DAT Class Syllabus

Basic Info

Instructor: Jonathan Bechtel

Contact Info: jonathanbechtel@gmail.com

Instructor Associate: Vonn Johnson

Instructor Associate Contact Info: vonn.n.johnson@gmail.com

Class Duration: 01/21 – 03/26

Class Slack: http://bit.ly/dat-slack-01-21

Office Hours: Tues: 5:30-6:30 (in class), Friday 5:30-6:30 (via Slack)

Github Repo: http://bit.ly/dat-01-21

Class Schedule

Week	Class 1	Notes	Class 2	Notes	Reading
Unit 1: Python Foundations					
1	01-21-2020	Intro, Class setup	01-23-2020	Python Foundations	None
2	01-28-2020	Python Foundations	01-30-2020	Flex/Python Foundations	None
Unit 2: Pandas and Exploratory Data Analysis					
3	02-04-2020	Pandas – Connecting, Syntax	02-06-2020	Pandas – Common Operations	Py4DA: Ch.5, Ch.6
4	02-11-2020	Pandas – Grouping, Dates	02-13-2020	Graphing – Pandas, Seaborn	Py4DA: Ch. 7, 8, 9
Unit 3: Model Building and ML Fundamentals					
5	02-18-2020	HW II Presentations, Stats Intro	02-20-2020	Ordinary Least Squares, SKlearn intro	ISL: 2.1
6	02-25-2020	Cross Validation	02-27-2020	Data handling/Data prep	ISL: 3.1 – 3.3; 5.1

7	03-03-2020	Regularization, Lasso/Ridge regression	03-05-2020	Ensembles	ISL: 6.4
8	03-10-2020	Classificaion	03-12-2020	HW Presentations/FLEX	ISL: 8.1, 8.2
Unit 4: Additional Topics					
9	03-17-2020	API Data / Web Scraping	03-19-2020	FLEX	ISL: 4:1 – 4.3
10	03-24-2020	FLEX	03-26-2020	Final HW Presentations/FLEX	

Important: This schedule is tentative and subject to change. It's normal for the class to diverge from the set schedule to some degree, and class material will be modified in an appropriate manner to make sure the class itself is completed in the most appropriate manner. Some flexibility is built into the syllabus with the expectation that class material will naturally take its own unique direction.

Class Expectations

This class is pass and fail, and a student's ability to get a passing grade is dependent on their attendance and successful completion of homework projects.

Attendance: Students must have an 85% attendance mark throughout the course. Classes missed entirely are worth one unit of attendance, and uncommunicated tardiness or early absences count for a ½ unit of attendance.

In general, judging late arrivals and early dismissals will be lenient if reasons are communicated beforehand.

Likewise, if students have arrangements that come up that force them to miss additional amounts of time then I'm happy to make accommodations as long as the issue is communicated to me clearly and in a good faith manner.

Homeworks: Every homework must be successfully completed in order to receive a passing grade. Every homework is Pass/Fail. It's understood that students might have different levels of aptitude upon entering the class, and so most homeworks give students a choice of projects to choose from depending on the amount of time they have available as well as their level of expertise in the particular subject.

Class Homework

The class is broken up into 4 modules, and each one has its own homework assignment at its completion. With the exception of the 1st homework, students will be expected to give an 8-12 minute presentation on their project in front of the class when it's due.

Each homework assignment is Pass/Fail, and students will receive a written evaluation after each homework assignment approximately one week after the assignment is due with their final grade and feedback.

Successful completion of every homework assignment is required in order to receive a passing grade for the class.

Homework Descriptions

Homework 1: Python Foundations Coding Challenge; Due Date: 02/04/2020

Overview: The primary purpose behind this homework is to make sure every student has a minimally acceptable level of basic programming knowledge to get through the rest of the class. It's designed to mimic coding challenge type questions one might encounter at a job interview and is meant to test an ability to grasp basic types of problem solving one might encounter when trying to write functions.

Homework 2: Exploratory Data Analysis with Pandas; Due Date: 02/18/2020

Overview: Homework 2 is designed to test a student's ability to take a dataset, perhaps a messy one, and be able to clean and format it in a manner that allows for coherent insight and analysis. Students will be given two different datasets to evaluate, and a number of prompts for them to answer about the data itself, with some additional leeway to provide their own interpretation.

Students will also have the choice to use their own on this assignment if they find it relevant.

Homework 3: Model Building with Scikit Learn; Due Date: 03/12/2020

Overview: Homework 3 is designed to allow students to use different types of statistical techniques on a dataset in order to draw inferences from it and make useful prognostications about what might happen next.

Students will have their choice of three different assignments to choose from, that will range from a dataset that's fairly simple to others that are more elaborate and require more complicated processing and inferential techniques.

Homework 4: Independent Project; Due Date: 03/26/2020

Overview: The final homework assignment will allow students to choose a datset and project of their choosing and create their own learning path that seems appropriate for them. The idea is that at this point students should have a clear idea of how to use the skills learned in the course to best further their own learning agenda, and this project will give them a chance to accomplish this.

Unit 4 Bonus Topics

The material to be covered in Unit 4 is open ended, with 2-3 classes deliberately scheduled with no specific purpose to best accommodate class needs at that particular point in time.

A portion of Unit 4 might be spent going back over topics discussed in previous units if students feel like the class is moving too quickly for them, so it's possible it might just be used for additional review so students feel like they have a better grasp on material previously covered.

However, at the conclusion of Unit 3 students will have the ability to vote on their preferred course of direction to take in the class, and use remaining classes to cover additional topics which interest them.

Topics that can be covered include:

ARIMA(p, d, q) models (2 classes): A statistical technique that covers how to make forecasts based on how a target variable changes based on the passage of time. Very useful for making short-term forecasts where the passage of time is the principal variable.

Natural Language Processing: How to take natural language corpus and transform it into a dataset to be used for statistical learning. Will go over processing techniques, as well as most common statistical methods to be used on natural language datasets.

Boosting: A very widely used Machine Learning technique that aggregates decision trees in a manner that overweights the previous tree's bad predictions. Primarily used in a library called xgboost. Boosting is probably the most accurate way to model structured data. Very useful if you want to have top-shelf accuracy on information that's stored in tables.

Deep Learning (2 classes): Aka, neural networks. This is the primary analytical technique that's used on **unstructured** data. So if you want to evaluate images, audio files, text documents, etc, then this is the way you do it. Requires new libraries……either PyTorch or Tensorflow, as well as cloud based deployments.

Cloud Based Deployments: Lots of important data science happens using distributed, cloud based deployments, and it's useful to understand some of the basic architectures that are used to do machine learning at scale. This lesson would go over the basics of Amazon Sagemaker, which is an end-to-end machine learning platform that allows you to use as much storage and compute power as you need.

Model Deployment: Most of this class will be spent on the basics of how to take a machine learning model, and embed it into an actual software application. This will be useful if you want to get outside of your jupyter notebook and actually use a model inside some sort of larger ecosystem. Topics covered will be model serialization, as well as the basic syntax of the flask web app platform.

Class Errata

Class Style

Most classes will have a fairly similar format:

- One 'big idea' that will be the focus
- Usually about 45 minutes – 1 hour of presentation/slides, often with students doing a codealong
- An individual or group project that goes for about 25-50 minutes
- Overview of the material as well as wrap up and introduction to the next topic

General Teaching Philosophy

In general, I think my job in this course is broken down into two categories:

- I should explain complicated topics clearly, and make them more easily digested than they otherwise would be if using other avenues to learn.
- I should push you outside of your own comfort zone to learn topics and concepts you maybe don't see yourself as being able to, or being necessary for your own career path.

With regards to point 1, our time together should be breezy, enjoyable and (maybe) fun. With regards to point 2, the experience should be uncomfortable, frustrating and much less entertaining than other uses of your time.

If we're doing a good job, we should be spending a decent amount of time dipping our toes into both experiences. So while we should expect to have a good time together, please remember **that the most effective learning necessarily requires some discomfort.** You are only going to take this class once and it's imperative that we work together to make sure it has maximum impact for your career goals, which sometimes requires a certain appreciation of short-term pain in order to get some long-term gain.

Online Class Discussion

The class has its own slack channel, which every student is invited to at the beginning of class. This is intended to give students a chance to ask each other questions about homework assignments, discuss general topics, as well as provide a place for me to make announcements about class assignments or other class wide details.

Class Prompts

To make class material more 'complete', many classes will come with either suggested reading or prework that might help students better prepared for material, or allow more advanced students delve into more differentiated material to make class material more satisfying.

This material will be released to the class GitHub repo, and will be announced on the class Slack channel.

Completing these prompts is optional, but is meant to give students a more thorough, engaging learning experience that's appropriate for their level of expertise.

Class Material/GitHub

The class has a github repo that it uses to disseminate all class material. Students are expected to continually initiate pull requests to get new course material as it's released. **Class material will be dripped out class by class.** If a student is going to be absent arrangements can be made for them to get material beforehand.

Students are also expected to setup their own GitHub repo, which is where they will publish their homework assignments for me to grade.

Class Readings

There are no required class readings, and students aren't expected to do any class readings in order to follow along with the course, but if students want to do some additional background research or prep themselves for material optional readings are given from the following two books:

Python for Data Analysis, Wes McKinney

URL: bit.ly/dat-book-1

Description: This book was written by the primary author of Pandas and is a good overview for how to use its various nuts and bolts to accomplish data cleaning tasks. It's not available for free, and must be bought.

Introduction to Statistical Learning, Robert Tibshirani, Trevor Hastie

URL: bit.ly/dat-book-2

Description: This book is a good introductory explanation about various types of Machine Learning concepts, and is meant to target someone who is math-curious but doesn't want to get too drowned into low-level details about WHY a particular concept works the way it does.

It's approach is a bit 'high level', in that it's meant to communicate broadly what a concept does and its main strengths and weaknesses, but doesn't provide a lot of code examples. Some students find it a bit bland or dry, but it's generally one of the best resources for explaining what a concept is to an entrylevel practitioner. It's available for free online.

Class Dinner

There will be an optional class dinner held after hours so everyone has a chance to get together and meet one another in a more casual way. Dates will be set on the fly, but arrangements will be held over the class Slack channel, and will be held around weeks 3 & 7, as well as at the conclusion of class.