

The Exponential Learning Curve of Analytics

Sudhir Voleti*

We humans, being human after all, are hardwired to think linear. We find linear progressions natural and are able to better understand, explain, predict and evaluate them. Which is why truly exponential changes often tend to stump us. And this is true cutting across domain area, sector and even time. Consider the following famous quote attributed to a famous person:

“We always overestimate the change that will occur in the next two years and underestimate the change that will occur in the next ten.” ~ Bill Gates

A big reason for said over- and under-estimation is our simple inability to “get” change along exponential curves the way we “get” linear ones. Let me start with a motivating example.

The date is March 13, 2004 and the venue is California’s Mojave desert, site of the DARPA (Defense Advanced Research Projects Agency, of the US govt) Grand challenge in autonomous vehicle technologies. At stake is \$1 million in prize money. The task is to design and execute an autonomous vehicle design that will navigate and complete a 150 mile race course having numerous [small] obstacles such as sharp bends, turns, hillocks and mounds, large rocks etc. There were 15 participants. So what happened? Well, none of the vehicles could do even 10% of the course. The relative best performer was CMU’s (Carnegie Mellon University) modified Humvee, which did some 7.5 miles before (literally) crashing into a ditch. So DARPA declared the Grand challenge a bust-up and kept the prize money with itself. However, not all was lost. DARPA did manage to see some potential and some promise in the technologies and innovations on offer.

So one and a half years later, on October 8, 2005, at more or less the same venue, it called for a re-match. The prize money was now doubled to \$2 million. Also, the obstacles

were now tougher – at least three tunnels, narrow roads along cliff-edges, etc. So, what happened this time, a mere 19 odd months after the first (flop) show? Well, this time, among a dozen odd participants who registered, five actually managed to complete this (tougher) race. What’s more, four did so within 7.5 hours. The winner was a creation of Stanford’s Sebastian Thurn, which beat CMU’s entry by merely a 10 minute margin. Oh, it’s not over yet. The next date of interest is November 10, 2007, a mere 2 years later. DARPA again calls for a re-match, if you will. The prize money’s bigger now, too. So also is the level of challenge. Because this time, it is all happening in an urban setting. The rules require this time that the autonomous vehicles obey all of California’s traffic laws and demonstrate the ability for such things as merging into traffic, parking by the kerb etc. An astute reader would by this start wondering how they ever got permissions to run a race like that which potentially puts the safety of ordinary human drivers on the road at risk. What actually happened was that DARPA cleared main street and a few other streets of a Californian small town for a day, and hired some 300 professional drivers driving regular vehicles of all types to act as regular American drivers on a regular day out in a small town. OK, so what happened this time? Well, again 5 vehicles completed this, much harder task. On top of the points tally (again) was Dr Sebastian Thurn’s creation followed closely by CMU’s vehicle. However, later, it was found that Dr Thurn’s vehicle had incurred a couple of minor traffic violations (stopping late at a stop sign) and hence was dunked points for the same, pushing CMU to top spot. OK, so what happened next? Not hard to guess by now, I guess. A year later, in 2008, Google launches its self-driving car program with who else but Dr Thurn as its head.

So what was the point of this example? One point certainly is to demonstrate, though an easy-to-see example the phenomenal advances in configuring an

* Indian School of Business, Hyderabad, Telangana, India. Email: Sudhir_Voleti@isb.edu

entire battery of technologies -think of sensors (to keep tabs of the vehicle's internals as well as immediate external environment), radars (to assess relative speeds in own and neighboring vehicles and thereby collision risk), basic machine vision (to spot and read traffic lights, stop signs and pedestrians among other things), etc. These advances kept pace with a set of task challenges whose difficulty level rose exponentially. Two, the design and engineering teams didn't have a whole lot of data to start with. They quickly learned from the first race's failure enough to ensure completion of the next, much tougher track. Thus, the researchers were able to train the machine (the controlling computer(s) in the autonomous vehicles) fairly quickly with limited training data. Three, and this is where things mesh into where I was hoping to go to with this article, it points to how much and how fast the growing cognitive capabilities of machines are likely to scale up, in the coming years.

From IBM's Deep blue beating Garry Kasparov (who some call the greatest human chess player ever) in the structured, rule bound game of chess using brute-force number crunching capabilities in 1997 to 2011 when IBM's Watson beat the best human players of a much more loosely structured (and pun filled) game 'Jeopardy', the cognitive capabilities of super machines has been on a relentless rise. Very recently, Alphabet's artificial intelligence (AI) project best the world's best 'Go' player. What has now entered this mix are factors such as (a) a flood of data pouring in (on which machines can be trained), and (b) open source software platforms that allow coders, developers, programmers, modelers to collaborate, extend, test, tweak and animate an ever expanding set of algorithms, routines and programs. Thus, someone on one side of the planet working on a different explanation, prediction or optimization problem could benefit from a fix someone on the other side of the planet came up with for a different problem in perhaps a totally different area. What results is borrowing, tweaking, extending, bug-fixing, bridging (with other computational platforms) and other such processes happening 24x7 in the open-source world. I often visualize open source routines

and packages coming up as a set of Lego blocks adding to a growing repository, which anyone could configure in a way that builds something very neat, useful and perhaps unique. Where this potentially leads to is the possibility of advances across different fields bootstrapping one another and spurring on (in bursts, fits and starts) exponential learning curves in certain domains.

Notice that the motivating example focuses on things that move in the physical world ('atoms') as opposed to just in the virtual world ('bits'), though the case for the latter in terms of exponential learning curves remains just as strong. Consider ROS, or 'Robot Operating System' built on the same open-source principles that currently drive progress in an R or a Python. The race is on to enhance-capability, cheaply, of basic robots using programming to make them smarter and optimize what physical world movements are currently allowed them by their design, sensor and tooling configurations. When Microsoft launched the Kinect, which became a runaway hit, what happened in the robotics world afterwards is worthy of note. Within weeks of Kinect's release, it had been hacked for machine vision applications by enthusiasts all over the world and the videos were posted on youtube. And why not? You now had substantial quality machine-learning capabilities built into a widely available set of hardware and sensors and priced at a mere few hundred dollars (relative to what machine vision applications would cost in the pre-Kinect era). Now consider a similar story playing out in advances in other fields not just sensors such as software applications. Consider what happened the last time a standard OS + inexpensive programming tools became available? What ROS is trying to do is to leverage open source architectures to build exponential learning curves into Robotics development, potentially threatening to disrupt entire industries and sectors along the way.

To quote an old Chinese curse, we do live in interesting times. Fasten your seatbelts and hold on tight. It'll be quite a ride ahead.