Proposal for NBER workshop:
Big Data and High-Performance Computing for Financial Economics

My co-authors, Alessio Saretto, Amit Goyal, and I have been working on multiple hypothesis testing and have a working paper entitled – "p-hacking: Evidence from two million trading strategies."

The motivation for and the idea behind our project is as follows:

An increasingly large body of literature studies the profitability of trading strategies based on signals obtained from publicly available information. Researchers are currently tracking a number of strategies well in excess of 300 and new papers keep adding to that list. For instance, Hou, Xue, and Zhang (2018) examine 447 strategies.

In his presidential address, Harvey (2017) questions the performance of these strategies due to a number of possible problems with the way in which these strategies are discovered. For example, the manner in which they are evaluated does not align with the actual research process: many strategies are investigated, but only those that are significant are reported as only they have a viable path to publication. Further, data snooping likely leads to a number of false rejections of the null. Also, a number of data choices, test procedures, and samples may be tried until a significant result is discovered and only the significant result is reported. Harvey refers to all this as p-hacking.

We take a comprehensive approach and propose an evaluation of all information contained in the most commonly used finance datasets. In particular, we examine the performance of a large number of trading strategies generated from a list of all accounting variables on Compustat and basic market variables on CRSP. We construct trading signals by considering various combinations of these basic variables and construct roughly 2.1 million different trading signals. Since we are not interested in promoting any particular strategy, the reader should think of our exercise not as a fishing expedition to find new strategies but as a thorough use of the data to properly evaluate an hypothesis, which Leamer (1978) refers to as data-mining.

We use such a large sample as a laboratory experiment to address two questions. First, can we put a bound on the magnitude of p-hacking? Second, after accounting for p-hacking, how likely is a researcher to find a truly abnormal trading strategy? There are two essential features of our study that enable us to answer these questions.

First is our procedure for generating trading signals. By considering strategies without filtering on their ex-post significance, and/or by not relying on published anomalies, our large-scale exercise allows us to avoid p-hacking and data snooping. Moreover, although all our results are robust to various sample definitions, in order to mitigate concerns about economic robustness (Novy-Marx and Velikov (2016) and Hou, Xue, and Zhang (2017)), we exclude stocks that have prices below three dollars and market capitalization below the twentieth percentile of the NYSE distribution (i.e., microcaps).

The second essential aspect of our study is that we rely on multiple hypothesis testing

(MHT) to control the proportion of false discoveries. When studying the entire distribution of trading strategies, one has to account for the fact that some strategies' performance will appear exceptional by luck, thus leading to some false rejections of the null hypothesis of no outperformance. The rate of false discovery increases with the number of strategies considered, even when the strategies are completely independent. For instance, while a significance level of 5% implies that Type I error (probability of false rejection) is 5% in testing one strategy, the rate of Type I error (i.e., the probability of making at least one false discovery) in testing ten independent strategies is $1 - 0.95^{10} = 40\%$.

The statistics and economics literature has proposed a variety of ways for controlling the Type I error in testing multiple hypotheses. We consider the three most common approaches: family-wise error rate (FWER), false discovery ratio (FDR), and false discovery proportion (FDP). FWER controls for the probability of making even one false rejection, FDP controls for probability of a user-specified proportion of false rejections in a given sample, while FDR controls expected (across different samples) proportion of false rejections. Besides the conceptual distinction in what they are trying to control, these methods also differ in their underlying assumptions. For our purposes, the most important of these assumptions is that of independent strategies. Trading strategies are not independent of each other, as there is cross-correlation in stock returns across different firms and in the information used to construct the signals, not only across different firms but also within a particular firm (i.e., total assets and profitability are not independent). Since FDP methods deliver statistical cutoffs that rely on the cross-correlations present in the data, we rely on these methods more heavily by implementing bootstraps to obtain distributions under the null.

We calculate two measures of risk-adjusted performance for each of our strategies. First, we construct a long-short portfolio based on the top and bottom decile of each signal's distribution. We then compute portfolio alphas using the Fama and French (2015) five factor model augmented with the Carhart (1997) momentum factor. Second, we calculate the Fama and MacBeth (1973), henceforth FM, coefficient for each signal following the methodology proposed by Brennan, Chordia, and Subrahmanyam (1998).

The main takeaway from the paper is that upon comparing the rejection rates obtained from MHT to the rejection rates obtained from classical single hypothesis testing (CHT), we find that 98% of the signals that survive the CHT thresholds would not survive MHT thresholds. Thus, the great majority of the discoveries (i.e., rejections of the null of no predictability) that are made by relying on CHT and without accounting for the very large number of strategies that are never made public, are very likely false.

Why do we need computing resources?

At this point we are now running simulations to assess how the MHT thresholds change with the number of signals being investigated. We have already run S=1000 simulations with N=5000 strategies and even up to N=1,000,000 strategies. We want to increase N to 5,000,000. Each simulation has B=1000 bootstraps and so for N=1,000,000, we have to run a trillion bootstraps. This takes a couple of weeks for each set of parameter values. We were hoping to speed this up with more computing power.