REFERENCE ONLY
Syllabus is subject to change
Students currently enrolled in this course should reference NYU Classes syllabus only

New York University

Tandon School of Engineering
Computer Science and Engineering
Course Outline Fall 2020 CS6513 Big Data

Professors: Raman Kannan

Office Hours: email and weekly virtual meetings

Statement of Academic Integrity

Students are expected to follow standards of excellence set forth by New York University. Such standards include respect, honesty, and responsibility. This class does not tolerate violations to academic integrity including:

- Plagiarism
- Cheating in an exam
- Submitting your own work toward requirements in more than one course without prior approval from the instructor
- Collaborating with other students for work expected to be completed individually
- Giving your work to another student to submit as his/her own
- Purchasing or using papers or work online or from a commercial firm and presenting it as your own work

Please refer students to the Tandon code-of-conduct for addition information at:http://engineering.nyu.edu/life/student-affairs/code-of-conduct

Instructor allows students to source knowledge from any source including friends, colleagues, internet, library, papers and books.

All evaluations are open book and open notes and your problem solving abilities and your ability to work with other students are assessed.

What follows is an elaborate description and consistent with the code-of-conduct.

Course Pre-requisites

This offering of the course is for students who wish to prepare for a career in processing very large amounts of data. As prerequisite, students must have significant experience in

- programming,
- distributed/parallel computing,
- some knowledge of algorithms,
- basic knowledge in databases, and
- unix shell scripting.

Even though particular programming languages are specified, students are encouraged to use any language including R, java, scala, python,julia.

This course is not suitable if you are not comfortable with serious/intense hacking.

Course Description

Big Data requires the storage, organization, and processing of data at a scale and efficiency that go well beyond the capabilities of conventional information technologies. The course reviews the state of the art in Big Data analytics and in addition to covering the specifics of different platforms, models, and languages, students will look at real applications that perform massive data analysis and how they can be implemented on Big Data platforms.
Topics discussed include:

1. DataStores: SQL and NoSQL stores,
2. Map reduce over Mongo,
3. Apache Spark,
4. large-scale data mining using R and
5. visualization.

The curriculum will primarily consist of technical readings and discussions and will also include challenge exercises where participants will prototype data-intensive applications using existing Big Data tools and platforms, namely R, Relational, non-Relational, and Spark.
Students may choose to use R and/or Spark over java or Scala or python.

Course Objectives

1. To learn about basic concepts, technical challenges, and opportunities in big data management and big data analysis technologies.
2. To learn and get hands-on experience analyzing large data sets using a combination of R,MySQL and mongo or any other non-relational database.
3. To learn and get hands-on experience analyzing large data sets using Apache Spark.
4. To learn about different types of scenarios and applications in big data analysis, including for structured, semi structured, and unstructured data.

Course Structure

Materials posted on classes plus intensive interaction via the e-learning platform. There will also be a reading list of research papers, and students are expected to perform hands-on homeworks and two projects.

Readings

The required text for the course is: **Mining of Massive Datasets**. Rajaraman and Ullman, Cambridge University Press, 2011. Available online at http://infolab.stanford.edu/~ullman/mmds/book.pdf

Additional reading: **Data-Intensive Text Processing with MapReduce**. J. Lin and Chris Dyer, Morgan Claypool , 2010. Available online at http://lintool.github.io/MapReduceAlgorithms/

A list of journal and conference papers, available on the internet or via the Dibner electronic library, challenges from real-world, additional notes and presentations will be provided.

Software Requirements

The course requires the following software packages, all freely available:

1. The R Project for Statistical Computing, http://www.r-project.org/
   Optional R Studio, http://www.rstudio.com/
2. MySQL for relational, mongo for document oriented data will be provided.
3. Spark over java or scala will also be provided.

All class related work must be done IBM cloud so the work can be centrally evaluated. A login will be provided free of charge. There is no installation required. Students are encouraged to use Xming (on Windows) and Quartz (on Mac) if there is aversion to command line interactivity.

.

Other Technical Requirements

We will be performing all our work on IBM Cloud. All the (functional) projects and tests required for this course have to be delivered on the IBM Cloud.

Access to IBM Cloud will be provided by the instructor free of cost.

Course requirements

Students are expected to do, and will be graded on: **All are individual effort.**

Description	Due date/details	Grade
SQL challenges.	Every week. Students must turn in their solution in their home directories, in a file named student_id_sql_challenge_nn.sql where nn 01 through 10	10 **max=10**
Competitive engineering CE01 Students are given a million rows and are asked to engineer a solution loads them into 1 or more tables.	Top 10% performing solutions will receive A for the assignment. Next 30% performing solutions will receive A- for the assignment. Next 20% performing solutions will receive B+ for the assignment. Next 30% performing solutions will receive B for the assignment. Bottom 10% will receive B-. (points 20 points) Hard Deadline 10/07/2020	A=20 A-=16 B+=12 B=8 **max=20**
CE02: Given 100000 (O) known observations dataset and 1000 new observations (N) Students are to compute kNN distance (euclidean distance)	Top 10% performing solutions will receive A for the assignment. Next 30% performing solutions will receive A- for the assignment. Next 20% performing solutions will receive	A=25 A-=21 B+=17 B=10 **max=25**

Description	Due date/details	Grade
between N and O. Task is to compute 100 million distances.	B+ for the assignment. Next 30% performing solutions will receive B for the assignment. Bottom 10% will receive B-. (points 25 points) Hard deadline 11/07/2020	
CE03: NLP Sentence Embedding sentence embedding exercise doc2vec exercise -- Students will be provided 50 or so files. D1 ... D50 (d documents) Students are challenged to parse them into sentences S1...Sn Compute the embeddings for each sentence. Construct a distance matrix -- where the values are distance from one sentence to every other sentence. (25 points)	Top 10% performing solutions will receive A for the assignment. Next 30% performing solutions will receive A- for the assignment. Next 20% performing solutions will receive B+ for the assignment. Next 30% performing solutions will receive B for the assignment. Bottom 10% will receive B-. (points 25 points) Hard deadline 12/07/2020	A=25 A-=21 B+=17 B=10 **max=25**
Expectation Report	Hard deadline 09/12/20	4
Reflection Report	Hard deadline 12/08/20	4
Participation and contribution to course Progress reports have to entered on newclasses under assignments named CE01,CE02,CE03 every week. At the end of the month grades will be awarded.	Hard deadlines CE01 – Sept due by 09/30 CE02 –Oct due by 10/31 CE03 –Nov due by 11/30	12
	TOTAL	100

Course Topics by Week: Subject to adjustment/revision

Week 1: Course Overview. This is course is projects driven – frameworks, architectures, applications

Week 2: Databases and Big Data: Persistence, Transactions, Querying, Indexing and SQL

Week 3: Introduction to R Programming Language from Data Analytics Perspective I

Week 4: Introduction to R Programming Language from Data Analytics Perspective II

Week 5: Basic Data Mining and Statistics in SQL and R

Week 6: Distributed Problem Solving in R:Shiny, RServ, etc

Week 7: Text Processing in R and Spark – basics TF/IDF, Word2Vec, Doc2Vec, LDA, Entity Extraction.

Week 8: Learning for Scalable Text Analysis

Week 9: Algorithms for Big Data: Finding Similar Items

Week 10: parallelizing Cross Validation, LOOCV

Week 11: parallelizing Stochastic Gradient Descent, Boosting, Locally Sensitive Hashing

Week 12: SparkML – SparkKnife, Occams Razor, Classifiers as instructions and MISD

Week 13: Student Presentations

Week 14: Student Presentations

Grade distribution will be as follows:

Top 30% of students will get A and A-,

Next 25% B+,

Next 25% B,

Next 20% other grades

Hadoop is not included in the topics and most of the demos are in R.

This is not a course on analytics but many programming challenges are drawn from many disciplines including machine learning, and data mining.

Schedule:

For calendar visit here
https://www.nyu.edu/registrar/calendars/university-academic-calendar.html (please review appropriate semester) .

All assessments are individual.
Competitive Engineering Challenges
Persistence (20%) due 10/07/2020
kNN Distance Computation (25%) 11/07/2020
Doc2Vec (25%) 12/07/2020
Because grading is based on performance submissions **must run on IBM Cloud** so that everyone is measured by the same yardstick.

Continuous SQL Challenges, required homework (10%) due weekly
(SQL submitted by students must run on IBM Cloud...)
Expectation/Reflection report (8%) due 09/12/2020
Attendance and Participation, Contribution (12%) 12/08/2020

Our virtual class will be held over the zoom on classes from 8 to 9 PM on WEDNESDAY, each week. Attendance is NOT required but students are encouraged to watch the recorded sessions at a minimum. Active participation is required and submitting a weekly progress report is also required to receive class participation grade of 1% point toward your grade except for the first and the last week (12%). First week students are asked to submit an expectation report (4%) and a reflection report on the final week

(4%). Submitting a weekly progress report is also required to receive class participation grade. Based on student feedback, the deadlines are rigid and I will not grant any extension.

This course requires a lot of work, allowing students to define their own projects.

Central learning objective are

1. distributed problem solving leveraging large volume and variety of data. This course does not address velocity, the third V of Big Data.
2. Thinking and devising distributed problem solving architecture is an essential learning objective.
3. Implementing/engineering solutions using R,java,scala and Spark is the third learning objective.

To achieve these objectives we will use data from financial services, text analytics. Our focus will not be analytics or statistics or the mathematics. But given some analytical function how to compute solve problems using that function in a distributed environment. Given some statistics or mathematics how to setup parallel solution so that problems that cannot be solved in a single computer is solved using a cluster of computers.

Students are encouraged to study Ken Birman (Cornell, ISIS), Yale (Linda Gelertner) and Condor (Processor hunter) and PVM …from the 90s. Think about strategies to incorporate missing features into R and Spark.