An indispensable Python : Data sourcing to Data science

Posted by **datumengineering** on August 27, 2013

Data analysis echo system has grown all the way from SQL's to NoSQL and from Excel analysis to Visualization. Today, we are in scarceness of the resources to process ALL (You better understand what i mean by **ALL**) kind of data that is coming to enterprise. Data goes through profiling, formatting, munging or cleansing, pruning, transformation steps to analytics and predictive modeling. Interestingly, there is no one tool proved to be an effective solution to run all these operations { Don't forget the cost factor here :) }. Things become challenging when we mature from aggregated/summarized analysis to Data mining, mathematical modeling, statistical modeling and predictive modeling. Pinch of complication will be added by Agile implementation.

Enterprises have to work out the solution: "Which help to build the data analysis (rather analytics) to go in Agile way to all complex data structure in either of the way of SQL or NoSQL, and in support of data mining activities" .

So, let's look at the ***Python & its eco system*** (I would prefer to call Python libraries as echo system) and how it can cover up enterprise's a*s for data analysis.

Python: functional object orientated programming language, most importantly super easy to learn. Any home grown programmer with less or minor knowledge on programming fundamentals can start anytime on python programming. Python has rich library framework. Even the old guy can dare to **start programming in python**. Following data structure and functions can be explored for implementing various mathematical algorithms like recommendation engine, collobrative filtering, K-means, Clustering and Support Vector Machine.

- Dictionary.
- Lists.
- String.
- Sets.
- Map(), Reduce().

Python Echo System for Data Science:

Let's begin with sourcing data, bringing into dataset format and shaping mechanism.

{ **Pandas**: Data loading, Cleansing, Summarization, Joining, Time Series Analysis }

Pandas: Data analysis covered up in python libraries. It has most of the things which you look out to run quick analysis. Data Frames, Join, Merge, Group By are the in-builds which are available to run SQL like analysis on the data coming in CSV files (read CSV function). To install Pandas you need to have NumPy installed first.

{ **NumPy**: Data Array, Vectorization, matrix and Linear algebra operations i.e. mathematical modeling }

NumPy: Rich set of functions for array, matrix and Vector operations. Indexing, Slicing and Stacking are prominent functionality of NumPy.

{ **Scipy**: Mean, variance, skewness, kurtosis }

Scipy: SciPy to run scientific analysis on the data. However, statistics functions can be located in the sub-package **`scipy.stats`**

{ **Matplotlib**: Graph, histograms, power spectra, bar charts, errorcharts, scatterplots }

Matplotlib: 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms.

Moreover, how can we second python support to Big data Analytics and Machine Learning. Below resources can be utilize for various big data applications:

- Lightweight Map-Reduce implementation written in Python: **Octopy**
- Hbase interaction using python: **happybase**
- Machine learning algorithm implementation in Python: **Scikit**. It has built on NumPy, SciPy, and matplotlib.

Having said that, Python is capable enough to give a way out to implement data analysis algorithms and hence to build your own **data analysis framework**.

Watch out this space for implementations of various algorithms in Python under one umbrella i.e. **Python data analysis tools**.

Source: http://datumengineering.wordpress.com/2013/08/