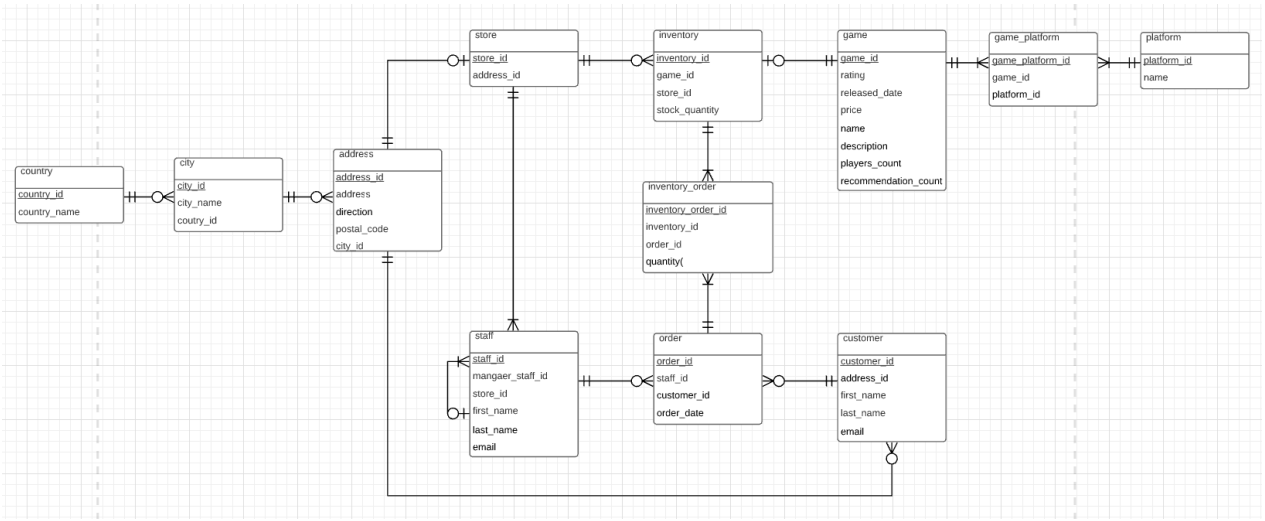


Fig. ER Diagram

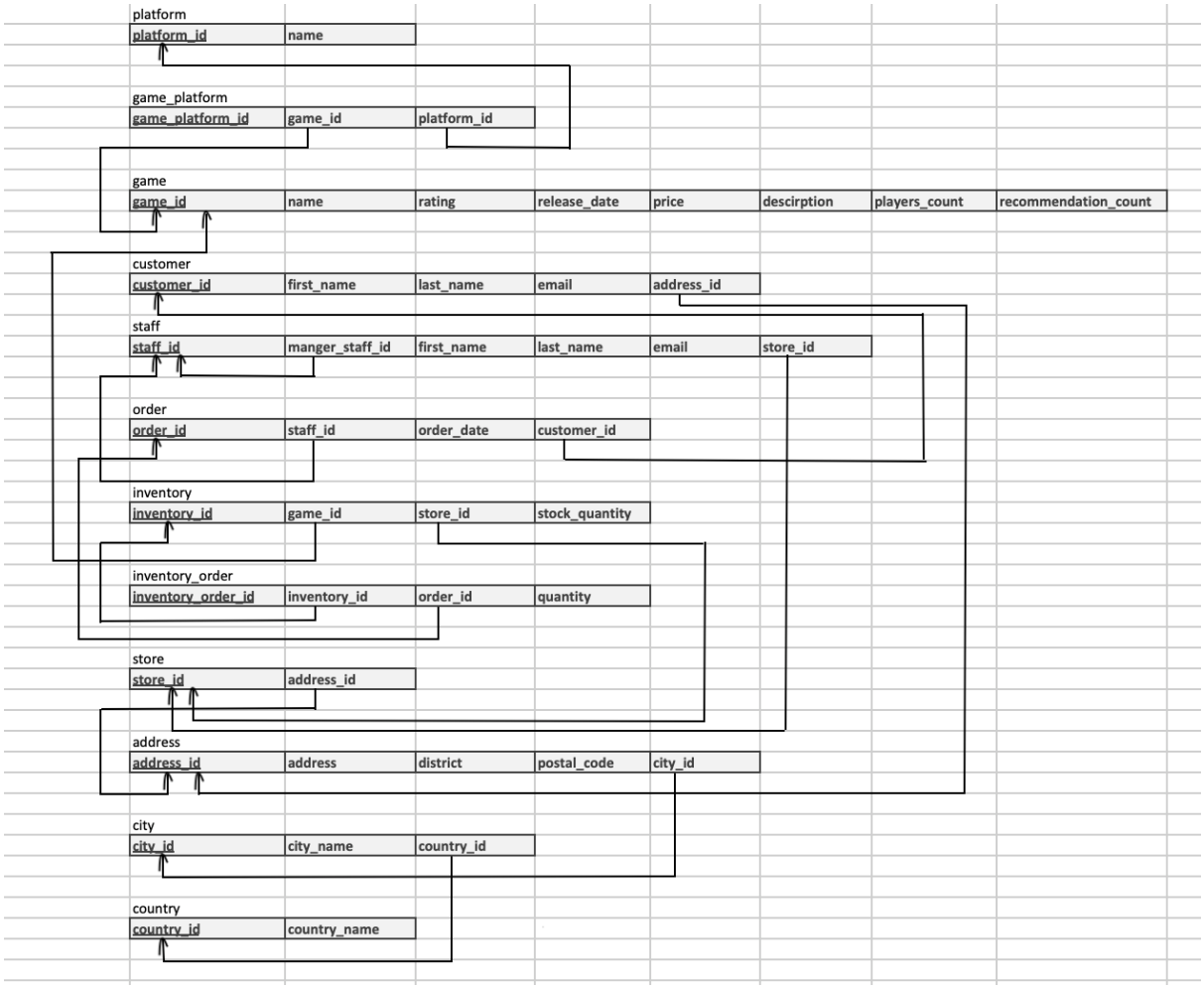


Fig. Relational Database Schema 3NF

Data Warehouse

A data warehouse is a central repository of information that can be analyzed to make better informed decisions. In our project, we created a data warehouse to calculate the total sales and the number of sold products. We create some dimension tables and a fact table as below.

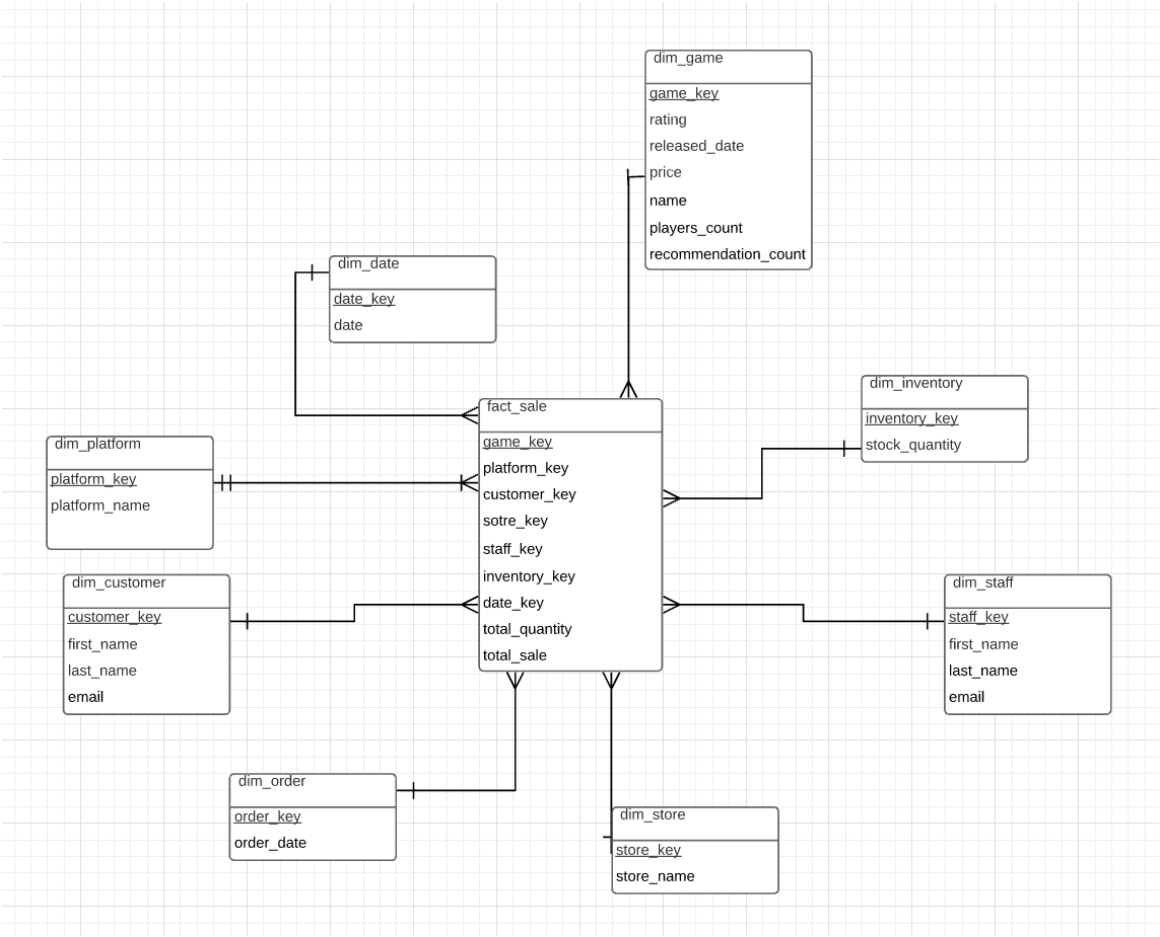


Fig. Structure of Data Warehouse

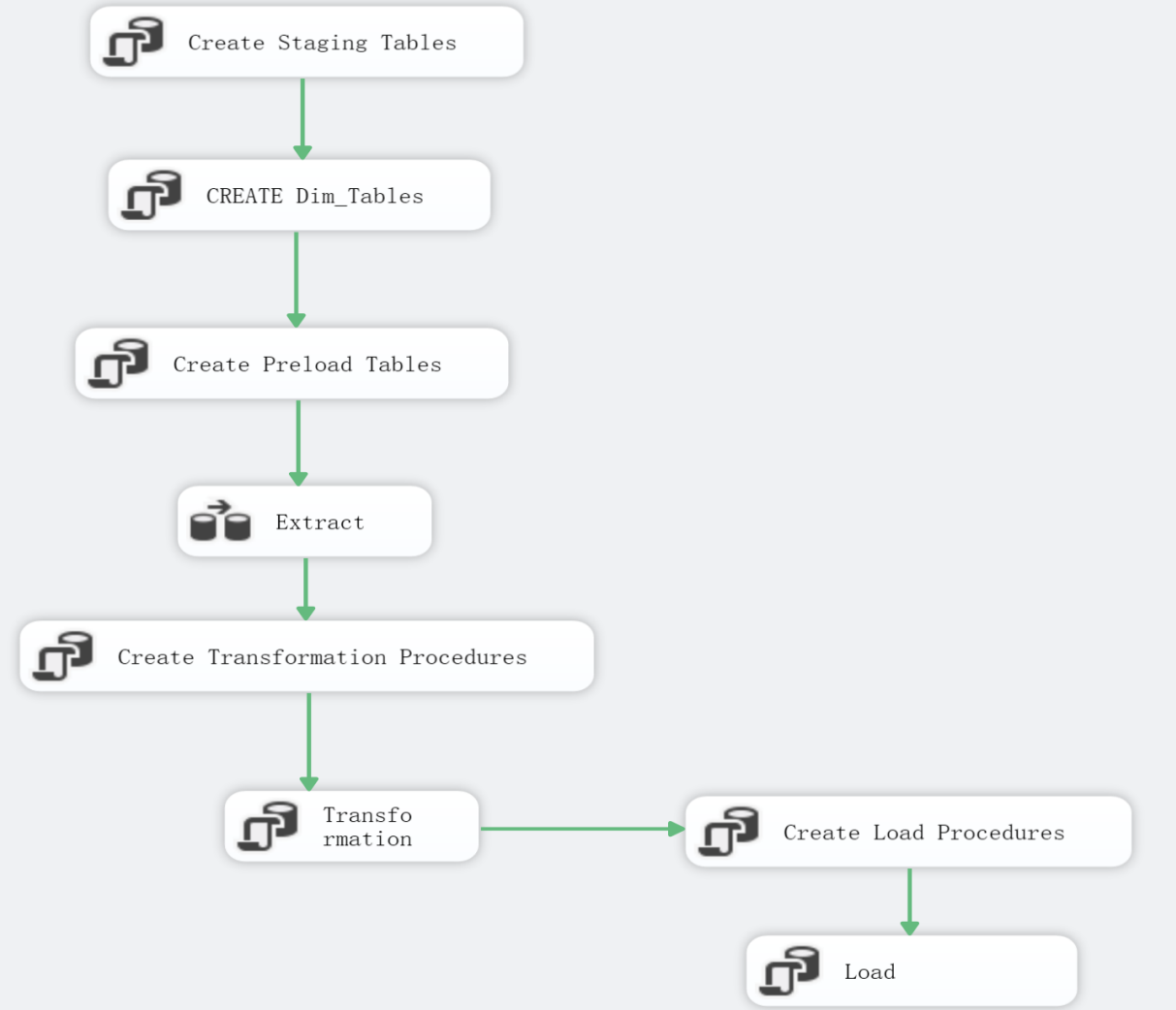


Fig. ETL Process Diagram

Extract

In our program, we create dim_table, staging_table and preload_table. Then use procedures to extract data we used in the source data warehouse. In the visual studio, extract procedures were replaced by SSIS tools. We have two methods to extract data. The first execute by using T-SQL. The other one is executed by SSIS.

Transform

This process is to transform the staging tables data into preload tables. When inserting data into the proload tables, the process will compare staging data with dimension table data. If there is new data from staging tables, insert data directly into preload tables. If the data in staging tables only change some details, the process will update data from dimension tables. Transformation process also has to determine slow changing types. The customer table in our project is SCD type2. Others are SCD type1. We used T-SQL to execute this process.

Load

Load process is to load data from preload tables to dimension table and the fact table.

SSIS

Deploy SSIS for all tables as ETL procedures. Therefore, we build extract, transform and load in control tasks. After connecting the sql server and database, start the program. All of ETL processes will execute automatically.

SSAS

A SSAS project is created to finish data analysis works. In the project, a multi-dimensional cube with dimensions DimStore, DimStaff, DimCustomer, DimOrder, DimInventory, DimDate, DimGame and DimPlatform is created. The figure shows the structure of the Data Source View.

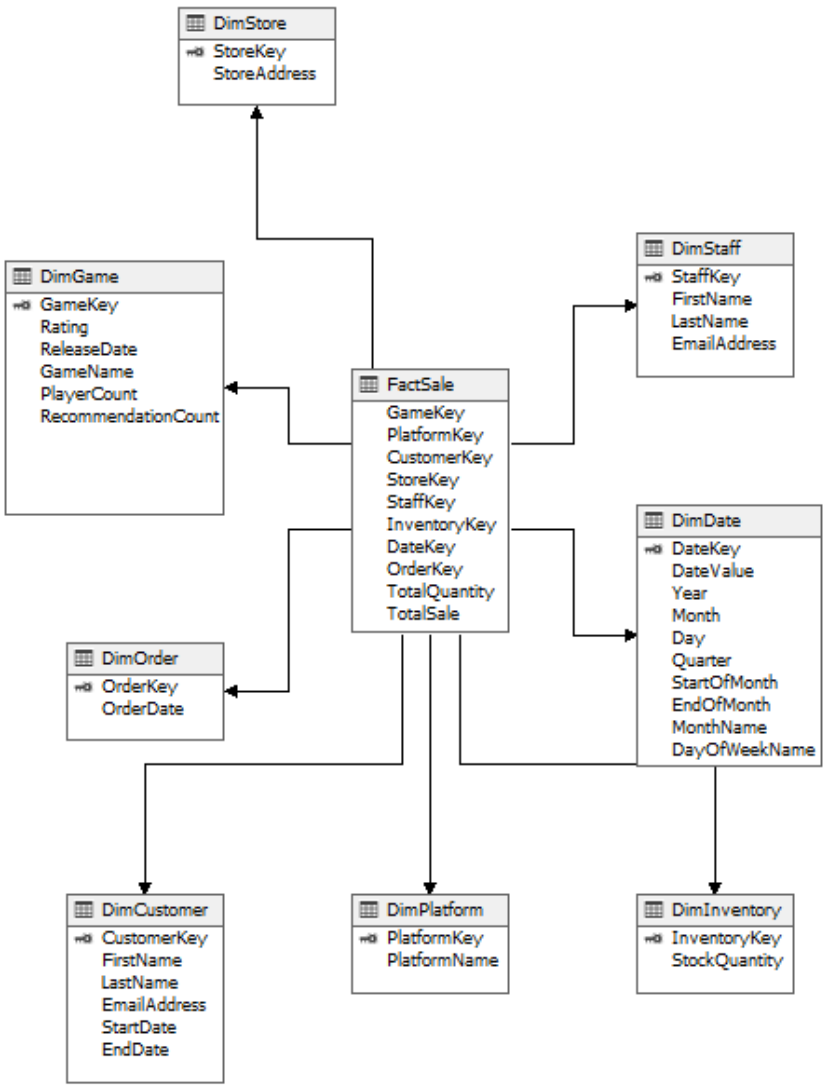
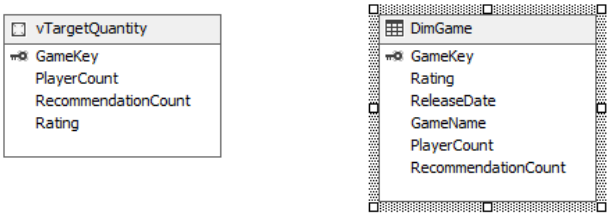


Fig. Structure of Data Source View

In the multi-dimensional cube, each dimension has a hierarchy with important attributes. For the fact table FactSale, there is one measure group Fact Sale and seven measures Max Total Sale, Min Total Sale, Max Total Quantity, Min Total Quantity, Total Quantity, Total Sale and Fact Sale Count in it. In addition, two calculations named AVG Sale and AVG Quantity are calculated by the quantity, sale and Fact Sale Count. Moreover, the project contained a MDX query that queries the total quantity with feature Game Name and Platform Name.

SSAS-Apply machine learning algorithm to the selected dataset

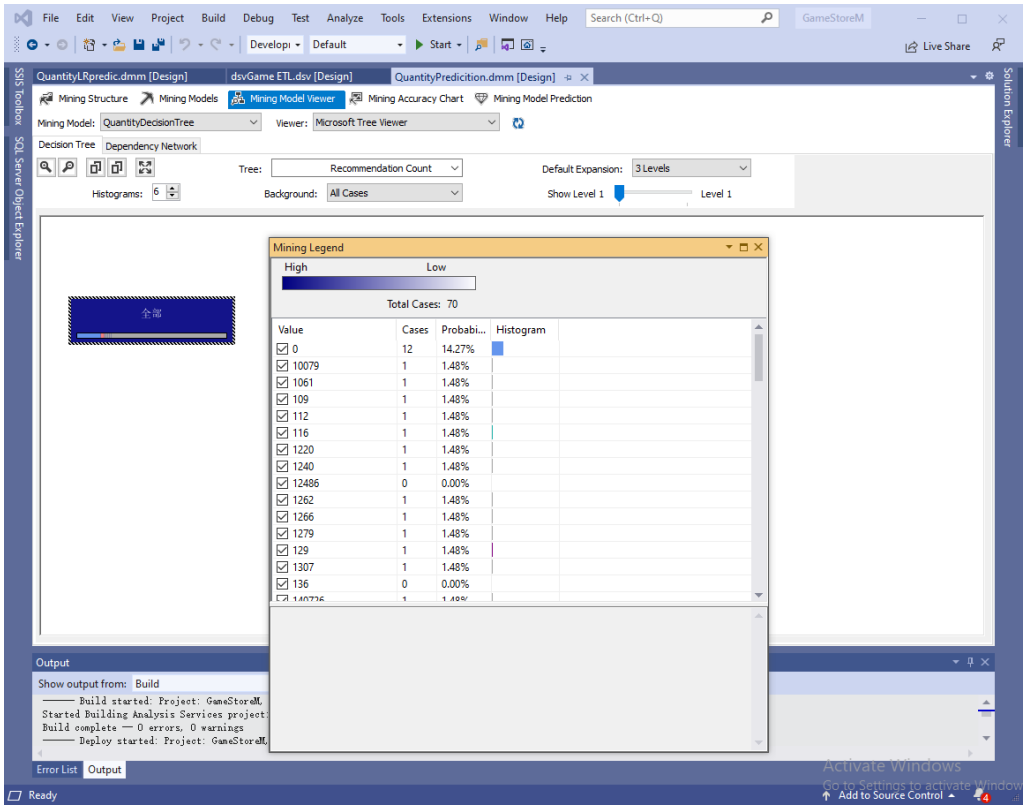
We build a model to predict the total recommendation count of each individual game by player count and rating. Therefore, build a view which contains total recommendation count as output, player count and rating as inputs. That information is from the dimGame table.



Decision tree is a popular data mining algorithm, used to predict discrete and continuous variables. The results are comparatively easy to understand, which is a reason the algorithm is so popular. If you predict continuous variables, you get a piecewise multiple linear regression formula with a separate formula in each node of a tree. The algorithm uses the discrete input variables to split the tree into nodes. A tree that predicts continuous variables is a Regression Tree. This algorithm can also predict discrete outputs.

Linear Regression predicts continuous variables only, using a single multiple linear regression formula. The input variables must be continuous as well. Linear Regression is a simple case of a Regression Tree, but it is a tree with no splits.

We first try to use the decision tree as a Microsoft machine learning algorithm, and the result is shown as the following figure.



Bonus:

The accuracy score given by the decision tree model is 0.09, and the lift char is used for validating the data mining model.

It plots the actual plot and ideal model which contains an accuracy of 100%, as the number of cases increases, the model we have created tends to become more accurate and more cases there are.

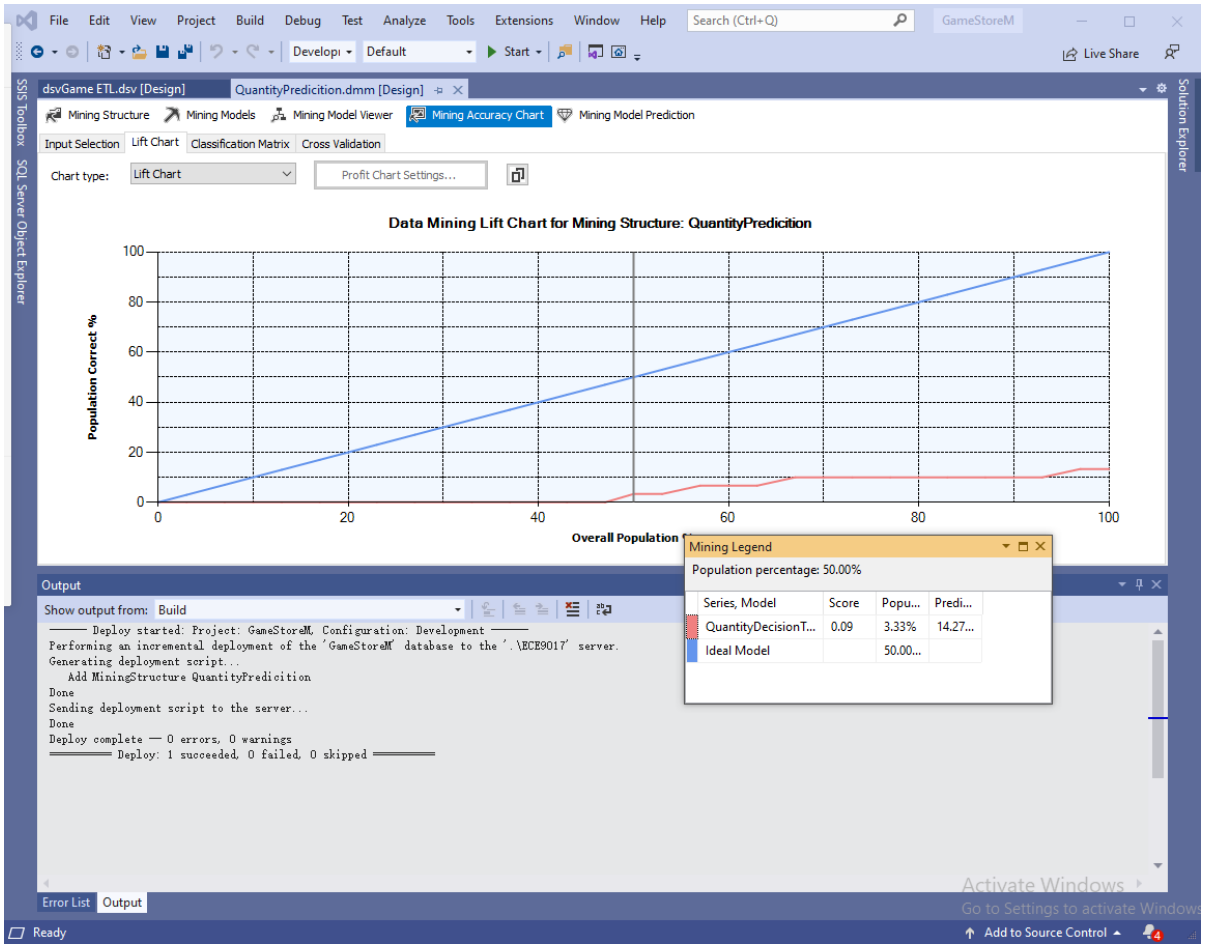


Fig. Decision Tree Result

Considering that we are making predictions for continuous variables, we change to apply Linear Regression algorithm and give a try. The following scatter plot shows that most predicted values are higher than actual values, but we can see that there is a positive relationship between predicted values and actual values.

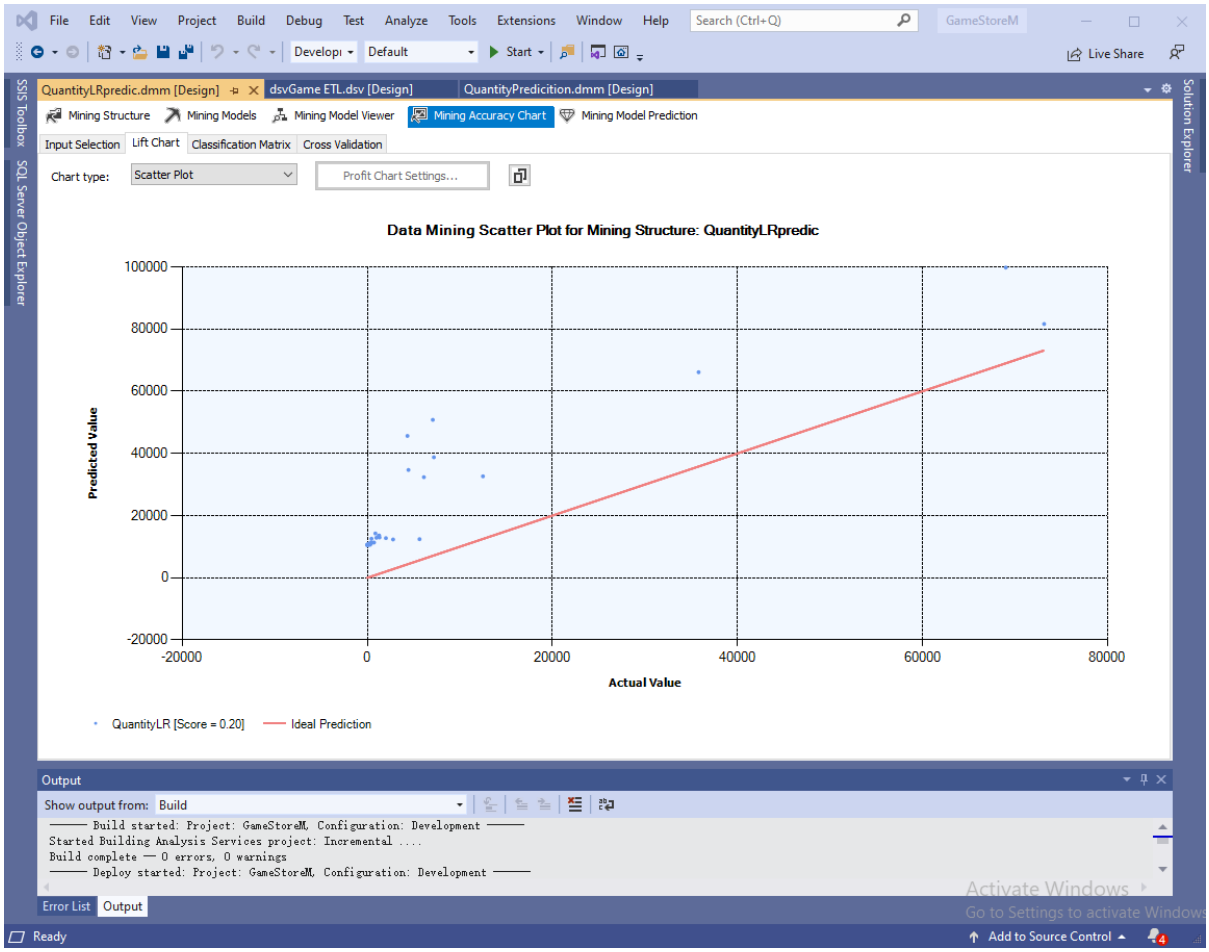


Fig. Linear Regression Result