Evelyn Olalekan-Elesin

Scientific Computing,
Masters Studies.
0404503
28/12/2021

Background

I would like to apply my Mathematical knowledge towards Data Analytics, Artificial Intelligence and Machine learning and so I started learning Python on my won while also studying and so when it was time for my internship, I applied to different roles such as Data Science Intern/ work student roles but ore experience was required of me and so I decided to enrolled in a 12 weeks intensive online bootcamp for Data science at Spiced Academy as my internship to gain more knowledge and hands on experience. Below are the details and breakdown of what I learnt and its application. Here is the link to all my projects all through the bootcamp.

Version control with Git bash
Version control is needed and used to store changes in program code or files over time especially when working in an organization or in a collaboration.
To do this, a repository (both local and remote) are needed. A repository is a directory (or folder) that is being tracked with version control.
The local repository is the git repository on the computer or machine, which has a remote repository counterpart which is physically located on a different machine, i.e. “in the cloud.” This remote repository is usually hosted on a website like github.com.

Data Visualization
This is the technique used to encode numbers to visual objects to communicate data. This technique is needed because not everyone understands tables and so for easy communication and understanding, after reading, cleaning, understanding the data gathered from different sources and analyzing them, one can then proceed to plot graphs and tables in order to show stakeholders and other people concerned.

Machine learning
"A program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T as measured by P, improves with experience E" (Tom Mitchell, 1997)
Machine learning models learn, identify, and then make decisions with minimal intervention from humans. Data is the lifeblood of all business and so Machine learning can be used to unlock the value of corporate and customer data and then in turn making decisions that keeps the company ahead of its competition.
Machine learning is divided into 2 parts:
Supervised learning

Unsupervised learning

Supervised learning is divided into 2 parts which are classification and regression. These are methods with which the data is fed into the machine.
Unsupervised learning is the use of artificial intelligence (AI) algorithms to identify patterns in datasets containing data that are not labeled. They usually perform more complex processing tasks than supervised learning systems.

Machine learning workflow

1. Define the business goals: This is important before venturing into analyzing data for any business.
2. Get the data: We can then proceed to get the data. Data can be collected from customers, consumers, users and so on.
3. Train-test split: After collecting the data, we then proceed to split into training data (which we will use to train the machine for it to learn the features in order to predict the test data, and also data that the machine has not seen before) and test data which will be used to see how well the model worked and how good it can predict.
4. Exploring the data: Due to the various methods that data has been collected, we have to explore the data which might involve cleaning the data, checking for missing values.
5. Feature Engineering: After the data has been explored, we can then go ahead to process it for better representation. This could be handling rows with missing values, (Imputation), transforming categorical variables into binary features (One hot Encoding), transforming numerical variables into categorical features and so on.
6. Train models: We can then proceed to train the transformed dataset with any machine learning model you wish to go for.
7. Cross Validation: This process is used to estimate or evaluate the machine learning model that was chosen.
8. Calculate test score.
9. Deploy and monitor

Some Machine learning models

1. **Logistic Regression**: It is a classification Machine learning model method, under supervised learning. It uses a linear model and then converts it to probabilities using the sigmoid function.
2. **Random Forest**: This is an ensemble method for both classification and regression. This is done by constructing a multitude of decision trees during training. Prediction is made by aggregating (majority vote or averaging) the predictions of the ensemble
3. **Decision Trees**: This method is used to visually represent decisions and decision making as it uses a tree-like model of decisions using conditions.
4. **Naive Bayes**: This is a classification algorithm that is suitable for binary and multiclass classification. It predicts by assigning class labels using conditional probability.
5. **Linear Regression**: This predicts a scalar, and hence a regression method. It is used to predict the value of a variable based on the value of another. The value we are trying to

predict is the dependent variable and we are trying to predict it from an independent variable.

Databases
In computing, a database is a data structure that stores organized information or data electronically in a computer system. The data can then be accessed easily, modified, controlled, managed and organized. The database could live on a computer (also called the host) or in the cloud.

Some types of Databases
Relational databases
NoSQL database
Object oriented databases

Examples of relational databases are MySql and PostgreSQL.

In order to create and interact with the database, software is needed (e.g. Postgres) which will act as an interface between the database and the user or other applications.

PostGres is an entirely separate piece of software outside of Python (in fact, it is largely written in the C programming language) and so each user needs to install based on the local machine operating system.

To access the PostgreSQL database, we need:
The name of the database
Host name: 127.0.0.1 (local host)
Port: 5432
User
Password.
To put it together:

Psql -h localhost -p 5432 -U postgres

We can then go ahead to create tables (CREATE TABLE name), list tables (\d), check what databases exist(\l).
We can then start analyzing data from the tables by selecting columns (SELECT * FROM table_name)

Apart from relational databases, we also have non relational databases also referred to as NoSQL.
A non relational database is a database that does not use the tabular schema of rows and columns often used in relational databases. However, it tends to be more flexible and is a document oriented database which uses a storage model that is optimized for specific requirements of the type of data to be stored.

An example of a NoSQL database is MONGOdb.

MONGOdb is the most popular Non relational database and it uses JSON-like documents schema to store data.
Data is organized in collections in a MONGOdb which is similar to tables and can communicate with other programming languages such as Python using pymongo.

Cloud database
A cloud database is a database built and accessed through a cloud platform. This enables users to host databases without having to buy hardware. Choosing a cloud database is important especially as an organization so that users can access the database virtually from anywhere using APIs or web interface.
The RDS (Relational Database service), which is an Amazon service that hosts databases in the cloud. Other cloud computing providers are Microsoft and Google amongst others.
AWS(Amazon Web Services) offers lots of important cloud computing such as:
EC2 instance: This is a virtual server one can customize oneself.
S3 buckets: These are large, yet cheap storage space.
RDS: Relational database in the cloud e.g. Postgres.

Time series Analysis
Time series analysis can be defined as a specific way of analyzing a sequence of data points collected over a period of time. Time series shows how variables change over time and to see the changes, we need a large data set to ensure consistency.
Time series is important and useful to companies because it helps them to see patterns, trends, seasonality and what causes it. They can also use it to analyze and then predict future events. Time series are used especially to forecast the weather, interest rate, stock prices and so on.

The components of a time series are as follows:
Trend: The trend can be upward or downward sloping, it can be linear or polynomial. We can then model it by using linear regression with timesteps.

Seasonality: Seasonality can be defined as when the data set we are working with experiences regular and predictable changes that are repetitive every calendar year.
Remainder: This is what remains when we remove the trends and seasonality from the data set. It is the random fluctuations that the trend and seasonality cannot explain.
The reason we decompose these components is so that we can improve the understanding of the time series and also improve its accuracy.

There are different ways to model the remainder of the time series data using statsmodel Python APIs.

1. **Autoregressive model**: This is a time series model which uses observations from previous time steps as input to a regression equation to predict the value at the next time step.

2. **Autocorrelation model:** This model makes an assumption that the observations at previous time steps are useful to predict the value at the next time step.
3. **ARIMA**: short for 'Auto Regressive Integrated Moving Average' is actually a class of models that 'explains' a given time series based on its own past values, that is, its own lags and the lagged forecast errors, so that equation can be used to forecast future values. An ARIMA model is characterized by 3 terms: p, d, q, where, p is the order of the Autoregression term, q is the order of the moving average term and d is the number of differencing required to make the time series stationary.

MCMC Simulation

Markov Chain Monte Carlo (MCMC) simulation is a class of algorithms for sampling from a probability distribution based on constructing a Markov chain that has the desired distribution as its stationary distribution.
A Markov Chain describes a Stochastic process where each state depends only on the previous one. Each transition in a Markov Chain happens with a transition probability that is conditional on the present state. These probabilities can be written as a transition probability matrix P.
Long term dependencies exist in Markov Chains, but they are fully encoded in the transition probabilities. If you know the current state, that's enough. Knowing the past states does not provide additional information.
Assumptions of MC models:

- there is a finite state space
- Markov Assumption: a state only depends on the previous state
- no hidden states: all states are known and observable
- discrete time: time is measured in discrete steps
- time-homogenous: transition probabilities do not change over time

Object oriented programming
Object-Oriented-Programming (OOP) is a programming paradigm based on the concept of classes and objects. In OOP, programs consist of objects that interact with each other.
The objects

- represent real world objects (or concepts).
- contain data (called attributes). The attributes describe the state of an object.
- contain functions (called methods). The methods alter the state of the object or let the object do something.

Thus, objects combine data and code into one structural unit.

Deep learning

Deep learning is a subfield of Machine learning concerned with algorithms inspired by the structure and functions of the brain which is also called Artificial Neural network. Artificial Neural Network and Deep learning usually refer to the same thing.
Neural Networks are used for for nonlinear problems tasks with complex input data:

- recognizing objects in images

- face recognition
- speech recognition in a home assistant
- automatic translation

There are different types of Artificial Neural Network

- Single Neuron: inputs and bias are multiplied with weights and then transformed by an activation function. Activation functions are functions that are added into an artificial neural network in order to help the network learn complex patterns in the data. Types of Activation functions are Softmax, RELU, Tanh, Sigmoid and lots more.
- Feed Forward Network: (also called Multi-Layer Perceptrons or MLPs), there are one or more hidden layers. The information moves strictly from left to right. One hidden layer large enough is sufficient to learn anything. In practice, *deep* networks with multiple layers and other architectures are more efficient.

To program a Neural Network using Python, we need Tensorflow and Keras. TensorFlow is a free and open-source software library for machine learning and artificial intelligence. It can be used across a range of tasks but has a particular focus on training and inference of deep neural networks. Keras is a Python library that makes building neural networks with TensorFlow easy.

Building and training a neural network in Keras consists of three steps:

- First, build a sequential Keras model by stacking layers on each other.
- Then compile the model to create a TensorFlow computation graph.
- Finally, fit the model with your training data.

Pretrained Networks

Pretrained Networks are networks that have already learned to extract powerful and informative features from data. Using a pretrained network with transfer learning is much faster and easier than training a network from scratch.
There are different types of Pretrained Networks such as Resnet18, Resnet50, MobileNet and so on.

Applications of Deep learning

YOLO (Real Time object detection)
Adversarial Networks
Style transfer

Principal Component Analysis : This is a method of unsupervised learning which is used for dimension reduction. It is a procedure that allows condense such data sets to the “most informative” dimensions.
K-means clustering : This method assigns data points to a predefined number of clusters. The algorithms starts with initial estimates for the K cluster centers (centroids). The centers can either be randomly generated or randomly selected from the data set.

Weekly Projects

Every week, we worked on projects and present every friday based on what was taught and learnt during the week

Week 1: This week's project was about creating a git repository where we pull our lecture notes from and also where we submit (push) our daily task to. We learnt how to import a dataset from a csv (and other format) into a python environment, especially ipython also known as Jupyter notebook. We also did data wrangling, how to analyze a dataset and also how to visualize it and make it understandable for stakeholders to understand. The project was based on analysis, creating different graphs such as scatterplot, histogram using Matplotlib.

Week 2 : I used machine learning models to predict passengers' survival on the Titanic. This is a classification problem and so we used the machine learning model for classification which is Logistic Regression, decision trees and random Forest. We calculated the train and test accuracy and also did cross validation score to test the models validity. We also calculated the loss function in Logistic Regression and also evaluated our classifiers.

Week 3: The goal for this week's project is to build and train a regression model on the Capital Bike Share (Washington, D.C.) Kaggle data set, in order to predict demand for bicycle rentals at any given hour, based on time and weather, e.g. "Given the forecasted weather conditions, how many bicycles can we expect to be rented out (city-wide) this Saturday at 2pm?"
I splitted the data into a training and test set, conducted an exploratory data analysis, trained a regression model, iteratively optimized the model by expanding or selecting features, calculated a RMSLE for the training and test set and then uploaded my code to GitHub.

Week 4: In this week's project, I built a text classification model on song lyrics. The task is to predict the artist from a piece of text. To train such a model, I first collected my own lyrics dataset. I used selected albums from BlackEye Peas and 50 cents. I downloaded the html page which has the songs by web scraping it and using regular expressions, I then extracted the hyperlinks of the songs page. Since we need the lyrics of the songs, I vectorized the words using the bag of words method, then trained a classification model that predicts the artist from a piece of text.

Week 5: In this week's project, I build a dashboard summarizing the Northwind Database. It is a sample database that is shipped along with Microsoft Access. The data is about "Northwind Traders", a fictional company. The database contains all sales transactions between the company and its customers as well as purchases from Northwinds suppliers. I also learnt about using databases such as PostgreSQL and how to access the data stored in it.

Week 6: In this project, I built a data pipeline that collects tweets and stores them in a database. Next, the sentiment of tweets is analyzed and the annotated tweets are stored in a second

database. Finally, the best or worst sentiment for a given is published on Slack every 10 minutes.

Week 7: I got the temperature data from www.ecad.eu. I then built a baseline modeling the trend and seasonality of the dataset. I plotted and inspected the different components of the time series and then removed the trends and seasonality from the dataset. Then I modeled the remainder after the trends and seasonality was removed using the AutoRegression model. Then I tested the remainder for stationality.

Week 8: This week, I wrote a program that simulates customer behavior in a supermarket using Markov Simulation. This will help to understand our customers better in order to optimize the layout, staffing and service of our supermarkets. I modeled the way customers move through a representative shop. The main business goals are to understand customer behavior, to explain customer behavior to our non-data staff and to optimize staffing so that the queues do not get unnecessarily long. I used the following to model a supermarket with six areas: *entrance, fruit, spices, dairy, drinks and checkout*.

Week 9: The goal of this week is to build an Artificial Neural Network that recognizes objects that you hold into the webcam. I did so by collecting raw data of oranges and apples from my webcam, built a neural network from scratch, and then trained the neural network using the dataset that I have. I also learnt about pre-trained neural networks and transfer learning.

Week 10: This week's goal was to build a recommender system with a web interface and I used a movies dataset> My recommender system recommends movies based on what you like or have watched before.

Week 11: This week, I learnt how to apply software engineering to my movie recommender. This includes learning how to build a webpage with HTML, CSS and Javascript. I also learnt how to use streamlit to build my app which is what I used for my final project at the end of the bootcamp.

Week 12: This week was all about working on finl project which we presented to everyone who attend. We had to do an online graduation due to the COVID restrictions. I built a recommender app to recommend Youtube videos similar to the ones a viewer has seen before. I used the Trending videos API to get the trending videos from 45 countries all over the world. I analyzed the data, drew some graphs and then went ahead to build an app that recommends similar videos based on what the viewer has watched and likes.

In all, I am glad I enrolled in the bootcamp as it has opened my eyes to different possibilities. I am well equipped and would also like to implement this in my thesis as would be researching (starting in January) Random Forests which next to deep learning are one of the most powerful machine learning approaches.

Thank you.