Department of Electrical Engineering and Computer Science
EECE 401 Senior Design I
Fall 2021
Solution Design
Capital One:
Algorithmic and Visualization Capabilities for Machine Learning
Caelia Thomas, De'Johnna Wright and Joshua Whitaker
Advisor: Imtiaz Ahmed

Introduction

Capital One is an American bank that specializes in credit cards, auto loans, and savings accounts. They want to work with Howard researchers to improve their algorithmic and visualization capabilities in a way that supports machine learning-driven analysis of large and complex data sets. As of right now, they have not provided specific technical information about the data sets to our team, and they have not provided specific project information. In this light, it is possible to consider a hypothetical scenario that takes into consideration Capital one's specialization.

Given that Capital One deals with customers' finances, it is possible that they will need a visualization algorithm to aid in evaluating customer eligibility for different loans. So in a hypothetical scenario, our team would be responsible for creating an algorithm that can handle large complex sets of customer information (such as age, salary, savings.. etc). Then, the algorithm must find a nonlinear regression in order to create an effective visual representation of the customer information that aids in evaluating customer eligibility.

Section I.

Solution I: TensorFlow (Joshua Whitaker)

TensorFlow is a commonly used open-source machine learning platform that is compatible with Python. TensorFlow provides the ability to visualize nonlinear regression and is compatible with Numpy, a Python library that can work with numerical data sets.

Solution II: PyTorch (De'Johnna Wright)

PyTorch is an open source machine learning platform used for deep learning applications using GPUs and CPUs. This tensor library is compatible with Python and Numpy, and is easier to work with and offers fast computation time.

Solution III:Keras (Caelia Thomas)

Keras is a deep learning API written in Python that is mainly used for easy and fast prototyping. Its Python foundation allows for easy debugging and extendability to other projects. The user-interface has a reputation for being easy to learn and adapt to, which is helpful for new users of the future algorithm.

Section II.

For the top two solution designs TensorFlow and PyTorch were chosen. TensorFlow is an

open source machine learning platform which offers graphical visualization of nonlinear regressions. It is a well known software with a polished interface that has debugging tools such as TensorBoard, which make it easier to resolve its neural network. It is compatible with many coding languages including Python, which is the preferred language for this project. However, TensorFlows interface is more static, making it difficult to utilize.

PyTorch is an open source machine learning library developed for python programs which is used for deep learning applications. PyTorch offers computational graph support at runtime which means it has fast computation time, very efficient memory management which is important when dealing with large data sets, it is compatible with Numpy, and it has a dynamic interface and controls which supports GPU and CPU, and easy debugging using Pythons IDE. On the other hand Pytorch was released in 2016 and is newer compared to other libraries so there's not much detail on how to operate the library.

Section III.

The criteria for the decision matrix can be seen in Figure 1. As illustrated, we decided to weigh visualization capabilities the highest at 0.5 Since the primary goal of our software is to be able to provide a meaningful visual output of the regression of the data, it is crucial for the library we choose to be proficient in this task. Next, computational efficiency was weighed slightly less at 0.3. Although not as important as the libraries' visualization capabilities, it is still essential that our software is able to produce the results in a timely manner. Data handling capability was weighed at 0.2, since it is necessary that the libraries are able to handle complex data sets. Finally, the difficulty of learning the libraries was weighed at 0.1. Since we are constrained by the time that we graduate to finish the project, it would be inefficient for us to have to spend a lot of time learning how to use the libraries.

	Visualization Capabilities (0.5)	**Efficiency (0.3)**	**Learning Difficulty (0.1)**	**Data Handling Capability (0.2)**	**Total**
Pytorch	3	4	4	3	**3.7**
Tensorflow	5	3	2	3	**4.2**

Table 1.

Section IV.

The top solution design we chose was TensorFlow. Our software will consist of a design where Capital One will provide us with a data set as an input to our system. Either manually or using the software itself, the data will be cleaned. Cleaning the data involves eliminating missing data fields and ensuring any data entries are in the correct format. This process ensures that the output obtained will not be based on faulty inputs. Once the data is properly formatted, the machine learning algorithm will compute the nonlinear regression. The software will then provide two outputs; one output will be the visual output of the nonlinear regression, and the other output will serve as a test result to be used to verify the accuracy of the algorithm.

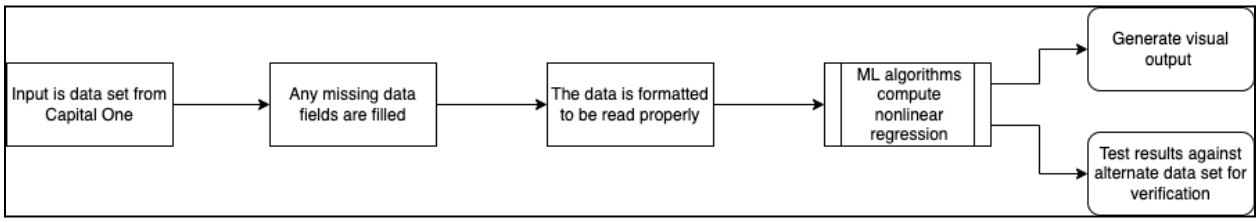

Figure 1.