CSE 599c — Big Data Management Systems (Balazinska)
Project Idea Description

Tony Cannistra
tonycan@uw.edu

To Spark or Not To Spark: An analysis of advantage at scale.

Most domain scientists perform analyses on data at a scale that few industry-scale data analysts would call “Big Data.” In reality, many of these datasets often exist in the gray area between “small data” and “big data.” As such many but not all analyses could benefit from advances in data management techniques. Many researchers operate by querying tabular data of a sort that is well-suited for relational database management systems, but the overhead to managing a DBMS for one-off analyses is often difficult to justify. In addition, the intellectual investment required to learn a big data management system is often too much to bear.

As a result, many researchers rely on flatfiles and the DataFrame abstractions in Python and R to perform their analyses, but suffer when the size of the data reaches a certain threshold. The DataFrame API provided by the latest versions of Spark exposes an interface that is familiar to many, so the question of whether launching and administering an HDFS/Spark cluster is a worthwhile task for a domain researcher no longer depends on the ease of use of the system, as many of the operations exposed by Spark DataFrames are familar. Rather, it depends on whether the scale of the data to be analyzed warrants such a system.

The purpose of this project is to better understand the scale at which Spark provides an advantage in analysis over the native Python (Pandas) and R DataFrame abstractions from a runtime perspective. To assess this, we will draw from our experience in the domain of ecological/climate modeling to select common queries on large datasets. We will vary the size of these datasets and measure the time required to perform these common queries using Spark, the Python Pandas library, and the R `dplyr` library.

The central practical motivation for this work is to develop rough thresholds to help researchers fluent in Python and R decide whether or not to begin a new analysis in Spark or the native DataFrame abstractions given a dataset of known size. Though runtime is one of many considerations when choosing between these options, there is value in having a rough notion of comparative performance when deciding how to implement an analysis.