The one algorithm everybody in your organization needs to learn

Algorithms! The ultimate business jargon you start hearing in rather unusual contexts where "innovating minds" in your organization try to prove that your company is *state-of-the-art*: ***"We want to let algorithms take this kind of decisions"***. I know, dear reader... this statement smells to buzzwords and is at least as meaningless as absurd... but let's try not to be too hard with ourselves... At least the right level of attention is there!

An algorithm is nothing more than a self-contained step-by-step list of operations to be performed... that's it! When ***data-driven-toddlers*** speak about letting algorithms take a particular business decision, they most probably mean having a mechanism that regularly gets inputs from other systems and based on a more or less sophisticated model computes an outcome for these inputs, based on which a set of instructions is performed.

Yet, there is an algorithm that everybody shall master... An algorithm that, if properly implemented, has the power of changing your entire organization... A sequence of well defined instructions with enormous transformative power...

The one algorithm companies really need to master

Data-driven business steering is no rocket science. It's only hard to implement because of the terrible legacy big corporations cannot get rid off. The worst way of legacy is not of IT nature, but relates to a rather comfortable mind set.... to the resistance to embracing any kind of change... to the need to defend a quite beneficial status quo, where some enjoy a power position...

The following algorithm, even looking as simple as it does, requires a seamless integration of the Business Intelligence function into the operation... It requires the emergence of a new discipline called Business Data Science, the need for the organization to separate real value adding insights from insights "for the show", a culture where all data assets are accessible to everybody and the proper infra-structure enablement.

Once all these ingredients are available, the data-driven decision making recipe can be easily cooked, just by following the algorithm below:

The **problem definition (#1)** is by far the most important step. You start off with a particular problem you want to solve. And the problem does not go along the lines of *"I need to show off as a data scientist"* -as we discussed in previous posts-.... The problem is a top priority business issue your company really struggles with! It is really important to make the problem *"measurable"*. Part of this problem definition is actually to quantify the status quo -base line- and to define the required level of improvement vs. baseline.

The **Insights generation step (#3)** is where the Data Science Job comes into picture. After applying all possibles analytical techniques and after engaging with the Business folks to fully understand the context, the first **insights are created**.

The insights you created needs to be fully **relevant for the problem**... Otherwise either you need to redefine your problem (**#22**) or if the definition is still valid, you need to find new way of creating more relevant insights (ideally, it requires an additional design thinking like deep-dive with business stakeholders to explore further dimensions of the problem and solution space) (**#3**).

Insights are as valuable as the impact of the actions you can derive from them. In **#7**, we not only get a list of potential actions informed by our insights, but also rank them by priority. Priority is typically defined by a cost-benefit ratio, including time-2-market, complexity and other impacting factors. Many of the actions you might have identified can be considered as "low hanging fruits" you want to implement anyways, especially if the cost is low and the benefit is promising. Others require more effort and more "organizational commitment" (ring-fencing of IT resources, etc). You usually don't know much about the impact of a particular action, but it is not an issue, as long as you have the proper testing culture in place to try several actions in parallel (**A/B - Muti-variate testing**)

Then you start implementing your "actions" to get the first results (**#11**). In order to quantify the contributing impact of your results to your problem, you need a set of metrics (a baseline, as we said before), that you probably included in the problem definition (**#1**).

Once you are able to measure the impact, you can then compare it (**#14**) with the goal you have set yourself (usually given as a threshold expressed as an improvement over the baseline in the metrics used for the problem definition).

A practical example: optimizing the e-Commerce Conversion Rate

(#1) Problem definition: "we need to increase the sales over our website by 2K units/month". It means that either we need more visitors (e.g.: 14% uplift) at the current conversion rate, or we need to improve our conversion rate (e.g.: +6%).

(#3) Insights generation: your Data Science team thoroughly analyses your web logs, identifying click-path bottlenecks, promising technical insights (e.g.: page loading time for particular browsers taking too long, rendering quality bellow expectations, etc), conversion issues for particular products, etc.

(#5) Insights relevance check: you try to explain to which extent the issues you found in the previous point contributes to the fact that your visitors are not buying on your site. For example, you discover that visitors coming over older versions of Internet Explorer are not able to place any product in the shopping cart... but only below 1% of all visitors uses this older browser versions. Making your e-Commerce site backwards compatible with IE 6 is not going to help you a lot with your conversion issue.

(#7) Get Recommended Actions by priority: you work out with your business stakeholders (online marketing managers, portal developers, etc), how turn your insights into actions (e.g.: you proved a negative correlation between loading time and conversion and you could analyze which elements are taking the longest to load.. Let's say the images are taking too long, then you evaluate the impact of all sorts of caching strategies, usage of CDN, using of single image sprites, etc... and you compute the cost and the benefit of each and every potential solution to decide what to do next:

Category	Action	Impact Score	Cost Score	Time 2 Market	Overall Score
Reduce loading	Reduce Image Size	68	12	99	58

time	by using Sprites				
Reduce loading time	Load related content asynchronously	45	35	70	45
Product competitiveness	Couple hero product price to market	92	75	22	61
Reduce loading time	Implement AMP standard	82	67	39	59
...	...	...	...	...	...

(#11 and #12) Take action and quantify impact: for example by creating several projects to enhance the usability of your portal, to make sure your offers remain competitive by monitoring the price distance to the main competitors, etc.

(#13 and #14) Target achieved: after putting all your measures in place and actively steering your conversion uplift, your business eventually reaches the envisioned target... and voilà! you can be proud your Data Science team has become an integral part of your business optimization strategy... not just a bunch of business-agnostic people playing with charts in their ivory tower.

Caveat reader: for the sake of simplicity, I avoid working with loops... although the entire process can be seen as a continuos improvement procedure. On the other hand, this algorithm in cheap pseudo-code has been written in a way that hopefully a non-technical reader can follow (focus is on readability and not on efficiency).