International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 05 Issue: 08 | Aug - 2021 ISSN: 2582-3930

# Using Tkinter of Python to build a GUI for Car Evaluation System

Aarushi Atrey[1], Shradha[2], Yougal Joshi[3], Lalita Luthra[4]

[1, 2, 3]Dr. Akhilesh Das Gupta Institute of Technology & Management (INDIA)

## Abstract:

Cars are vital in everyday life. They play an important role as they provide us with a mostly hassle-free mode of transportation. Every car is different in terms of price, features and the level of luxury it provides. Buying a car is a major decision on the part of the buyer as they have to take several parameters into consideration. Manufacturing and business are interested to know the popular features on which buyers make their decision. Data mining algorithms can be employed in this respect. The purpose of this research is to employ three different, but vastly popular, algorithms for evaluating the dataset. This research focuses on comparing the speed, accuracy and performance of these algorithms.

## Keywords:

Car acceptability, Support Vector Machine, K-Nearest Neighbors, Artificial Neural Network, Data Mining

## 1. Introduction:

Python is being widely used to create scripts which cover different necessities in computational scenario. Tkinter offers the basic components necessary to build a GUI that helps users to quickly inform a set of parameters defining which device to use, its configuration to set, among others, and easily start or stop operations.

## Why Python?

- Widely used in many different applications, including synchrotron laboratories.

- Getting started with programming a GUI (Graphical User Interface) in Python is not much of a difficult task.

- Python has a diverse range of options for GUI frameworks.

- All these GUI frameworks are easy and popular, some of them being open-source.

## Why Tkinter?

- Its a standard GUI (Graphical User Interface) package of Python.

- Facility to build interfaces and large number of examples and tutorials on the internet about it.

- Widgets, geometry management and event handling are the 3 main Tkinter concepts and they meet most needs.

The automotive industry is a major industrial and economic force worldwide. It makes 60 million cars and trucks a year, and they are responsible for almost half the world's consumption of oil. The industry employs 4 million people directly, and many more indirectly.

The car valuation process can be used before buying any car. It helps car buyers to know the original value and price of any car before buying it. It is a beneficial tool and helps to quickly know the market price of any vehicle. By using this tool, the car owners can estimate the true value of any car according to its make, model, mileage and condition. The seller cannot manipulate the buyer about the prices and this would help buyer to negotiate the price before making the final decision. In this way, by using **car valuation** the car buyers can get done with a profitable deal.

Data mining involves exploring and analyzing large blocks of information to glean meaningful patterns and trends. It can be used in a variety of ways, such as database marketing, credit risk management, fraud detection, spam Email filtering, or even to discern the sentiment or opinion of users. The data mining process breaks down into five steps. First, organizations collect data and load it into their data warehouses. Next, they store and manage the data, either on in-house servers or the cloud. Business analysts, management teams and information technology professionals access the data and determine how they want to organize it. Then, application software sorts the data based on the user's results, and finally, the end-user presents the data in an easy-to-share format, such as a graph or table.

Differentiating a decent car from an average to a terrible one is generally done physically with assistance of our neighbourly mechanics who instructs us to purchase the car. It would be nice to have a way of finding out the acceptability of the car. The focus of this research work is to compare three influential algorithms; K-Nearest Neighbours (KNN), Support Vector Machine (SVM) and Artificial Neural Network (ANN) in term of speed and accuracy they depict on the data set of cars.

## 2. Related Work:

One crucial step in data mining projects is to find an efficient classification algorithm so that results can be trusted upon. It also depends on the experimental design of the system. If the selection of algorithm is not made thoughtfully the outcomes of data mining task could be compromised, consequently resulting in invalid conclusions. Researchers have focused on this point and have compared various algorithms in terms of accuracy and speed. This section presents a brief overview of the essential work done in this domain.

In research work conducted by S. Makki [1] backpropagation neural network (BNN) and naïve Bayesian classifier (NB) has been employed for data mining classification for evaluation on car data set. These two algorithms are tested on data set and results show that BNN is much more accurate as compared to NB although it is difficult to implement and it runs slow. In [2] author presents a comparative study on multiple prediction algorithm for analyzing breast cancer survivability. In experiments a large data set with 10-fold cross validation has been used. Results demonstrate that decision tree is the most accurate, artificial neural networks takes third place and logistic regression method is the last in terms of accurate results. R. Russo [3] in his work has applied machine learning algorithm to dataset which describes movie. The basic aim is to create a movie recommender system for movie watchers. Neural networks, NB, simple rule classifiers and decision tree are compared. Results indicate that NB and neural networks perform better in terms of evaluating given dataset. In [4] author proposes a methodology to evaluate an adaptive tourist service of onboard cars. The system evaluated provides personalized information to tourist on cars. In the research work layered sampling strategy is employed and system suggestions to users are compared for accuracy. S. Singh [5] evaluates the performance of different classification methods. Three algorithm are studied in this research; K-Nearest Neighbors, Support Vector Machines (SVM) and Artificial Neural Networks (ANN). The results demonstrate that SVM and ANN are better predictors.

## 3. Tkinter:

Tkinter is the most commonly used library for developing GUI (Graphical User Interface) in Python. Tkinter or “Tk interface”, is a module of python that provides an interface to Tk GUI toolkit, developed in TCL (Tool Command Language) and multiplatform, with support for Linux, MAC OS and MS Windows.

Widgets, geometry management and event handling are the 3 main concepts of Tk, which also apply for Tkinter.

### 3.1 Widgets:

Widgets in Tkinter are the elements of GUI application which provides various controls to users to interact with the application. Some examples are frames, labels, buttons, text entries, checkboxes, tree views, scroll bars, and text areas.

Inside a Python script, widgets are objects, or instances of classes that represent mentioned window components. To instantiate an object on Tk, and then on Tkinter, is necessary to indicate its parent, that maintain a window hierarchy between all elements. On that hierarchy, the main window is the root. Each widget has a set of configuration options which control how they are displayed or how they behave, like a “text” option for components that display some text, as a label, or “command” option when they accept events, at the mouse click of a button.

### 3.2 Geometry Management:

An important step of interface design is to organize the widgets onscreen window. The most useful method to do that using Tk, or Tkinter, is by geometry manager, like “grid”. In practice, “grid()” is a method available to all supported widgets saying to them where exactly to be positioned in an invisible matrix of columns and rows.

Combination of nested frames and grid is the better approach to design a Tk/Tkinter interface.

### 3.3 Event Handling:

Tk/Tkinter manages the event loop that receives user actions over the window components, controlled by oper-ating system, like button presses, keystrokes, mouse movement, and window resizing.

Individual widgets know how to respond to events. Ba-sically, it provides a callback that can be assigned to a procedure in Python code as a configuration, like “com-mand” for button widgets. For events without a callback command associated with them, it is possible to use an event binding, which in practice is the use of “bind()” method on a widget to capture any event and then execute an arbitrary procedure or method.

Another important method available for widgets is “af-ter()”. Using it is possible to create an execution thread forked from the main application loop. At this new thread, a set of Python instructions is performed in paral-lel with interface updating. Besides, widget that calls it continues to be responsive to any user input.

## 4. Dataset Description:

The dataset used in this research is a collection of the records on specific attributes of cars. The dataset is created by Marko Bohanec and donated by Marko Bohanec and Blaz Zupan in 1997. We obtained the dataset from the UCI dataset repository. The car dataset, as described on the UCI repository was derived from a simple heirarchical decision model originally developed for the demonstration of DEX, M. Bohanec, V. Rajkovic: Expert system for decision making.

| | | | |
|---|---|---|---|
| **Data Set Characteristics:** | Multivariate | **Number of Instances:** | 1728 |
| **Attribute Characteristics:** | Categorical | **Number of Attributes:** | 6 |
| **Associated Tasks:** | Classification | **Missing Values?:** | No |

**Table 1: Car Evaluation Dataset**

The target attribute in the dataset is “class”, and it assumes the following 4 values as a combination of the deterministic features:

- Acceptable: This is denoted as ‘acc’
- Good: This is denoted as ‘good’
- Unacceptable: This is denoted as ‘unacc’
- Very Good: This is denoted as ‘vgood’

Data analysis was performed on the dataset to identify patterns within the data and present the data in tables based on the rage of the attributes and its frequencies.

| **Class** | **Frequency** | **Relative Frequency in %** |
|---|---|---|
| **Acc** | 385 | 22.28 |
| **Good** | 70 | 4.05 |
| **Unacc** | 1207 | 69.85 |
| **Vgood** | 66 | 3.82 |
| **Total** | 1728 | 100 |

**Table 2: Frequency of ‘class’ output in dataset**

Table 2 shows the frequency of the class output which is the final outcome from the dataset. It shows that more than half of the cars evaluated were not acceptable. And the frequency of the cars falling under category ‘vgood’ was the least when compared to others.

## 5. Experiments and Results:

This section aims at demonstrating the experimental setup and the results obtained. It describes the basic working of algorithm, how data is prepared for testing and what results claim about all three of the data mining algorithms.

## 5.1 Classification Method:

The classification methods applied in this research are K-Nearest Neighbours (KNN), Support Vector Machine (SVM) and Artificial Neural Networks (ANN). The k-nearest neighbours algorithm (k-NN) is a non-parametric method proposed by Thomas Cover used for classification and regression. In both cases, the input consists of the k closest training examples in the feature space. In k-NN classification, the output is a class membership. An object is classified by a plurality vote of its neighbours, with the object being assigned to the class most common among its k nearest neighbours.

Support Vector Machines (SVMs) are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. Developed at AT&T Bell Laboratories by Vapnik with colleagues (Boser et al., 1992, Guyon et al., 1993, Vapnik et al., 1997), it presents one of the most robust prediction methods, based on the statistical learning framework or VC theory proposed by Vapnik and Chervonenkis (1974) and Vapnik (1982, 1995). Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier.

Artificial neural networks (ANNs), usually simply called neural networks (NNs), are computing systems vaguely inspired by the biological neural networks that constitute animal brains. An ANN is based on a collection of connected units or nodes called artificial neurons, which loosely model the neurons in a biological brain. Each connection, like the synapses in a biological brain, can transmit a signal to other neurons. An artificial neuron that receives a signal then processes it and can signal neurons connected to it. The "signal" at a connection is a real number, and the output of each neuron is computed by some non-linear function of the sum of its inputs.

## 5.2 Data Cleaning:

Data cleaning is the process of preparing data for analysis by removing or modifying data that is incorrect, incomplete, irrelevant, duplicated, or improperly formatted. This data is usually not necessary or helpful when it comes to analyzing data because it may hinder the process or provide inaccurate results. There are several methods for cleaning data depending on how it is stored along with the answers being sought. Data cleaning is not simply about erasing information to make space for new data, but rather finding a way to maximize a data set’s accuracy without necessarily deleting information. The dataset used in this research has also been cleaned to ensure quality for model creation. We have converted nominal attributes into numeric attributes. Table 3 shows the conversion.

| Attribute | Nominal | New Numeric Value |
|---|---|---|
| Buying | vhigh | 4 |
| | High | 3 |
| | Med | 2 |
| | Low | 1 |
| Maintenance | Vhigh | 4 |
| | High | 3 |
| | Med | 2 |
| | Low | 1 |
| Luggage Boot | Small | 3 |
| | Med | 2 |
| | Big | 1 |
| Safety | Low | 1 |
| | Med | 2 |
| | High | 3 |

**Table 3: Nominal to Numeric Conversion**

## 5.3 Data Pre-processing:

Once the dataset has been chosen, raw input data should be pre-processed, otherwise it will negatively affect the results obtained. It is extremely crucial to the performance of neural network. The two basic pre-processing techniques are data transformation and normalization. Transformation manipulates raw data inputs creating a single input to the network, while normalization tends to distribute data evenly scaling it onto an adequate range. This can help network in learning process enhancing its ability to understand the association between given inputs and generated outputs.

## 5.4 Dataset Split:

The pre-processed dataset is split into two shares of varying sizes for utilizing one half as training data and second half as testing or validating network. The methodology of data splitting can have considerable influence on the performance of model. Inappropriate data splitting can result in incorrect and extremely variable performance. Classifying algorithm uses training data for learning. Training model is built by comparing the attributes of dataset with class/label. After training, the model is tested on test data which is the other half of split dataset. In this research work 3 splits are being tested.

| Training | Testing |
|---|---|
| 90% | 10% |
| 66% | 44% |
| 50% | 50% |

**Table 4: Training and Testing Dataset Split**

## 5.5 Results:

This section presents results of the experimentation setup. The process is as follows; it is supervised learning method. We have trained the model utilizing attributes inclusive of class attributes. As it is a supervised model, the model is built basing on the class values in correspondence to the values of attributes individually. The results achieved by various experimentation setup in KNN, SVM and ANN are elaborated in Table 5, 6 and 7 respectively. The tables show the percentage splits employed which are; 90:10, 70:30 and 50:50.

| Percentage Split | | Time in Seconds | | KNN | |
|---|---|---|---|---|---|
| Training% | Testing% | Build | Test | Correct% | Incorrect% |
| 90 | 10 | 0.07 | 0.01 | 89.01 | 10.99 |
| 70 | 30 | 0.01 | 0.01 | 90.17 | 9.83 |
| 50 | 50 | 0.01 | 0.02 | 88.42 | 11.58 |

**Table 5: Performance of KNN**

| Percentage Split | | Time in Seconds | | SVM | |
|---|---|---|---|---|---|
| Training% | Testing% | Build | Test | Correct% | Incorrect% |
| 90 | 10 | 0.02 | 0.05 | 94.21 | 5.79 |
| 70 | 30 | 0.00 | 0.03 | 91.13 | 8.87 |
| 50 | 50 | 0.00 | 0.04 | 83.79 | 16.21 |

**Table 6: Performance of SVM**

| Percentage Split | | Time in Seconds | | ANN | |
|---|---|---|---|---|---|
| Training% | Testing% | Build | Test | Correct% | Incorrect% |
| 90 | 10 | 7.01 | 0.00 | 90.75 | 9.25 |
| 66 | 44 | 7.19 | 0.01 | 96.24 | 3.76 |
| 50 | 50 | 6.98 | 0.02 | 81.37 | 18.63 |

**Table 7: Performance of ANN**

## 6. Conclusion:

The fundamental objective of this research work was to compare and contrast three data mining algorithms; K-Nearest Neighbours, Support Vector Machines and Artificial Neural Network in terms of accuracy they offer. Paper initiates with an introduction of the domain and talks about the previous influential work conducted. Next it leads to a detailed elaboration of the experimentation setup describing dataset, data cleaning and pre-processing. Results of all three of the algorithm are presented.

The results demonstrate that ANN provides the greatest accuracy when the train to test split ratio of the dataset is 70:30, SVM provides the greatest accuracy when the train to test split ratio of the dataset is 90:10, while KNN outperforms the other 2 when the train to test split ratio of the dataset is 50:50. Overall, the greatest accuracy is that of the Artificial Neural Network when dataset is split in a ratio of 70:30.

## References:

[1]. Makki, S., Mustapha, A., Kassim, J. M., Gharayebeh, E. H., & Alhazmi, M. (2011). Employing neural network and naive Bayesian classifier in mining data for car evaluation. In Proc. ICGST AIML-11 Conference (pp. 113-119).

[2]. Delen, D., Walker, G., & Kadam, A. (2005). Predicting breast cancer survivability: a comparison of three data mining methods. Artificial intelligence in medicine, 34(2), 113-127.

[3]. Russo, R. (2006). Bayesian and neural networks for motion picture recommendation. Technical Report, Boston COllege.

[4]. Console, L., Gena, C., & Torre, I. (2003). Evaluation of an on-vehicle adaptive tourist service. In Proceedings of the Second Workshop on Empirical Evaluation of Adaptive Systems, held at the 9th International Conference on User Modeling UM2003, Pittsburgh (pp. 51-60).

[5]. Singh, S. (2005). Modeling Performance of Different Classification Methods: Deviation from the Power Law. Project Report, Department of Computer Science, Vanderbilt University, USA (April 2005).

[6]. Bohanec, M., & Rajkovic, V. (1988). Knowledge acquisition and explanation for multi-attribute decision making. In 8th Intl Workshop on Expert Systems and their Applications (pp. 59-78).