INFO I590 Data Science On-Ramp

Fall 2022 Syllabus

Instructor: Nasim Anousheh

OVERVIEW

The Data Science On-Ramp contains a series of self-paced modules to build and enhance your data science skillset, oftentimes demanded or desired in data science related jobs. Each module is equal to one (1) credit hour. As such, the Data Science On-Ramp course has variable credit hours; you may select between one to three (1-3) credit hours during enrollment/registration.

Modules will be presented in text format, as well as video. Students will have access to a Graduate Teaching Assistant (TA) for support and office hours. If you encounter any problems, please feel free to reach our TAs either at their office hours or schedule an appointment which fits better to your schedule.

Additionally, each module has its own grading policy. In general, grading is based on assignments/projects, online discussions, and quizzes. If you select more than one module, the scores of the mini modules taken within the same semester will be averaged but not across multiple previous semesters;

Finally, these modules do not follow a specific or pre-defined sequence. You may learn your selected modules in sequence or in parallel. It is strongly encouraged you learn in parallel. This is because:

1) You may participate in the online discussion with other classmates.
2) The TAs will have weekly office hours and monthly live demos based on weekly and monthly contents of the mini courses.
3) You have a good reason to get your assignment done on time rather than rush to finish them before the end of the semester.

A list of available modules within the Data Science On-Ramp course are listed below. Further details can be found on the continuing pages:

Data Processing
Machine Learning with Python
Machine Learning with R
Machine Learning with Spark
NLP in Python
Tableau
Web Scraping
Basics of Scala
Deep Learning Principles
Introduction to Hadoop Framework
Introduction to Spark
Kaggle Cases

Data Processing

During this course, we will be working on processing, manipulating, cleaning and crunching data. This course provides a general overview of data processing techniques using Python. All examples are provided in Python; therefore, there are two modules dedicated to basics of Python programming. The main Python package that will be focused on throughout this course is Pandas; however, basic functionality of other relevant packages such as NumPy and Matplotlib are introduced as well. The course examples are implemented in Python 3.x and the assignments and quizzes ask students to write in Python 3.x as well. This course is not based on data from a specific field of science and the examples are mostly randomly generated.

This course gives students primary and general methodologies and hand-on experience to apply basic data processing techniques on real world data. It will enable all the student identify the problems in the dataset and evaluate data quality. Students will know how to clean and reconstruct the dataset based on the desired format. Students will also learn how to properly visualize different aspects and characteristics of data

Course Structure:

- Introduction
- Python Programming I
- Python Programming II
- NumPy for Data Processing
- Pandas for Data Processing
- Storing and Loading Data with Pandas
- Data Pre-Processing
- Data Wrangling
- Group Operations
- Advanced Pandas
- Basics of Time Series
- Data Visualization with Matplotlib

- Basic Concepts in Machine Learning
- Feature Conversion
- Construct Synthetic Datasets for Machine Learning Tasks
- Issues in Datasets and Evaluation Techniques
- Missing Data Handling
- Simple Feature Selection Techniques
- Advanced Feature Selection Techniques
- Dimension Reduction Methods (Random Projections, Principal Component Analysis, and Multi-Dimensional Scaling)
- Dimension Reduction for Visualization (t-distributed Stochastic Neighbor Embedding)
- Dimension Reduction with LDA

Machine Learning Principles

The goal of this course is to provide students with the knowledge and breadth of Machine Learning. This involves some of the crucial paradigms in the field such as the anatomy of Machine Learning problems, Gradient Descent, Regularization, Cross- Validation, Overfitting, Bias/Variance tradeoffs and more. Other topics covered are various practical algorithms used in Machine Learning such as Supervised Learning Problems using Linear Regression, Decision Trees, SVMs, Naive Bayes, and Logistic Regression, Unsupervised Learning Problems using K-Means and K-Nearest Neighbors, and Semi-supervised learning. Other miscellaneous topics are covered as well, such as Deep Learning, Reinforcement Learning, Ensemble Methods, and more. There are also various portions where the content is generated upon the students' demands, where they can learn about what is trending and popular in the field of Machine Learning in the current day. All of these topics will be learned using readings, quizzes, surveys, discussions, and of course coding assignments using Python and Jupyter Notebook!

Course Structure:

- Course Introduction and Overview
- Introduction to Machine Learning and the Development Environment for This Course
- Linear Regression
- Overfitting, Underfitting, the Bias/Variance Tradeoff, and Regularization
- Decision Trees
- Cross Validation
- Support Vector Machines
- Maximum Likelihood Estimation (MLE), Maximum A Posteriori Estimation (MAP) and Gradient Descent
- Naive Bayes
- Logistic Regression
- Supervised Learning, Unsupervised Learning, Semi-Supervised Learning, Reinforcement Learning, and Deep Learning
- K-Means Clustering
- Miscellaneous Topics 1
- Miscellaneous Topics 2
- Wrap-up week

Machine Learning with Python

Machine learning is a technique which is used to teach computers, without being explicitly programmed. In this course, you will learn about basics of python and extending them to use different important packages like Matplotlib, Scikit-learn in python and about different kinds of classifications and classifiers used in machine learning.

We will begin our course with basic python programs because it is good to have some basic level of python experience before we go into advance concepts like machine learning. In industries, most of the computer programmers use two important approaches to write complex applications, recursive approach, and iterative approach. You will learn about these important concepts from modules as well as from programming assignments. Scikit-learn is one of the best open-source machine learning packages in python with large active open-source community. We will use this package to learn machine learning in applied fashion. At last, we will show you how you can build recommendation system using Scikit-learn package.

Course Structure:

- Introduction to Python
- Installing Python and setting up PyCharm IDE and Anaconda
- Python strings, constants, variables and scope
- Arithmetic and binary operations
- Control structures, functions, and exception handling
- Using NumPy and Pandas library in Python
- Introduction to Matplotlib in Python
- Machine learning with Scikit-Learn and Scipy
- Concepts and implementation of linear regression using Numpy and Logistic Regression
- Introduction to Scipy
- Overfitting of curve and Ridge Regression using Python
- K-Means algorithm and its implementation using Scikit-Learn
- Implementation of SVM and Decision tree using Scikit-Learn
- Expectation and Maximization Algorithm and implementing it using Scikit-Learn
- Principal Component Analysis (PCA) and its implementation using Scikit-Learn
- Neural Networks and its implementation using Scikit-Learn

Machine Learning with R

In this course it is expected that you know the basic functionalities in R coding andwe are going to cover the machine learning topics, how to implement them, what are the famous packages in R community, and how we can use those packages and how we can play with the different parameters in the packages which will affect the results.

This will be a short course with 10 modules which will cover almost all widely used machine learning algorithms. Don’t worry, I won’t be adding a lot of theory to it rather I will be adding a lot of screenshots and code to give you a much better experience.

My expectation is whatever task we are going to perform, please try to do hands-on side by side on your system. Don’t take this class as a theory lecture rather take it as a lab session.

Course Structure:

- Getting started
- Principal Component Analysis
- Linear Regression
- Logistic Regression
- Clustering
- Decision Trees
- Neural Networks
- Support Vector Machines
- Text Mining
- Time Series Analysis

Machine Learning with Spark

Through this online course, we will introduce you how to do Machine Learning on large scale using Apache Spark. The course is designed to be simple, to the point and instructive for the beginners in Spark. We hope you enjoyed the "Introduction to Spark course" which is a prerequisite for the "Machine Learning with spark" course.

The "Machine Learning with spark" course starts with introduction to *Linear Algebra* and *Python in Spark* to brush-up your skills. The course discusses the MLlib which is Spark’s scalable machine learning library consisting of common learning algorithms and utilities, including classification, regression, clustering, collaborative filtering, dimensionality reduction, as well as underlying optimization primitives. The course ends with topics like Text Mining, building a machine learning project pipeline and a final project. Our preference has been to use real world examples to make sure that students can imagine how the skills will be helpful in a real-world setting. We want to give you some hands-on experience by developing simple programs that can be easily deployed in many other situations only by being modified slightly. Moreover, we have tried to make the course easy to proceed by covering every basic concept and skill you need to develop your Machine Learning models in Spark.

Course Structure:

- Introduction to Linear Algebra
- Introduction of Python for spark
- Developing word count application of large data set using Spark
- Decision trees implementation in Spark
- Linear regression
- Logistic regression
- Unified view on Linear methods
- Unsupervised machine learning: Clustering
- Text analysis using Spark RDD
- Frequent patterns and occurrences in Spark
- Machine learning pipelines

NLP in Python

Text mining starts generally with the process of information retrieval. We need to identify the source of data and then collect from this source. General sources are web, blogs, social media platforms, reviews and comments, etc. Once we collect the data, we need to clean the noise in it, such as the removal of duplicate data entries, unwanted information such as url's, image links, etc. There are number of steps involved in denoising the data and this depends on the kind of data that you have at hand. Once we clean the text data, we can apply natural language processing techniques such as parsing, pos tagging, etc.

The whole idea is to convert something not so structured into something meaningful and structured. Once we have such a structured output, we can perform various tasks such as:

- Sentiment analysis
- Topic detection
- Document summarization
- Entity relational modelling
- Pattern recognition
- Predictive analytics
- Text categorization

In this course, we will cover some primary concepts in sentiment analysis. The above-mentioned tasks are extremely useful for gaining insights into textual data. We will explore the topics in detail.

Course Structure:

- Use regular expression to match string patterns
- Basic linux commend line functions
- Set up python and install packages using pip
- Basic functions in Python
- Basic python function working on strings
- Twitter APT to grab tweets
- Handle Json format and how to deal with it in Python
- Clean a tweet’s content by removing non-useful characters
- Use nltk to run semantic analysis on sentences
- Two projects

Tableau

Tableau is a leading data analysis software used by analytics, banking, and consulting organizations for data analysis. Tableau helps users to design/develop/deploy data science algorithms without writing huge chunks of code. The visualization of data joins and merges provides an easy way for a non-technical user to work on Data without worrying about coding in traditional scripting languages. In this course, we will learn about Tableau visualization from scratch to a professional level of understanding. We will also understand the techniques for building effective visualizations on various public data sets. The course consists of bi-weekly assignments, which mainly focus on a target problem and building visualization to discovery significant insights. There is also a final project for students to apply knowledge for a practical dataset and present their story-telling skills through interesting data visualizations.

This course will enable all the students gain all the important skills needed for building data visualizations and effective story telling. It will make the students proficient in using the tableau visualization tool and build impressive visualization storyboards in their professional careers.

Course Structure:

- Introduction to data visualization and its usage
- Familiarizing with the Tableau visualization tool
- Importing data in tableau, working with sample data set, exploring featuresin tableau. Building simple visualizations in tableau
- Working on features like filters
- Effective use of Details feature, sorting options, view tool bar, worksheetoptions.
- Creating dashboard and worksheets
- Creating calculated fields, groping set, creating hierarchy
- Working with Time Series data set
- Building effective geo maps and other custom visualizations
- Implementing K-means clustering and classification, prediction in tableau
- Final Project

Web Scraping

As a Data Scientist, one is responsible for crunching humongous amounts of data to extract insights and streamline businesses based on the results. But the role of a Data Scientist doesn't start with understanding and analyzing data. Before we do any analysis, we must have data at hand. The first step to solving any data problem is to identify the problem, followed by collecting relevant data, and cleaning and representing the data in a functional form. Then we can use visualization and other analytical techniques to glean any useful insights.

It is fundamentally essential that data scientists can collect data from various sources. Data could be available in a structured form via well-defined REST APIs or unstructured (raw) data from websites, and any other type of data in-between. The Web Scraping course is all about extracting data of interest from any source.

The course will be divided into 5 parts. The first part deals with the basics of Python, which is completely optional for students with prior experience using Python. However, I recommend taking a quick glance at it unless you use Python on a day-to-day basis.
The second part of the course deals with advanced Python coding necessary for web scraping. The third deals with extracting structured data using APIs. In the fourth part we throw light on basic tools and packages of Python for web and chrome development tool. Our fifth and final part deals with extracting raw data from web pages using Scrapy package.

Course Structure:

- Part 1: Fundamentals of Python (Optional)
  - Using iPython notebooks
  - Control flow
  - Functions
  - Data Structures: Lists, tuples, dictionaries
  - Iterables and generators
- Part 2: Essentials of Python
  - Object-oriented programming using Python
  - Error and Exception handling
  - File Input / Output
  - CSV files
  - JSON files
  - Strings and Regular Expressions
- Part 3: Structured Data Extraction
  - REST APIs
  - Twitter API
- Part 4: Fundamentals of Web Data and Developer Tools
  - HTML
  - XML
  - Chrome dev-tools
  - urllib package
  - BeautifulSoup package
- Part 5: Building Spiders using Scrapy
  - Scrapy package

Basics of Scala

Scala is a very fancy and new programming language. It is popular especially in industry in the recent years. As a functional programming language, it is somewhat like Java but with more flexibility. It can even run on JVM (Java virtual machine). This course was designed to get you familiar with Scala constructs and features. This course does not require any prerequisites, but students should have a basic understanding of object-oriented programming. This course uses a data-centric approach to Scala. All content in this course is standard basics in Scala. If you can follow each session closely, you are guaranteed to get some useful knowledge about Scala at the end. In addition, you can use Scala to solve some real-world problems.

Course Structure:

- Basic background of Scala
- Install Scala in your local environment
- Create a project in Scala IDE
- Scala REPL to run code in terminal
- OOP in Scala
- Write methods in Scala
- What is object in Scala
- Scala-particular basic concepts such as access modifiers and companion objects
- What are case object and case class
- Some synthetic methods
- Collections in Scala
- Sequences and sets in Scala
- Tuple and map in Scala
- Higher order functions in Scala

Deep Learning Principles

People who have some knowledge of machine learning and want to add deep learning to their arsenal are encouraged to take this class. While a machine learning class if not a hard prerequisite, knowing some general practical machine learning principles like regularization, validation sets, etc. will go a long way in helping you utilize the course to its maximum potential. But if you do not have a lot of machine learning experience but are comfortable with coding in python and have some working knowledge of very basic linear algebra and high school calculus, you are welcome too. Many machine learning principles have been introduced from scratch, but it is expected that you will learn the ones which haven't been dealt with in great detail. An introductory course like **Machine Learning Principles** will be very helpful before taking this class.

People who have never done any machine learning or aren't comfortable with programming in Python or aren't familiar with high school calculus and basic linear algebra shouldn't take this class. Finally, this is neither a completely theoretical course nor a hands-on recipe for implementing deep learning. If you want either of the two extremes, this course is not for you. It will try to strike a balance by first focusing on enough theory and then slowly build on more practical stuff.

You will learn about the following from this course:

- Feed-forward Neural Networks
- Deep Neural Networks
- Convolutional neural networks
- TensoFlow
- Keras

You will also develop a few interesting applications like handwritten digit recognition system in this course.

Course Structure:

- Machine Learning Primer
- Neurons – Introduction
- Neurons – Learning
- Neural Networks
- Neural Networks in Practice
- Deep Networks
- Practical issues in deep learning
- Convolutional Neural Networks
- Recurrent Neural Networks

Introduction to Hadoop Framework

Unlike many of the online articles that you may have already seen, here we do not want to talk about how you can improve your resume by acquiring Hadoop MapReduce knowledge and skills, nor do we want to emphasize the importance of Hadoop and MapReduce to the information technology industry, etc. We know that you already understand how important it is from different aspects; in fact, that is probably why you are taking this course.

Our goal in this course is trying to teach you some practical skills so you can do something cool using Hadoop, like developing a program to rank some documents based on their relevance to a search query. We will start the course in the form of questions and answers, which is we assume that you have already faced with some questions when wanted to learn about Hadoop and MapReduce by yourself, but never found a clear answer for them. Then we will proceed by introducing different aspects of MapReduce and other systems designed on top of Hadoop. Throughout the course, we will make sure that you get hands on experience by developing simple programs to work on real-world data and scenarios. Moreover, we have tried to make the course easy to proceed by covering every basic concept and skill you need to develop Hadoop and MapReduce programs, so you do not to look for other resources frequently while taking the course.

Course Structure:

- Basics of MapReduce
- Developing MapReduce programs in Java
- Installing Hadoop on your computer and running your first Hadoop program
- HDFS (Distributed File storage systems) and Yarn concepts
- MapReduce application development and configuration
- MapReduce Job architecture
- Inverted indexing technique for text retrieval
- Graph processing in Hadoop
- Analyzing stack exchange posts dataset using Hadoop
- Introduction to Apache HBase
- Writing MapReduce jobs on HBase
- Introduction to Apache Hive
- Analyzing Stack exchange dataset using Hive
- Final project-Implementing Pagerank algorithm using MapReduce

Introduction to Spark

Through this online course, we will introduce you what Apache Spark is, how it can be helpful, and where its power resides. The course is designed to be simple, to the point and instructive for the beginners. We will not be surprised to see many students who has already tried other online tutorials or coerces about Apache Spark, but very soon has found the concepts very confusing. However, here we understand this fact and it is number one priority to express all key concepts in a very straightforward language and try to avoid unnecessary and confusing fancy statements. Additionally, our preference has been to use real world examples to make sure that students can imagine how the skills will be helpful in a real-world setting. We want to provide you some hands-on experience by developing simple programs that can be easily deployed in many other situations only by being modified slightly. Moreover, we have tried to make the course easy to proceed by covering every basic concept and skill you need to develop Spark programs, so you do not need to look for other resources frequently while taking the course.

Course Structure:

- Introduction to Apache Spark
- Apache spark components: Spark Core, Spark SQL, Spark Streaming, Spark MLLib
- Installation of Apache Spark
- Writing your first spark application
- Resilient Distributed Datasets (RDD) in Spark
- Data partitioning in Spark
- Importing and exporting data into Spark
- Accumulators and Broadcast variables
- Spark interaction with R
- Introduction to Spark SQL

Kaggle Cases

In this course we will focus on classic workflow of taking kaggle competitions. We will discuss three introductory Kaggle competitions. They are tasks about **regression**, **binary classification,** and **multiclass classification**. We will get through all the necessary steps to complete these competitions, namely **exploring** and **preprocessing data**, **constructing**, **tuning,** and **evaluating models**. Specifically, we will mainly demonstrate and discuss the relevant algorithms and techniques about **missing value imputation, feature encoding and selection, linear regression**, **logistic regression**, **One-Vs-The-Rest**, **One-Vs-One**, **softmax regression**, **K-nearest neighbors**, **RBF regression**, **ridge** and **lasso regularization**, **K-fold cross validation** and **ensemble methods** such as **random forest** and **adaboost**, etc**.** All the models and techniques learned in class to solve competitions will be implemented in **Python**, with the help of popular Python packages **Jupyter notebook**, **scikit-learn** and **pandas**.

Course Structure:

- Introduction to Kaggle
  - Basic Knowledge Review_Part 1
  - Basic Knowledge Review_Part 2
- Case Study One: Linear Regression_Part 1
- Case Study One: Linear Regression_Part 2
- Case Study One: Linear Regression_Part 3
  - Project 1: Regression
- Case Study Two: Logistic Regression_Part 1
- Case Study Two: Logistic Regression_Part 2
- Case Study Two: Logistic Regression_Part 3
  - Project 2: Binary Classification
- Case Study Three: Multiclass Classification_Part 1
- Case Study Three: Multiclass Classification_Part 2
  - Project 3: Multiclass Classification