MASTER PLAN

FERNANDO RAMIREZ AND IFE OWOYEMI

1. High Level Overview

- Framework Development
- Kaggle Buddy
- Remote Runner
- Method Marketplace
- Time: For the ecosystem to mature
- Develop core of Tracking Infinity

2. Intro

One of the dreams that I have had is to find give the world the framework for education it needs. The optimal method for passing information from one generation to the next, and I think with this plan we can do that.

3. Developing core of Tracking Infinity: a knowledge auditing website for humanity

Developing the optimal core for education will require condensing the knowledge we have scattered into the different fields into core concepts in a uniform language. I think that that language will be largely correlated with the data science field and some principles from system design, but its hard to say what it will look like for sure because of the complexity of the existing fields and the their large dictionaries of specializing terminology.

4. Distilling Knowledge

The question becomes, how will we get them to light? I think the core concepts will be developed by distilled by the same communities that built them. However, attempting to convince experts in these fields to transcribe the successes of their respective fields for the sake of creating an curriculum is a pipe dream. Apart from not having the time to do it they will claim that it is too complicated to be put into a universal language, but they are wrong, and they will do it. They will transcribe their work for us when they go to develop systems to auto mate the grunt work of their fields. Thats where it will start, but as they see the benefits they will continue transcribing their work, by creating systems to manage them and then by using data analytics to model them. Thats all we need.

5. An ecosystem for analyitic's components

Now the question becomes what will the ecosystem we build to capture these systems and compare them look like?
I think that the ecosystem will work a lot like a web services marketplace, given that the problem it solves is very similar. Our data analytics scripting framework will utilize both existing accepted libraries, but also seek to monetize the process of developing data analytic components. which we call methods. The decision to incorporate a market place come from the belief that there should be an incentive to implement obscure methods apart from phd research. These implementations can be developed by teams then published to our method marketplace so that it can be available for input into scripts.

6. 3 Step Plan to popularize and increase access to Data Science field

Before we give rise to this ecosystem we need to make the problem of data analytics more accessible and sexy. The plan for this is 3 fold and is where we currently find ourselves.

There are 3 things that keep people from entering the field of data science

- The setup of a development environment
- The size of the problems
- The mathematics

I argue that we can tackle the third only after solving the first two. The first is to be solved by the development of a framework for formally logging the development of a data analytics pipeline, who's proof of concept can be found under the "kaggle" git repository under framirez730. The second will be solved by taking a page from the sites like hackerrank and github and applying it to the development of scripts for kaggle competitions specifically. From hackerrank, we can take the feasability of running code submitted by users remotely. From git we can reduce computational time by indexing applications of certain algorithms or methods on a problem, especially if the data sets are limited such as in the case of Kaggle. I will clarify.

If you checkout the "kaggle" repository you will find that the scripts are based on pipelines and stages. I argue that much like git saves hashes of changes from one file to another, to make their system feasible, we can do the same thing for the application of a certain method on certain kaggle competition. We can do this because kaggle competitions release one data set, and if we host it remotely, we can provide a web interface from which people, anybody with a computer really, can submit a script and have it run remotely. Running several scripts from people on large data sets sounds hard, but there are not that many methods that can be run on a given data set, and I argue that there will be large amounts of redundant applications of common methods on competition data sets in common ways. Gy using git style tracking of changes to scripts via sha hashes, we can check to see if a script that a user submits has been completed before by a different user and leverage the results to reduce computation remotely while still servicing the customers. This will also tackle the problem of people being interested in entering kaggle competitions but not having the hardware to run large jobs. By running them remotely we eliminate the need and allow people to focus their power on the math, there by solving the third problem and beginning the first step to creating tackling infinity!