**Architecture Design**

**TRAVEL PACKAGE PURCHASE PREDICTION**

|  |  |
| --- | --- |
| Written by | Ubaid Ahmad Khanday |
| Version | 1.0 |
| Date | 06-02-2024 |

**Index**

|  |  |
| --- | --- |
| **Content** | **Page No** |
| Abstract | 3 |
| 1. Introduction | 3 |
| 1.1 What is Architecture Design? | 3 |
| 1.2 Scope | 3 |
| 1.3 Constraints | 3 |
| 2. Technical Specification | 4 |
| 2.1 Dataset | 4 |
| 2.2 Logging | 7 |
| 2.3 New Feature Generation | 7 |
| 2.4 Deployment | 7 |
| 3. Technology Stack | 8 |
| 4. Proposed Solution | 8 |
| 5. Architecture Description | 8 |
| 6. User Input/Output Workflow | 11 |

**Abstract**

Travel Package Purchase Prediction is a critical task in the travel industry, aiming to forecast the likelihood of customers purchasing specific travel packages. Machine Learning offers a powerful approach to tackle this challenge by developing predictive models that analyze various factors influencing purchase decisions. This paper explores the application of machine learning algorithms to predict travel package purchases based on input data such as customer demographics, past purchasing behavior, and travel preferences. By leveraging statistical analysis and continuous learning from new data, these models can provide accurate predictions, enabling businesses to optimize marketing strategies, tailor offerings, and enhance overall sales performance. Using a dataset derived from customer interactions with travel packages, we employ a systematic methodology to build predictive models and evaluate their effectiveness. The findings demonstrate the potential of machine learning in guiding decision-making processes to drive sales growth and improve customer satisfaction in the travel industry.

**1. Introduction**

**1.1 What is Architecture Design?**

The goal of Architecture Design (AD) or a low-level design document is to give the internal design of the actual program code for the `Travel Package Purchase Prediction System`. AD describes the class diagrams with the methods and relation between classes and program specification. It describes the modules so that the programmer can directly code the program from the document.

**1.2 Scope**

Architecture Design(AD) is a component-level design process that follows a step-by-step refinement process. This process can be used for designing data structures, required software, architecture, source code, and ultimately, performance algorithms. Overall, the data organization may be defined during requirement analysis and then refined during data design work. And the complete workflow.

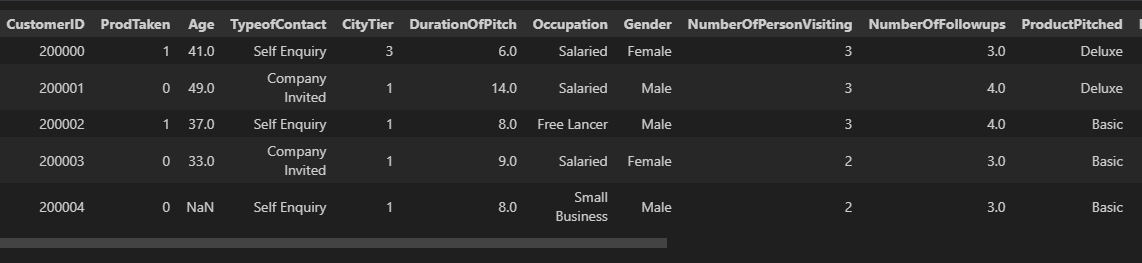
**1.3 Constraints**

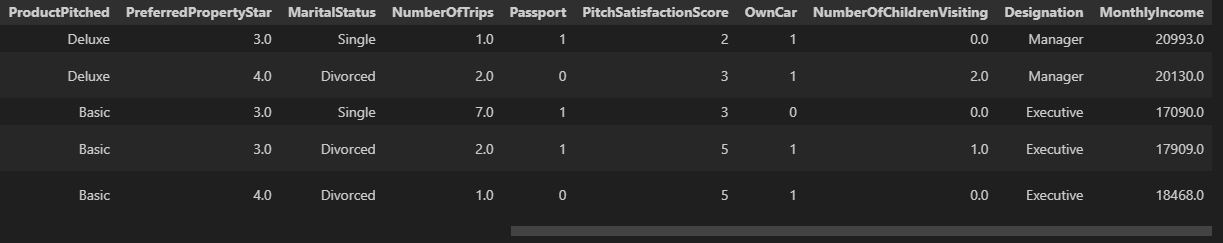
We only predict the expected casual and registered customers based on the weather condition and date information.

**2. Technical Specification**

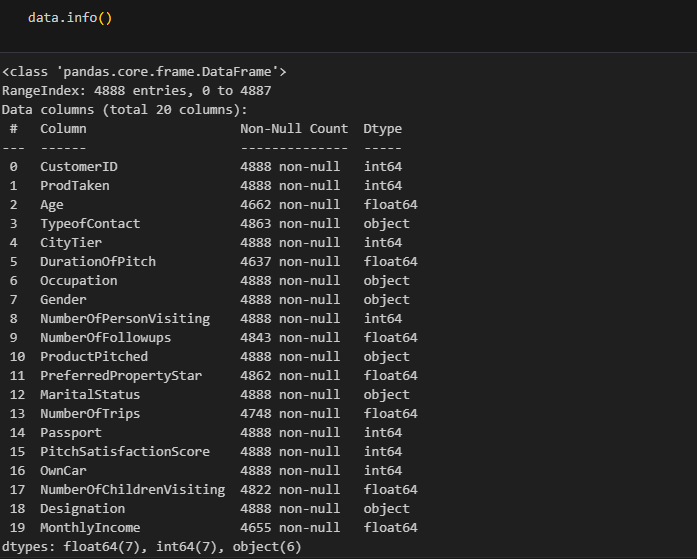
**2.1 Dataset**

The dataset for Travel Package Purchase Prediction contains 4888 observations and 20 features. Each observation represents a potential customer, while the features provide information about various aspects related to travel package purchases. The columns in the dataset are described as follows:





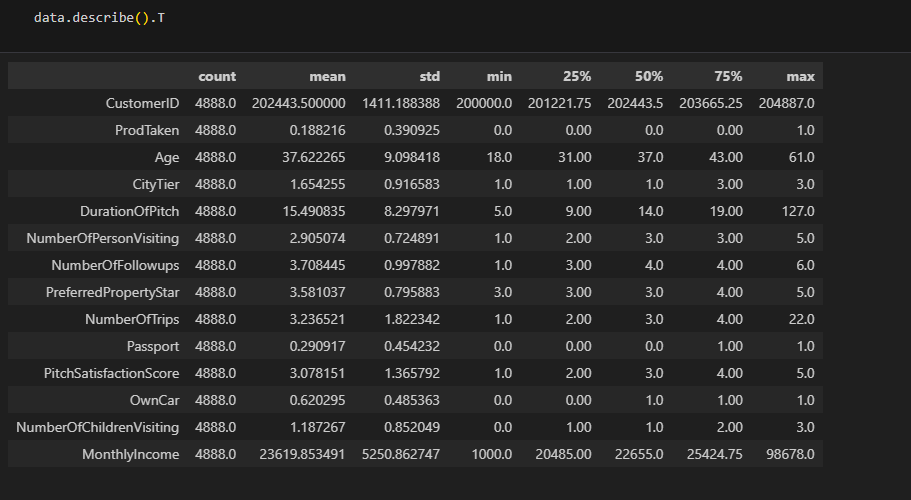
The data set consists of various data types from integer to floating to object as shown in Fig.



In the raw data, there can be various types of underlying patterns which also gives an in-depth knowledge about the subject of interest and provides insights into the problem. But caution should be observed

with respect to data as it may contain null values, or redundant values, or various types of ambiguity, which also demands pre-processing of data. The dataset should therefore be explored as much as possible.

Various factors important by statistical means like mean, standard deviation, median, count of values and maximum value, etc. are shown below for numerical attributes.



Preprocessing of this dataset includes doing analysis on the independent variables like checking for null values in each column and then replacing or filling them with supported appropriate data types so that analysis and model fitting is not hindered from their way to accuracy. Shown above are some of the representations obtained by using Pandas tools which tell about variable count for numerical columns and model values for categorical columns. Maximum and minimum values in numerical columns, along with their percentile values for median, play an important factor in deciding which value to be chosen at priority for further exploration tasks and analysis. Data types of different columns are used further in label processing and a one-hot encoding scheme during the model building.

**2.2 Logging**

We should be able to log every activity done by the user

* The system identifies at which step logging require.
* The system should be able to log each and every system flow.
* Developers can choose logging methods. Also can choose database logging.
* The system should be not be hung even after using so much logging. Logging just because we can easily debug issuing so logging is mandatory to do.

**2.3 DataBase**

The system needs to store every request into the database and we need to store it in such a way that it is easy to retain and look into the records.

The system should capture every data that any user gave and the prediction that has been made by that input.

**2.4 Deployment**

For the hosting of the project, we will use AWS ECR



**3. Technology Stack**

|  |  |
| --- | --- |
| Front End | HTML |
| Backend | Python |
| Deployment | AWS |

**4. Proposed Solution**

We performed EDA(Exploratory Data Analysis) to find the important relation between different attributes and will use a machine-learning algorithm to predict the future sales demand. The client will be filled the required feature as input and will get results through the web application. The system will get features and it will be passed into the backend where the features will be validated and preprocessed and then it will be passed to a hyperparameter tuned machine learning model to predict the final outcome.

**5** **Architecture detail**



**5.1Data Gathering**

Data source: <https://question.transtutors.com/6129343_1_tourism-data.xlsx>

Train and Test data are stored in .csv format.

**5.2 Raw Data Validation**

After data is loaded, various types of validation are required before we proceed further with any operation. Validations like checking for zero standard deviation for all the columns, checking for complete missing values in any columns, etc. These are required because The attributes which contain these are of no use. It will not play role in contributing to the sales of an item from respective outlets.

Like if any attribute is having zero standard deviation, it means that’s all the values are the same, its mean is zero. This indicates that either the sale is increasing or decrease that attribute will remain the same. Similarly, if any attribute is having full missing values, then there is no use in taking that attribute into an account for operation. It’s unnecessary increasing the chances of dimensionality curse.

**5.3 Data Transformation**

Before sending the data into the database, data transformation is required so that data are converted into such form with which it can easily insert into the database. Here, the ‘Age’,’Duration of Pitch’and “Monthly Income’ attributes contain the missing values. So they are filled in both the train set as well as the test set with supported appropriate data types.

**5.4 Data Preprocessing**

In preparation for model building, the customer data underwent thorough preprocessing. Missing values were handled based on data type and distribution (e.g., imputed with mean for numerical features, filled with mode for categorical features, or potentially removed if significant). Invalid values were corrected or removed depending on severity and impact. Outliers were identified and addressed and removed based on their influence on analysis.

Furthermore, feature scaling and normalization were applied to ensure all features operated on a similar scale, improving the effectiveness of the model building process.

**5.5 Feature Engineering**

After preprocessing it was found that some of the attributes are not important to the item sales for the particular outlet. So those attributes are removed. Even one hot encoding is also performed to convert the categorical features into numerical features.

**5.6 Pipelining**

In my project's preprocessing phase, we established separate pipelines for numerical and categorical features. The numerical pipeline handles tasks like imputation and scaling, while the categorical pipeline employs techniques like one-hot encoding. This tailored approach ensures that each feature type is appropriately processed, optimizing model performance and interpretability. By streamlining the preprocessing process, our pipelines contribute to the efficient transformation of the dataset and enhance the accuracy of our predictive models.

**5.7 Parameter Tuning**

Parameters are tuned using Grid searchCV. Many algorithms were used in this problem, logistic Regression ,Decision tree, XGBoost, SVM, AdaBoost,Random Forest etc. The parameters of these algorithms were tunned and passed into the model. We got the best accuracy from XGBoost Classifier with training accuracy of 99% and the testing accuracy of 92.5%.

**5.8 Model Building**

After doing all kinds of preprocessing operations mention above and performing scaling and hyperparameter tuning, the data set is passed into these models,

**5.9 Model Saving**

Model is saved using pickle library in `.pkl` format.

**5.10 GitHub**

The whole project directory will be pushed into the GitHub repository.Then from the git hub it will be uploaded to the cloud platform for the deployment.

**5.11 Deployment**

The cloud environment was set up and the project was deployed from GitHub into the AWS cloud platform.

App link- https://x9rs73j6e7.us-east-1.awsapprunner.com/

**6. User Input / Output Workflow.**

****