PREDICTION ALGORITHMS ON BLACK FRIDAY SALES

ABSTRACT

This project examines and uses the Machine Learning algorithms to build a precise model to predict the purchase amount of customer paid for various products which will help retail stores to better understand the customer purchase behaviour, analyses the customers' favourite products based on their purchases and creates a personalized offer for each customer against their preferred products.

The dataset used for the project is about customer purchases obtained from a shopping mall during one of its busiest days: Black Friday. This data is used to develop the Machine Learning models and Deep Learning Neural Networks and make predictions on customer purchases.

INTRODUCTION

Every day consumers make decisions on whether to buy a product. In some cases, the decision is based solely on price but in many instances the purchasing decision is more complex, and many more factors might be considered before the final commitment is made. To make purchasing more likely, in addition to considering the asking price, companies frequently introduce additional elements to the offer which are aimed at increasing the perceived value of the purchase.

In order to keep your customers satisfied you need to provide them with the product they want when they want it. This advantage of forecasting in business will help predict product demand so that enough product is available to fulfil customer orders with short lead times, on-time. The importance of Demand Forecasting is much higher in Made-to-Stock (MTO), Assemble-to-Order (ATO) or JIT Supply Business. In some businesses, multiple promotions running concurrently may result in the cannibalisation of both promoted and non-promoted SKUs. Integrating distributor-level promotions and related forecasts will allow you to improve the flow of goods and achieve better results in terms of availability and stock fill rates. Similarly, improving the ability to forecast the impact price changes will have on both revenue and gross margin dollars, when timed well!

LITERATURE REVIEW

One of the most common financial decisions that each of us makes on a nearly daily basis involves the purchasing of various products, goods, and services. In some cases, the decision on whether to make a purchase is based largely on price but in many instances the purchasing decision is more complex, with many more considerations affecting the decision-making process before the final commitment is made. Retailers understand this well and attempt to make use of it to gain an edge in a highly competitive market.

Our aim for this project is to use Data Science, Machine Learning and Deep Learning Neural Network to build a model to predict the purchase amount of customer against various products which will help retail stores to better understand the customer purchase behaviour and to create personalized offer for customers against different products. Data-driven prediction models using ML are promising tools as they are quicker to develop with minimal inputs.

SYSTEM ARCHITECTURE

OVERVIEW:

System design is the solution to the creation of a new system. This phase is composed of several systems. This phase focuses on the detailed implementation of the feasible system. It emphasis on translating design specifications to performance specification. In system design first the administrator must give the username and password to the students so that they can login. A model is a simplification of reality. A model provides the blueprints of a system. A model may be structural, emphasizing the organization of the system, or it may be behavioural, emphasizing the dynamics of the system. Use case diagram shows a set of use cases and actors (a special kind of class) and their relationships. We use case diagrams to illustrate the static view of a system. Use case diagrams are especially important in organizing and modelling the behaviours of a system. Activity diagrams address the dynamic view of a system. They are especially important in modelling the function of a system and emphasize the flow of control among objects. A class diagram shows a set of classes, interfaces, collaborations, and their relationships. Class diagrams that include active classes address the static process view of a system.

INTRODUCTION TO UML:

The Unified Modeling Language (UML) is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems, as well as for business modelling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems. The UML is a very important part of developing object-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects. Using the UML helps project teams communicate, explore potential designs, and validate the architectural design of the software.

Goals of UML:

The primary goals in the design of the UML were:

• Provide users with a ready-to-use, expressive visual modelling language so they can develop and exchange meaningful models.

• Provide extensibility and specialization mechanisms to extend the core concepts.

• Be independent of programming languages and development processes.

• Provide a formal basis for understanding the modeling language.

• Encourage the growth of the OO tools market.

• Support higher-level development concepts such as collaborations, frameworks, patterns and components.

Why Use UML?

As the strategic value of software increases for many companies, the industry looks for techniques to automate the production of software and to improve quality and reduce cost and time-to-market. These techniques include component technology, visual programming, patterns and frameworks. Businesses also seek techniques to manage the complexity of systems as they increase in scope and scale. In particular, they recognize the need to solve recurring architectural problems, such as physical distribution, concurrency, replication, security, load balancing and fault tolerance. The Unified Modelling Language (UML) was designed to respond to these needs.

UML Diagrams:

UML diagram is designed to let developers and customers view a software system from a different perspective and in varying degrees of abstraction. UML diagrams commonly created in visual modeling tools include.

In its simplest form, a use case can be described as a specific way of using the system from a users' (actors') perspective. A more detailed description might characterize a use case as:

- a pattern of behaviour the system exhibits
- a sequence of related transactions performed by an actor and the system.
- delivering something of value to the actor

Use cases provide a means to:

- capture system requirements.
- communicate with the end users and domain experts.
- test the system.

Use cases are best discovered by examining the actors and defining what the actor will be able to do with the system. Since all the needs of a system typically cannot be covered in one use case, it is usual to have a collection of use cases. Together this Use case collection specifies all the ways of using the system.

A UML system is represented using five different views that describe the system from distinctly different perspective. Each view is defined by a set of diagrams, which is as follows.

User Model View:

- This view represents the system from the user's perspective.
- The analysis representation describes a usage scenario from the end-user's perspective.
- Structural model view
- In this model the data and functionality are arrived from inside the system.
- This model view models the static structures.

Behavioral Model View:

- It represents the dynamic of behavioural as parts of the system, depicting the interactions of collection between various structural elements described in the user model and structural model view.

Implementation Model View:

- In this the structural and behavioural as parts of the system are represented as they are to be built.

Environmental Model View:

- In this the structural and behavioural aspect of the environment in which the system is to be implemented are represented.

UML is specifically constructed through two different domains they are:

- UML Analysis modeling, this focuses on the user model and structural model views of the system.
- UML design modeling, which focuses on the behavioural modeling, implementation modeling and environmental model views.

USE CASE DIAGRAM:

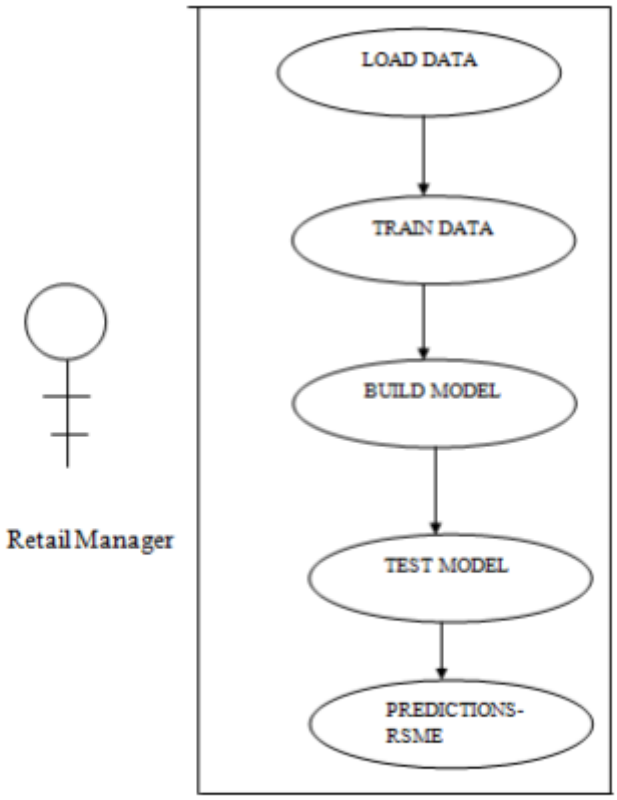

Diagram 1 User's Use Case Diagram

Naming:

A use case may have a name, although it is typically not a simple name. It is often written as an informal text description of the actors and the sequences of events between objects. Use case names often start with a verb. For example, names of possible use cases for an ATM machine might include Dispense Cash or Transfer Funds.

Relationships:

You can draw an Association relationship from a use case to an actor. You can draw a Generalize relationship between two use cases.

CLASS DIAGRAM:

A class diagram is a picture for describing generic descriptions of possible systems. Class diagrams and collaboration diagrams are alternate representations of object models. Class diagrams contain classes and object diagrams contain objects, but it is possible to mix classes and objects when dealing with various kinds of metadata, so the separation is not rigid.

Class diagrams are more prevalent than object diagrams. Normally you will build class diagrams plus occasional object diagrams illustrating complicated data structures or message passing structures.

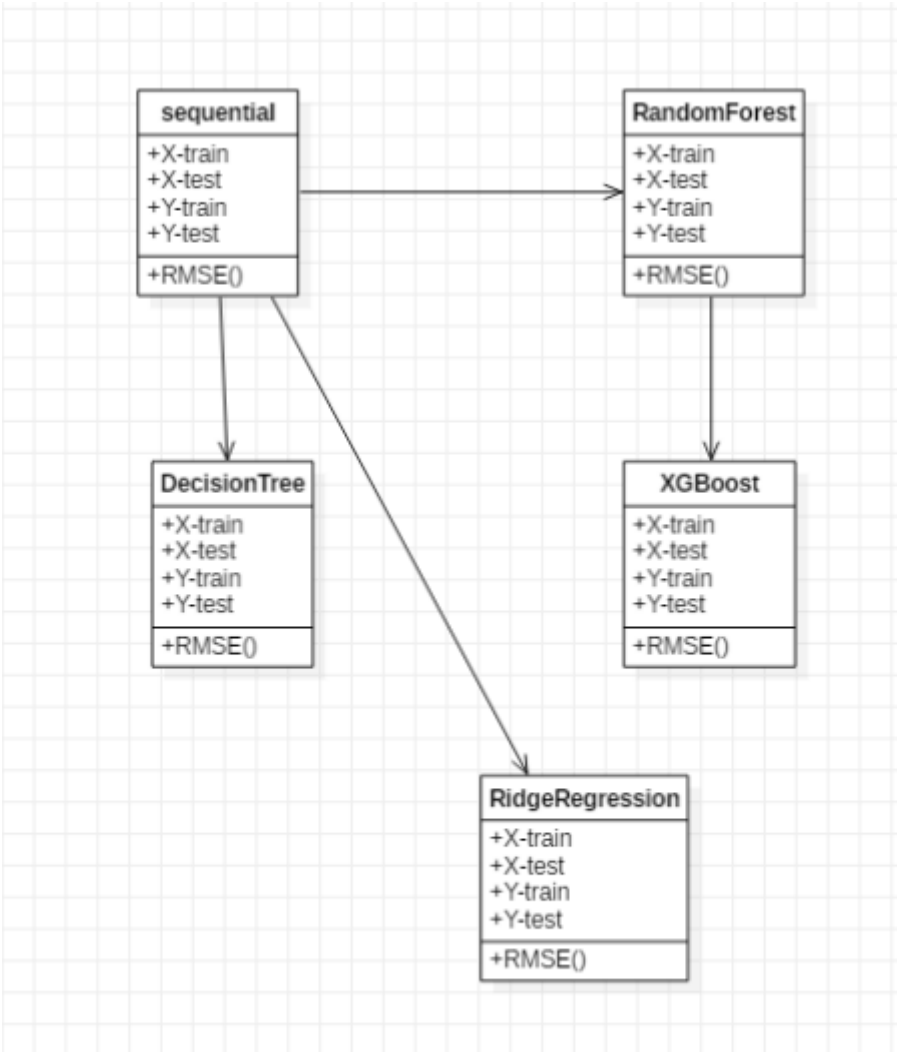

Diagram 2 Class Diagram

Class diagrams contain icons representing classes, interfaces, and their relationships. You can create one or more class diagrams to depict the classes at the top level of the current model; such class

diagrams are themselves contained by the top level of the current model. You can also create one or more class diagrams to depict classes contained by each package in your model; such class diagrams are themselves contained by the package enclosing the classes they depict; the icons representing logical packages and classes in class diagrams.

You can change properties or relationships by editing the specification or modifying the icon on the diagram. The associated diagrams or specifications are automatically updated.14 During analysis class diagram show common roles and responsibilities of the entities that provide the system's behaviour.

SEQUENCE DIAGRAM:

A sequence diagram is a graphical view of a scenario that shows object interaction in a time-based sequence, what happens first, what happens next. Sequence diagrams establish the roles of objects and help provide essential information to determine class responsibilities and interfaces. This type of diagram is best used during early analysis phases in design because they are simple and easy to comprehend. Sequence diagrams are normally associated with use cases.

Sequence diagrams are closely related to collaboration diagrams, and both are alternate representations of an interaction. There are two main differences between sequence and collaboration diagrams: sequence diagrams show time-based object interaction while collaboration diagrams show how objects associate with each other.

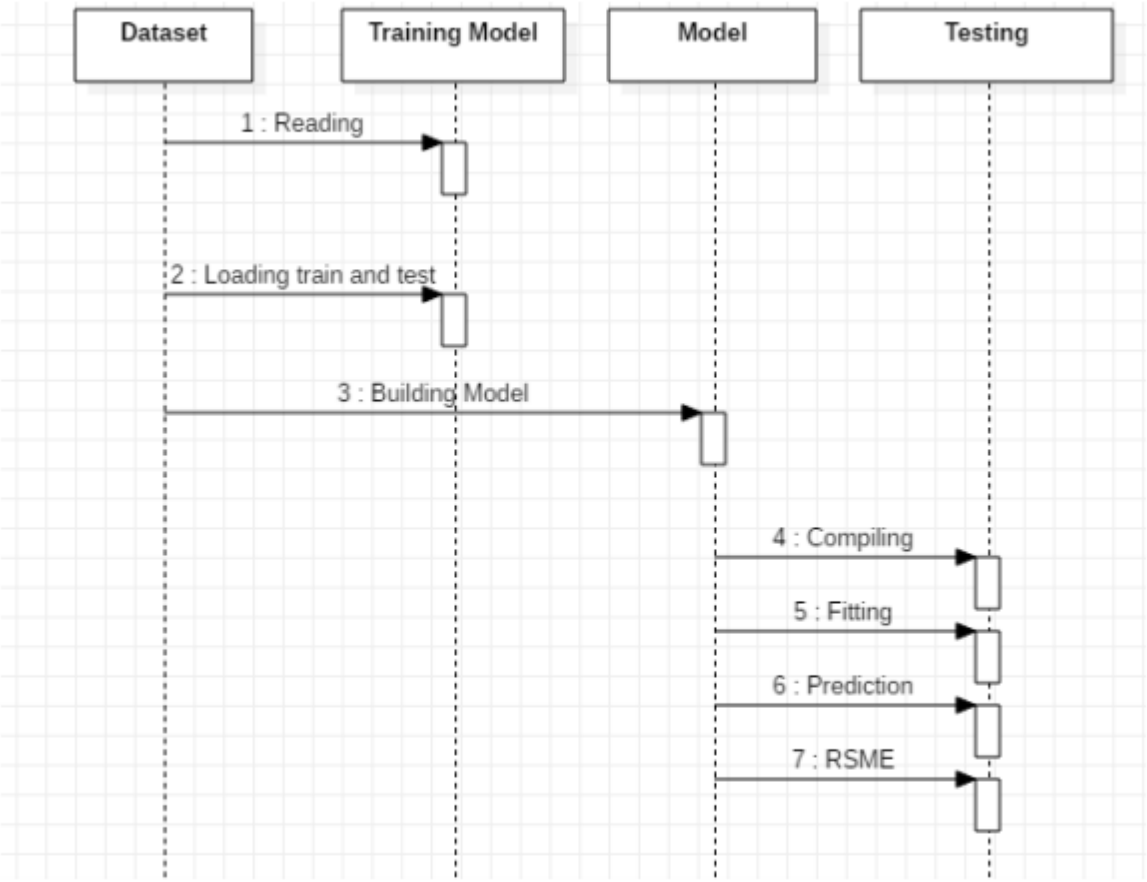

Diagram 3 Sequence Diagram

A sequence diagram has two dimensions: typically, vertical placement represents time and horizontal placement represents different objects.

The following tools located on the sequence diagram toolbox enable you to model sequence diagrams:

• Object

• Message Icons

• Focus of Control

• Message to Self

• Note Anchor

• Lifelines

ACTIVITY DIAGRAM:

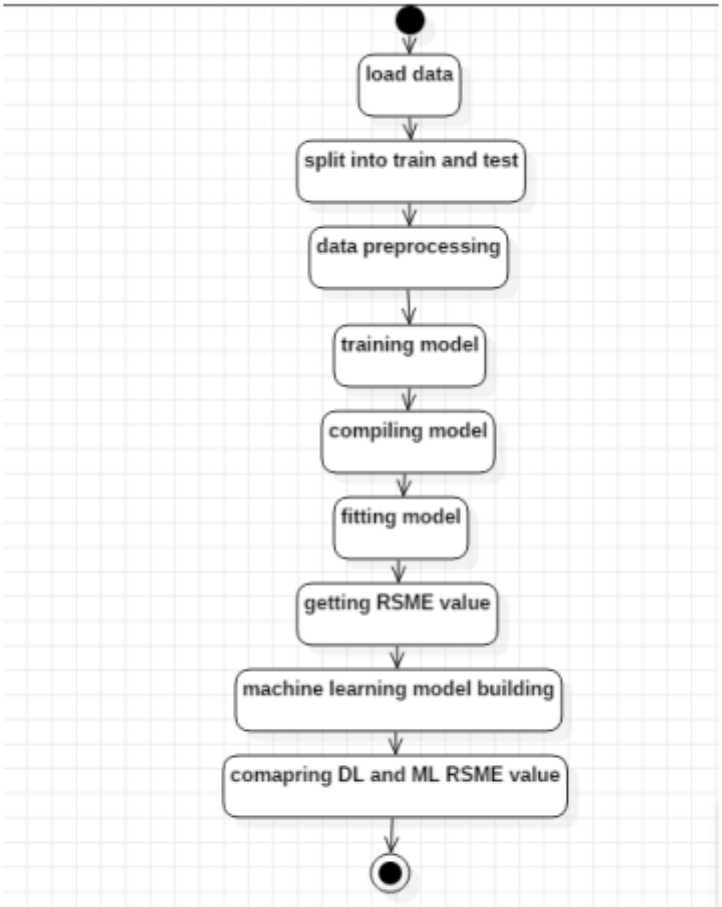

Diagram 4 Activity Diagram

COMPONENT DIAGRAM:

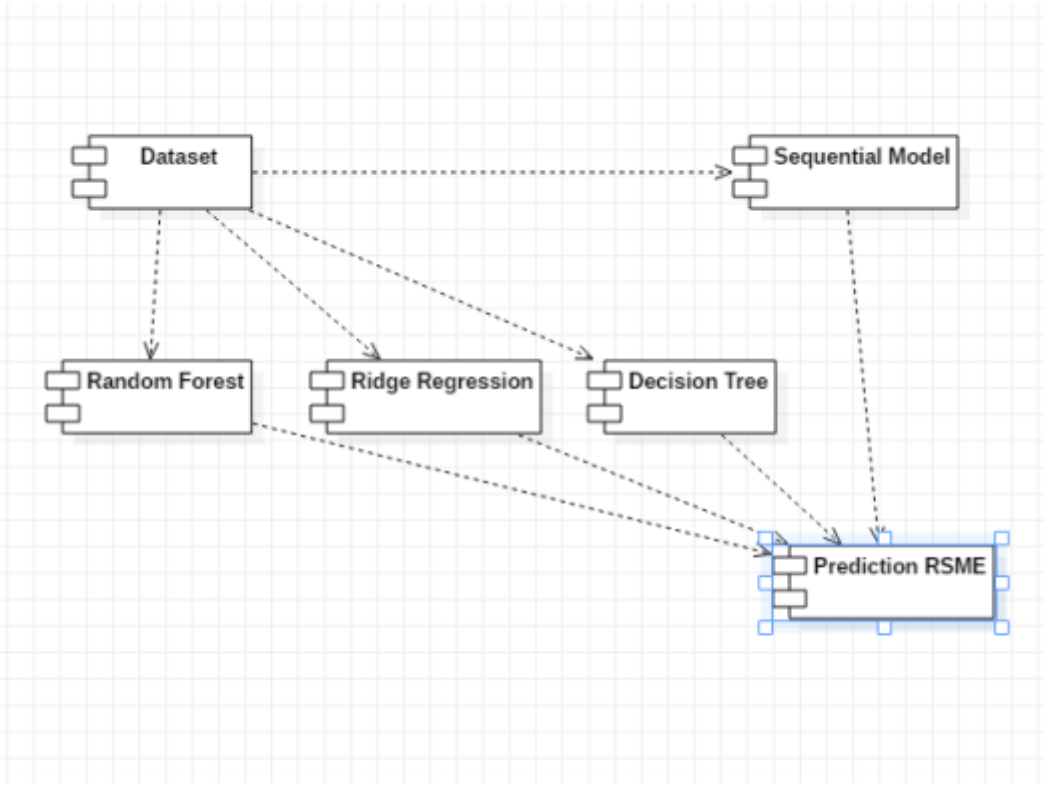

Diagram 5 Component Diagram

DEPLOYMENT DIAGRAM:

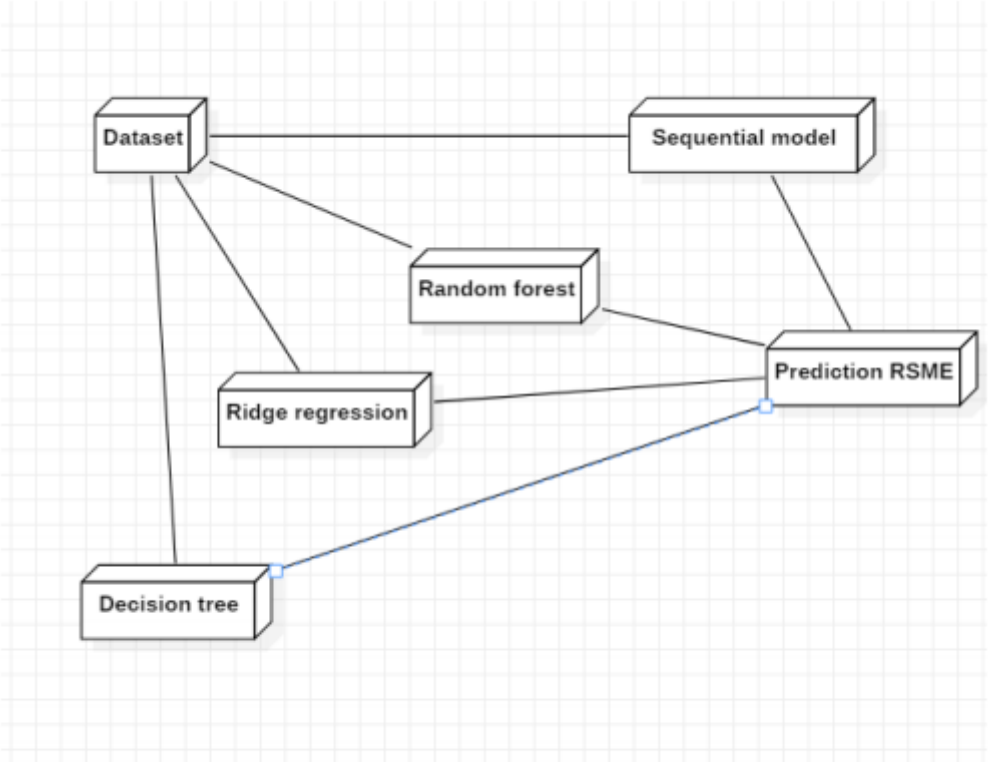

Diagram 6 Deployment Diagram

PROPOSED SYSTEM

Product recommendation is an important aspect of any sales and marketing strategy including upselling and cross-selling. Machine Learning and Deep Learning models will analyse the purchase history of a customer and based on that they identify those products from product inventory in which a customer is interested in.

The algorithms will identify hidden patterns among the items and will then group similar products into clusters. This process is known as unsupervised learning, which is a specific type of ML algorithm. Such a model will enable businesses to make better product recommendations for their customers, thereby motivating product purchase. In this way, unsupervised learning helps in creating a superior product-based recommendation system.

This project proposes to use Data Science, Machine Learning and Deep Learning Neural Network to build a precise model to predict the purchase amount of customer against various products which will help retail stores to better understand the customer purchase behaviour and to create personalized offer for customers against different products.

RESULT AND ANALYSIS

The below figure represents a snapshot of the dataset that has been used in this project.

	User_ID	Product_ID	Gender	Age	Occupation	City_Category	Stay_In_Current_City_Years	Marital_Status	Product_Category_1	Product_Category_2	Product_Category_3	Purchase
0	1000001	P00069042	F	0-17	10	A	2	0	3	NaN	NaN	8370
1	1000001	P00248942	F	0-17	10	A	2	0	1	6.0	14.0	15200
2	1000001	P00087842	F	0-17	10	A	2	0	12	NaN	NaN	1422
3	1000001	P00085442	F	0-17	10	A	2	0	12	14.0	NaN	1057
4	1000002	P00285442	M	55+	16	C	4+	0	8	NaN	NaN	7969

Figure 7.1 Snippet of Data

To understand the idea of the distribution of numerical variables and categorical variables, histograms are an excellent starting point.

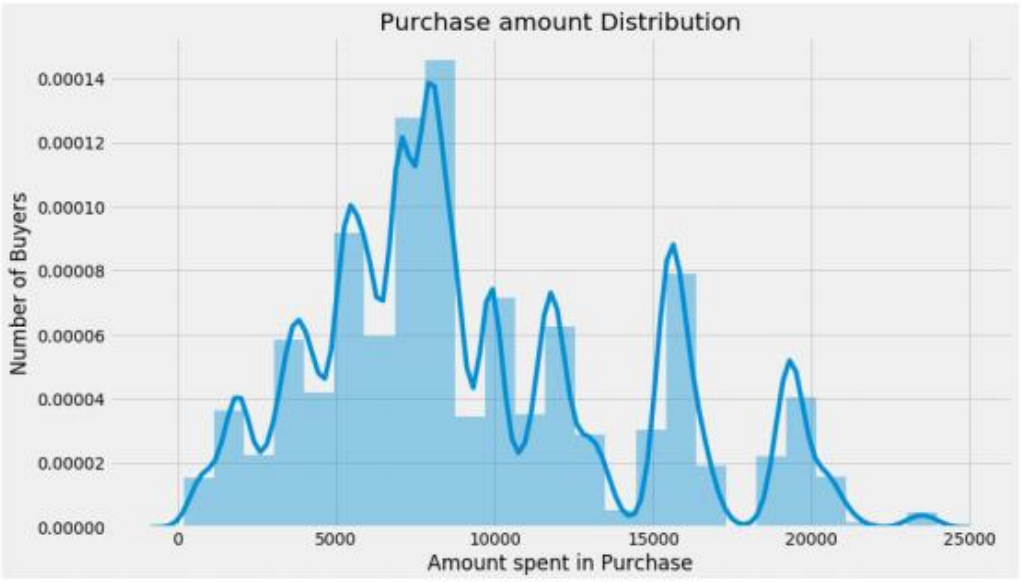

Figure 7.2 Distribution of the target variable

The below figure represents histograms (countplots) that were used to understand trends or distributions from the dataset.

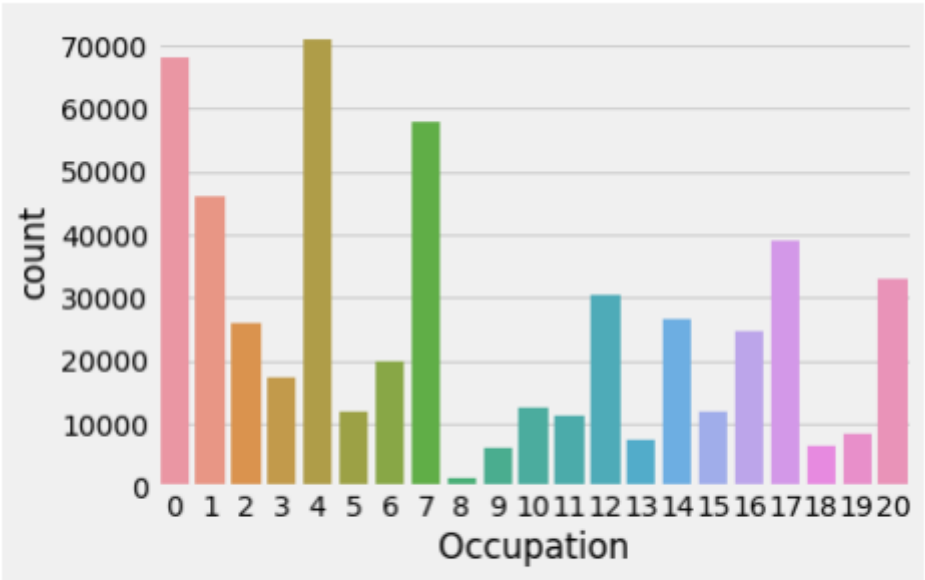

Figure 7.3 Distribution of the variable Occupation

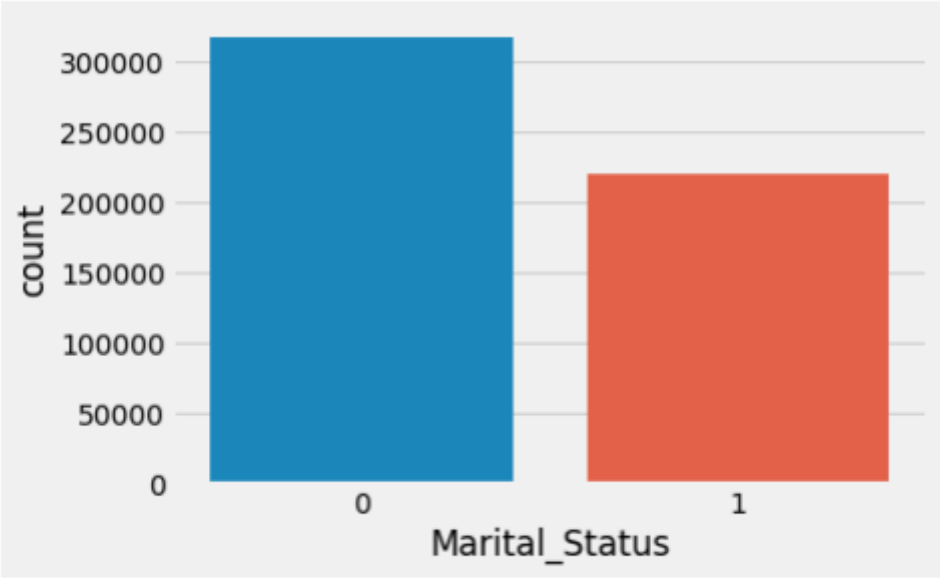

Figure 7.4 Distribution of the variable Marital_Status

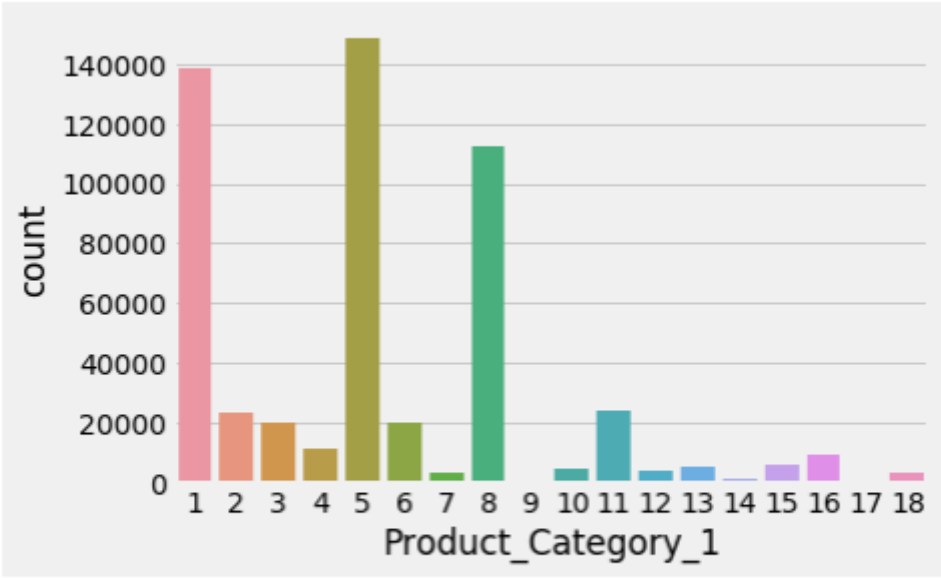

Figure 7.5 Distribution of the variable Product_Category_1

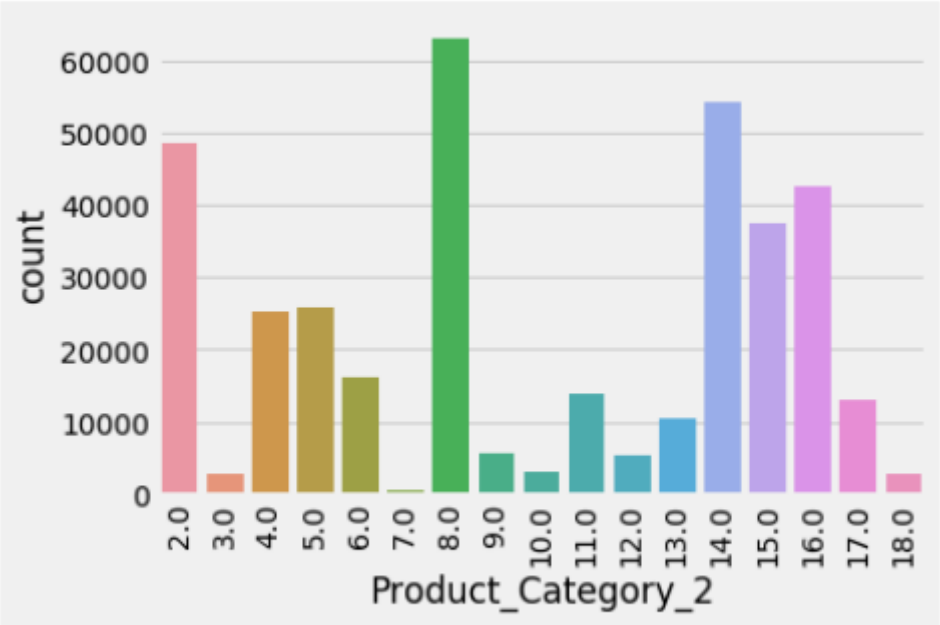

Figure 7.6 Distribution of the variable Product_Category_2

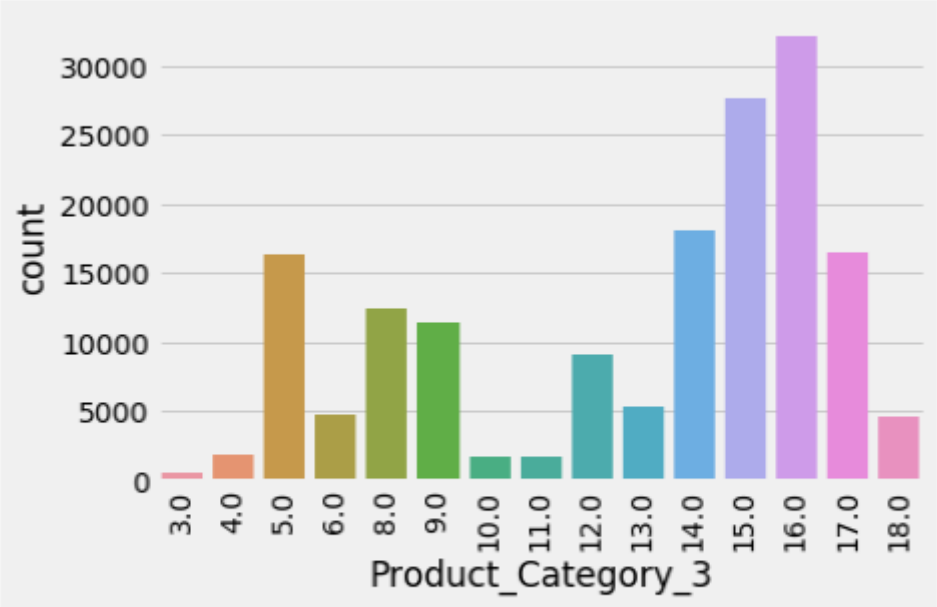

Figure 7.7 Distribution of the variable Product_Category_3

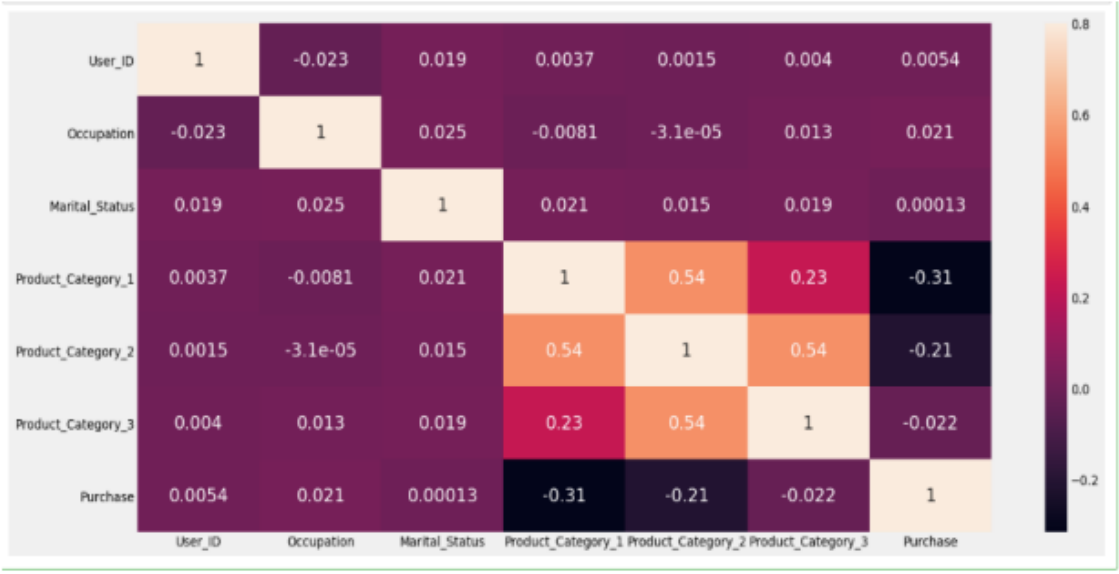

Figure 7.8 Correlation heat-map between Numerical Predictors and Target variable

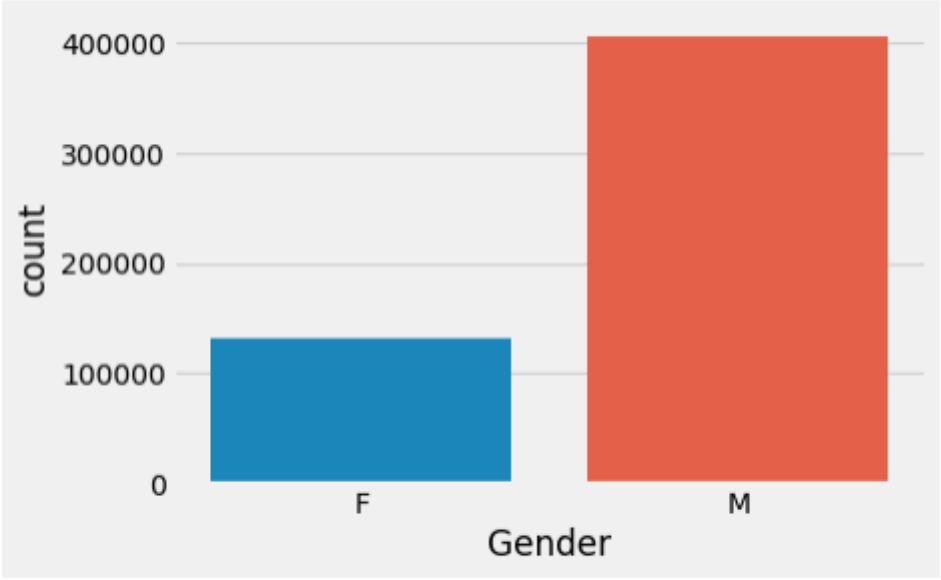

Figure 7.9 Distribution of the variable Gender

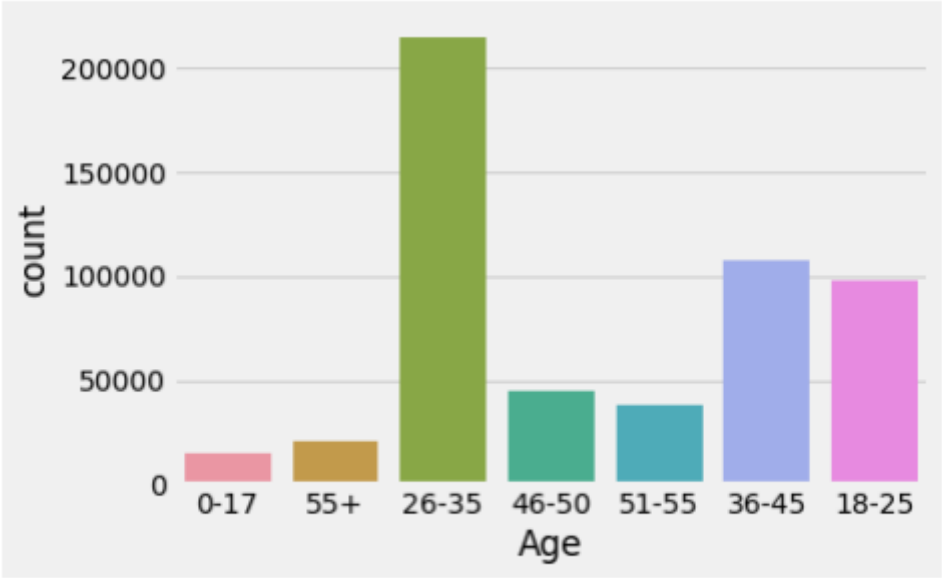

Figure 7.10 Distribution of the variable Age

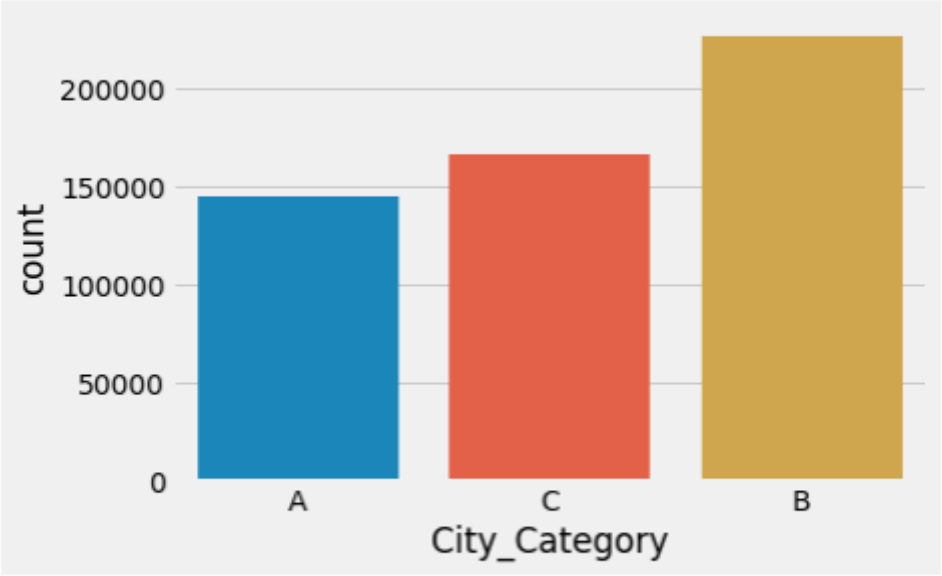

Figure 7.11 Distribution of the variable City_Category

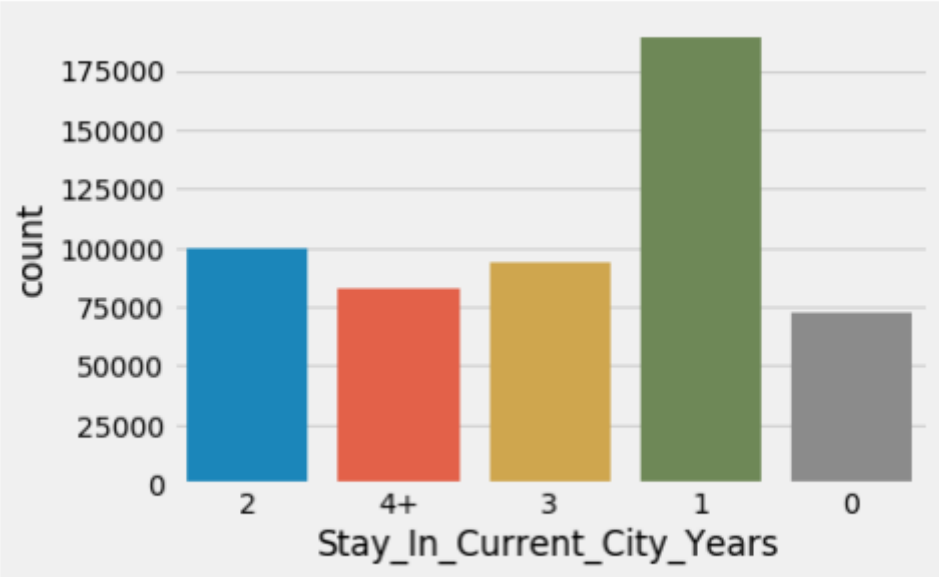

Figure 7.12 Distribution of the variable Stay_In_Current_City_Years

To understand the relationship between our target variable and predictors as well as the relationship among predictors we use bivariate analysis.

The below figure (Fig. 3 to Fig. 8) represents histograms (countplots) that were used to understand trends or distributions from the dataset.

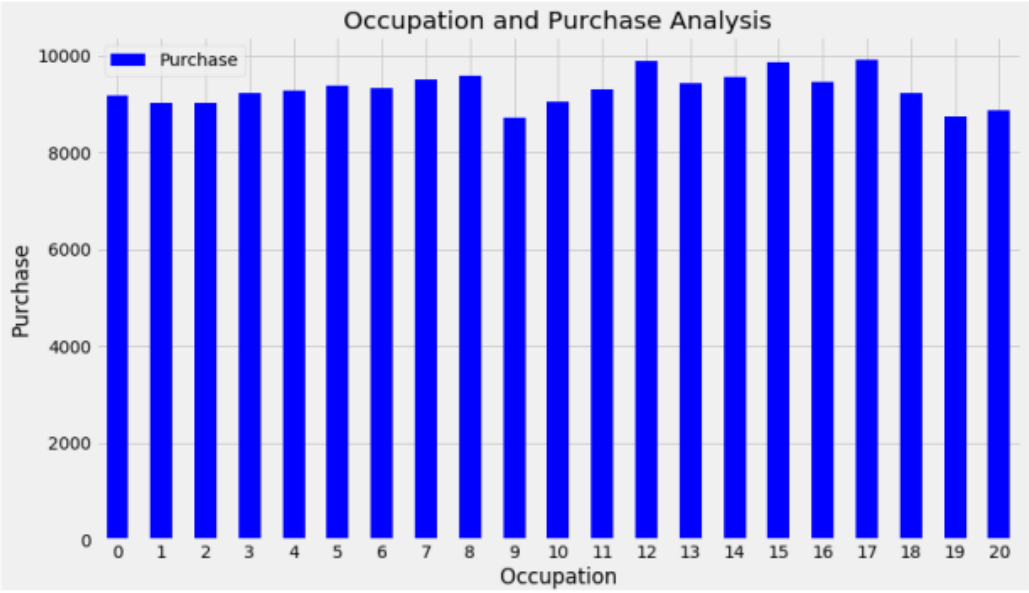

Figure 7.13 Occupation and Purchase analysis

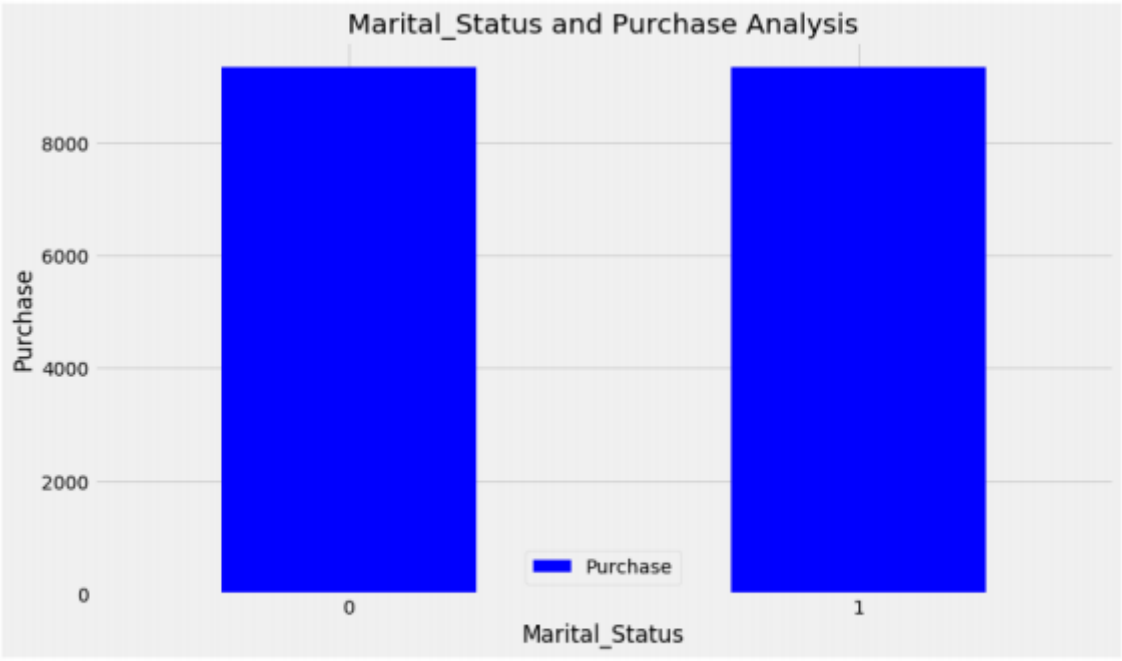

Figure 7.14 Marital_Status and Purchase analysis

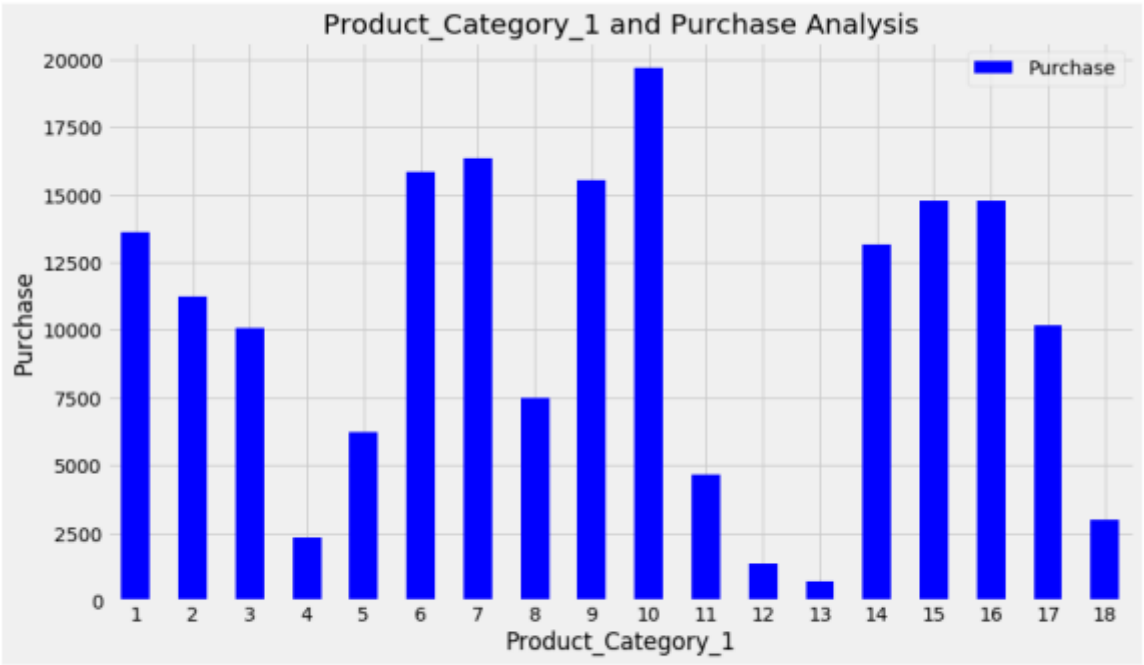

Figure 7.15 Product_category_1 and Purchase analysis

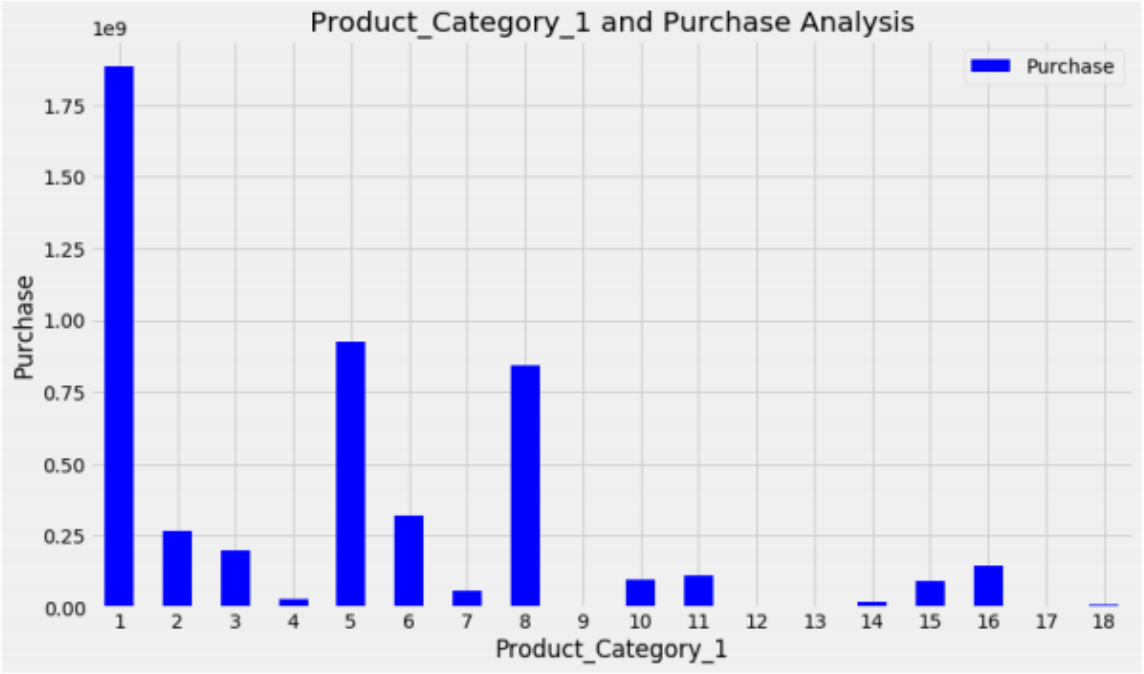

Figure 7.15.1 Product_category_1 and Purchase analysis

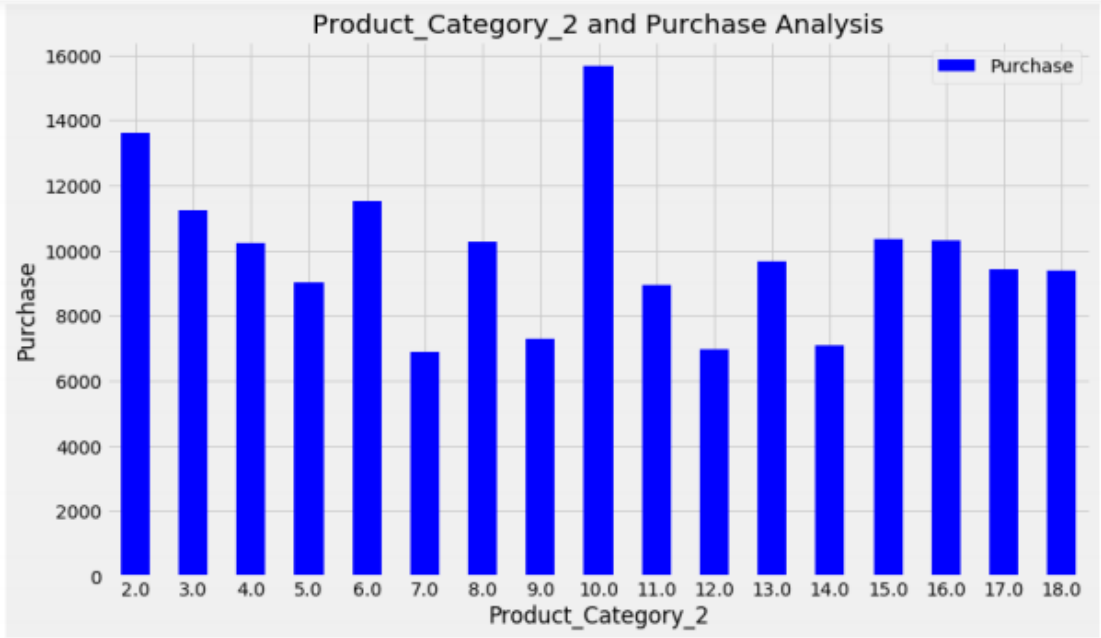

Figure 7.16 Product category 2 and Purchase analysis

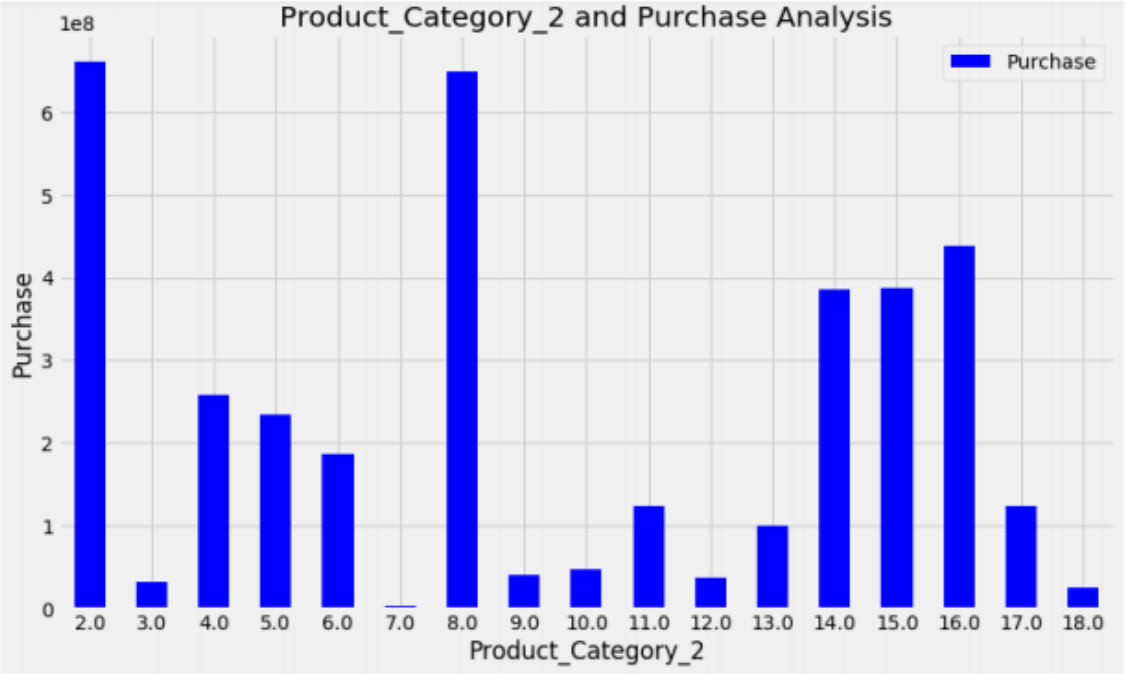

Figure 7.16.1 Product category 2 and Purchase analysis

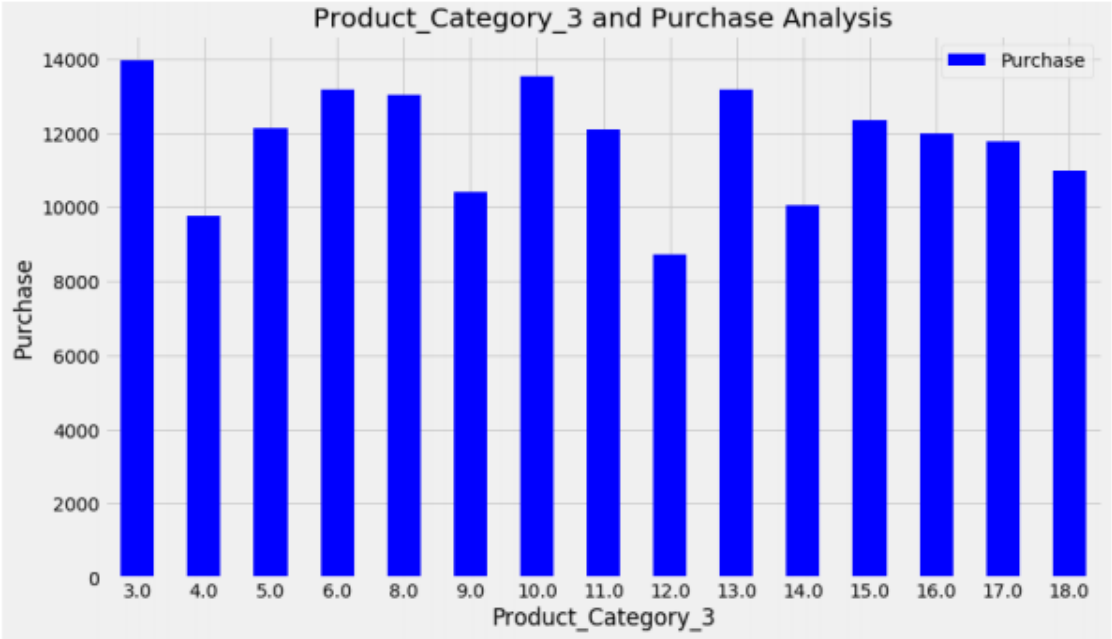

Figure 7.17 Product_category_3 and Purchase analysis

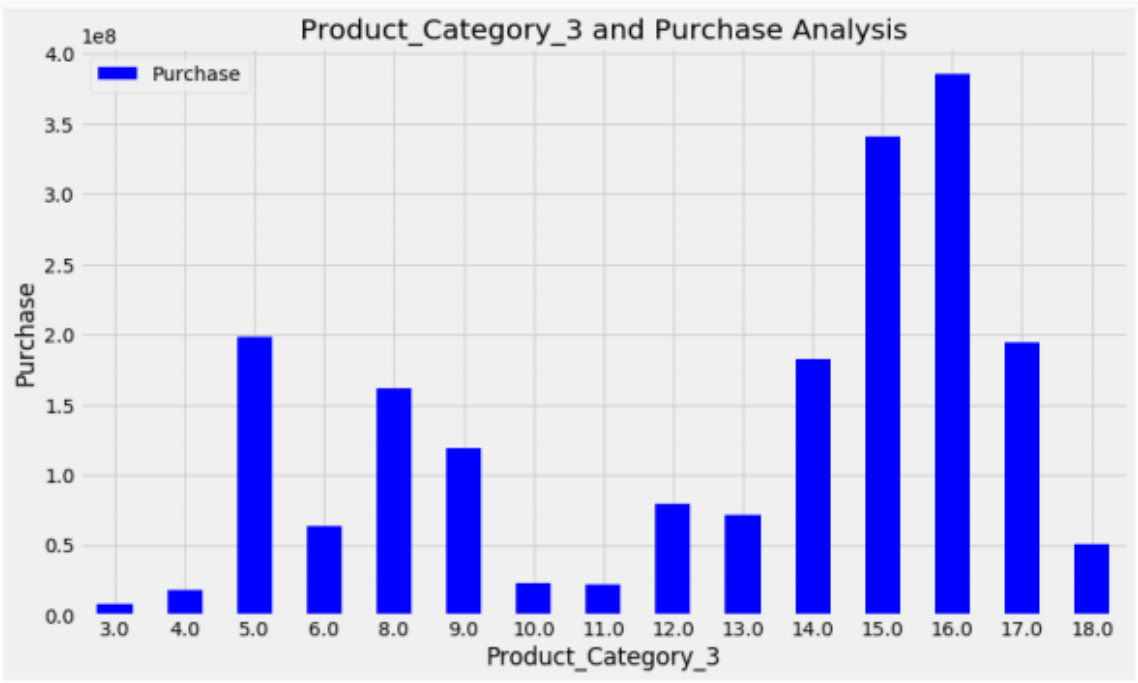

Figure 7.17.1 Product_category_3 and Purchase analysis

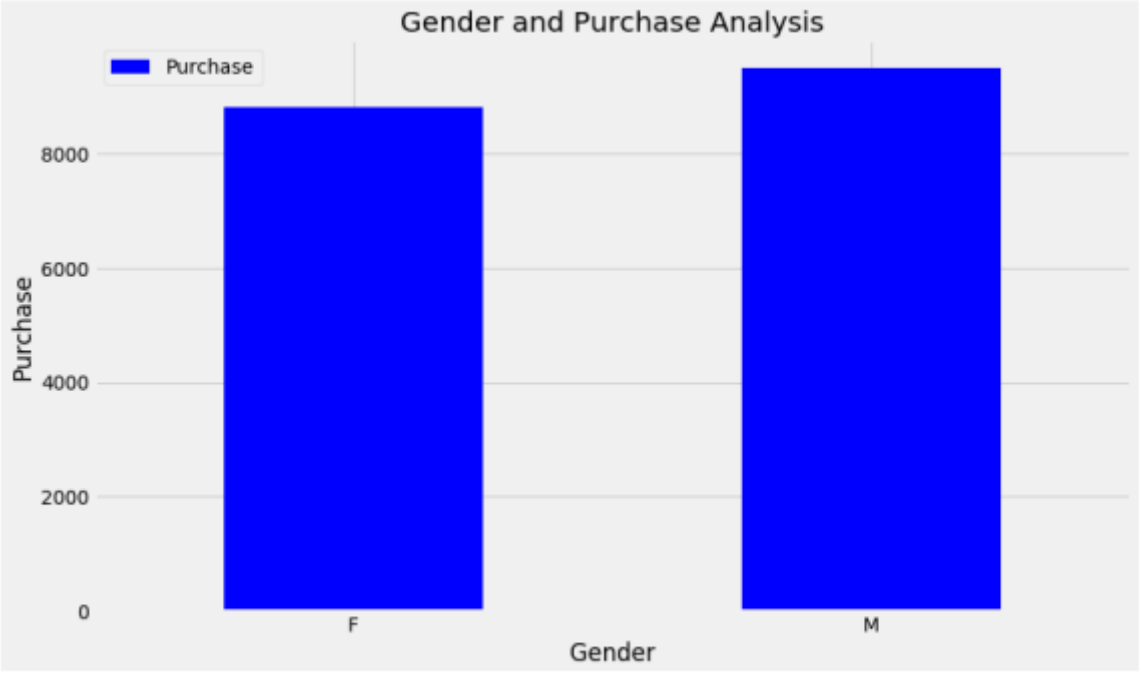

Figure 7.18 Gender and Purchase analysis

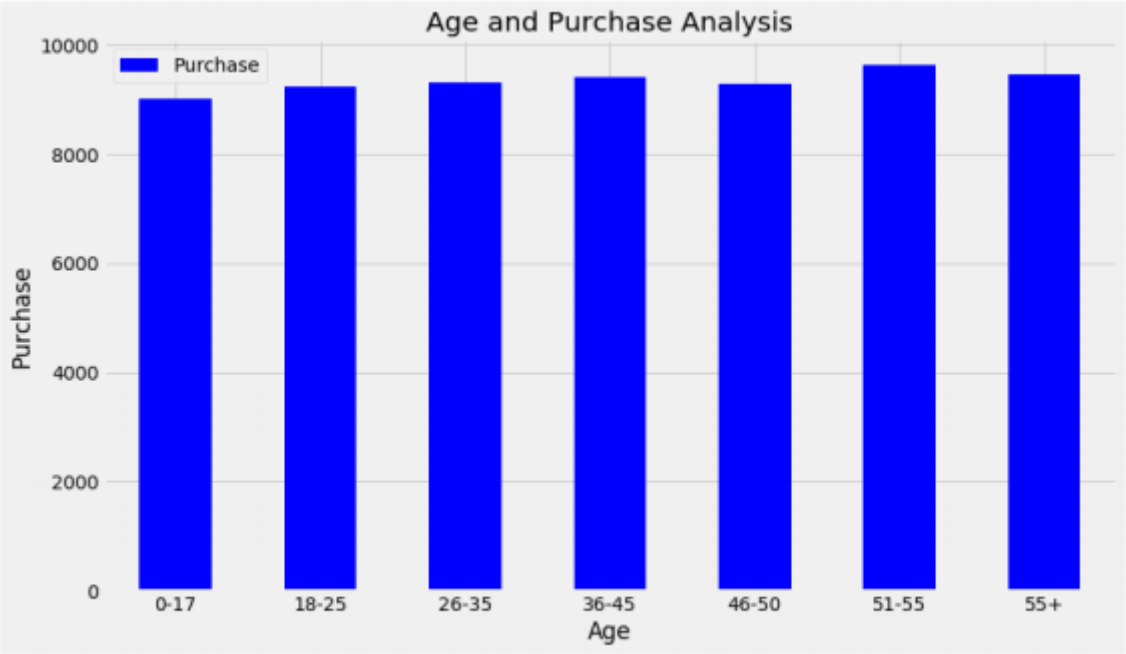

Figure 7.19 Age and Purchase analysis

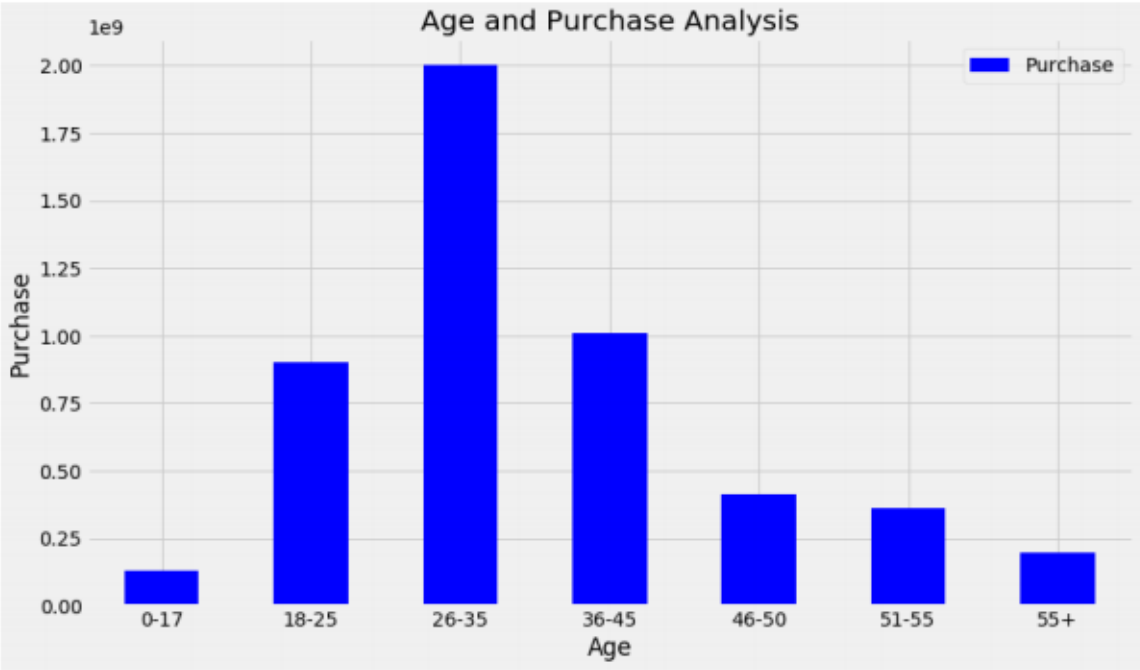

Figure 7.19.1 Age and Purchase analysis

Figure 7.20 City_Category and Purchase analysis

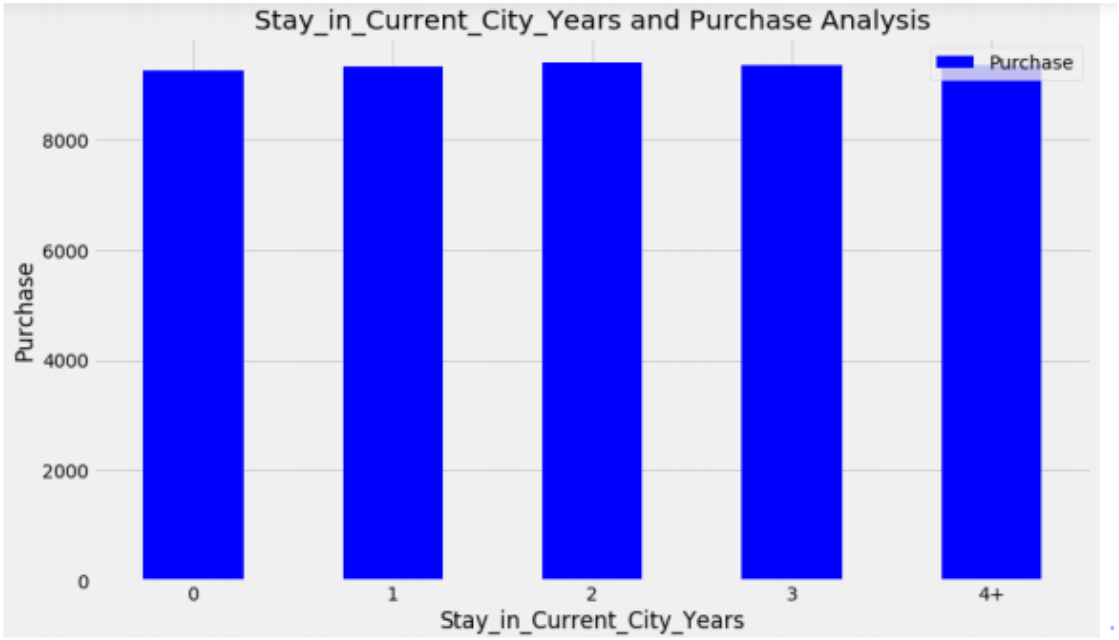

Figure 7.21 Stay_In_Current_City_Years and Purchase analysis

```
Epoch 1/10
39999/39999 [==============================] - 3s 65us/step - loss: 10978207.9054 - acc: 0.0071 - val_loss: 10338.0684 - val_
acc: 0.0109
Epoch 2/10
39999/39999 [==============================] - 2s 54us/step - loss: 2910.3492 - acc: 0.0187 - val_loss: 3934.3172 - val_acc:
0.0242
Epoch 3/10
39999/39999 [==============================] - 2s 62us/step - loss: 1155.9571 - acc: 0.0287 - val_loss: 1503.0925 - val_acc:
0.0393
Epoch 4/10
39999/39999 [==============================] - 2s 43us/step - loss: 431.7861 - acc: 0.0538 - val_loss: 504.4355 - val_acc: 0.
0800
Epoch 5/10
39999/39999 [==============================] - 2s 48us/step - loss: 152.2749 - acc: 0.0809 - val_loss: 161.6579 - val_acc: 0.
1038
Epoch 6/10
39999/39999 [==============================] - 2s 55us/step - loss: 59.2492 - acc: 0.1139 - val_loss: 58.1581 - val_acc: 0.15
```

```
RSME for Sequential Model:
3.5712462490560175
```

Figure 7.22 Sequential Model- epochs and RMSE Value

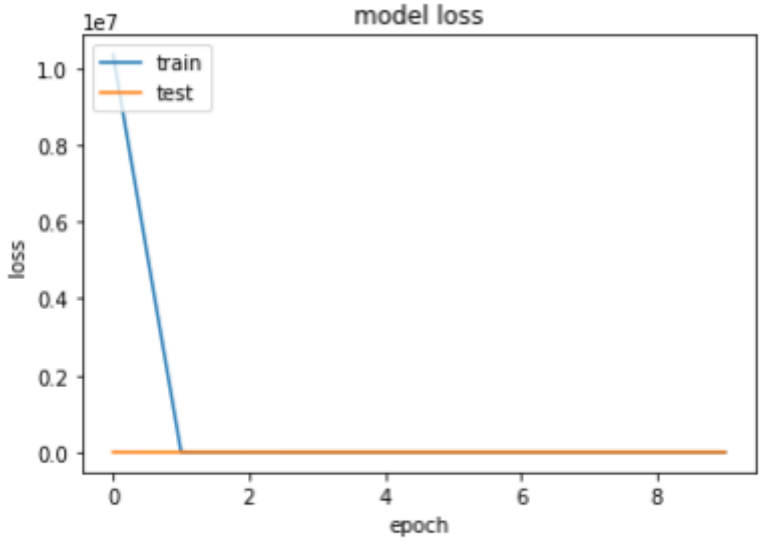

Figure 7.23 Model loss plot

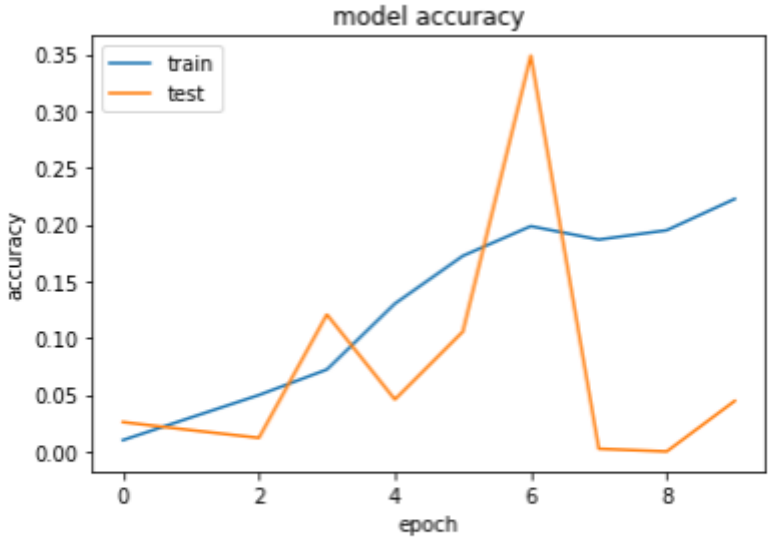

Figure 7.24 Model accuracy plot

```
Model Report
RMSE : 4593
CV Score : Mean - 4594 | Std - 99.12 | Min - 4341 | Max - 4738
```

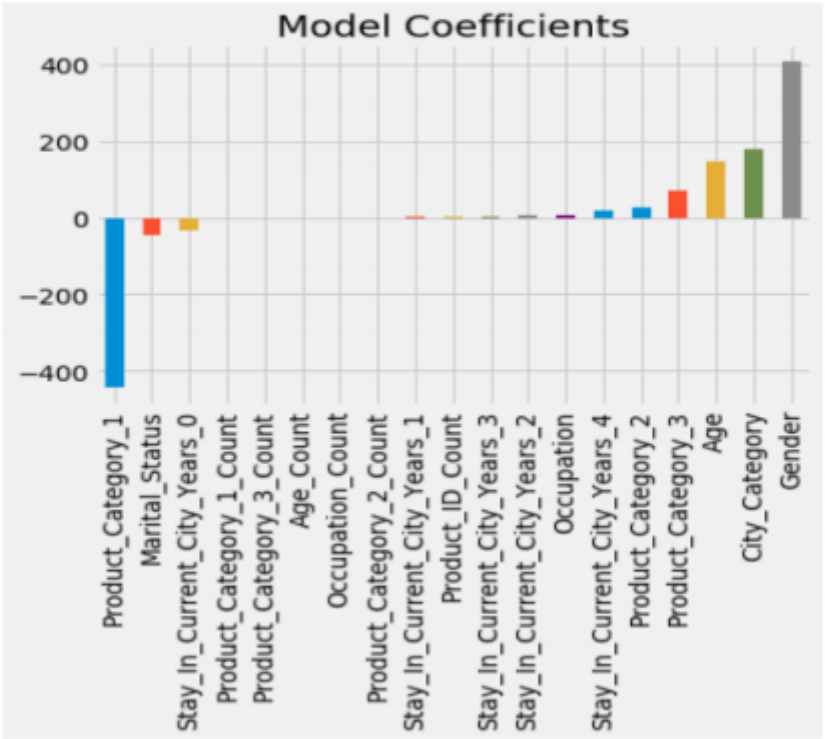

Figure 7.25 Ridge Regression Model

```
Model Report
RMSE : 3000
CV Score : Mean - 3024 | Std - 89.89 | Min - 2851 | Max - 3143
```

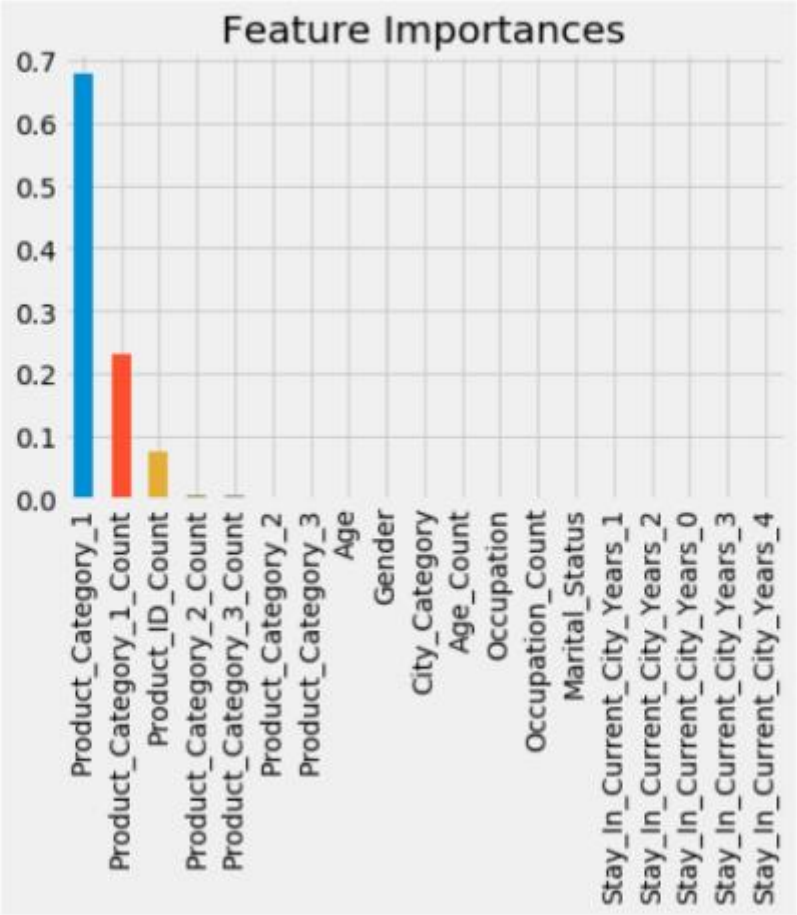

Figure 7.26 Random Forest Model

```
Model Report
RMSE : 2957
CV Score : Mean - 3022 | Std - 88.71 | Min - 2850 | Max - 3142
```

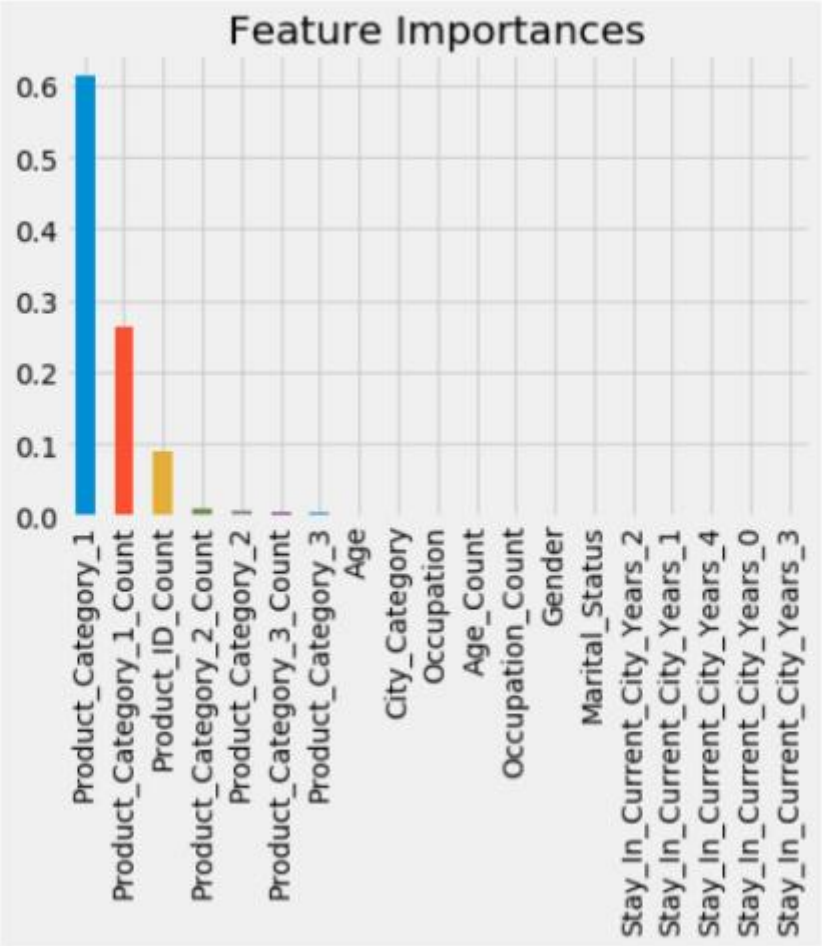

Figure 7.27 Decision Tree Model

The above figures denote the performance of both Deep Learning and Machine Learning models that were used in the comparative study. It is evident that Sequential model offered the best performance in terms of RSME Value.

CONCLUSION & FUTURE SCOPE

We have proposed and developed a system that can predict better customer purchases behaviour using Machine Learning and Deep Learning Neural Networks. It helps understand the behaviour of shoppers so that the retail managers will have better opportunity to market their products to customers effectively. We developed both Deep Learning, Machine Learning models and compared them to understand which one among these models performs better. The dataset for this project has been taken from Kaggle dataset repository. It consists of 12 features and 550 000 observations. In this project, we were able to achieve a low RSME value of ~2000 to 4000 for approximately 5MB of data. This system can be further enhanced to predict the better amount of customer purchases using more improvised Deep Learning and Machine Learning models in order to provide better products for the regular customers.