INFO 7390

Advances in Data Sciences and Architecture

Fall 2019 Course Syllabus

Course Information

Professor: Nik Bear Brown

Email: nikbearbrown@gmail.com

Office: 505A Dana Hall

Office hours:

Monday 4-5:15PM

Wednesday: 3-5:15PM

Course website: Blackboard (for raw scores, uploading assignments, getting materials, & forums)

Piazza:

<https://piazza.com/northeastern/fall2019/info7390>

Course Prerequisites

A basic knowledge of the python programming language is required. Assignments are done in python.

Course Description

Introduces techniques for machine learning and artificial intelligence.

Specifically, the following topics are covered:

* Probability Review, Bayes’ Rule
* Statistics
* Distributions, Exponential Families, Sufficient Statistics, Likelihoods
* Prior and Posterior Distributions, Conjugate Priors, Joint Inference
* Sensitivity Analysis, Hypothesis Testing, ANOVA
* Unsupervised Learning (Clustering)
* Cross-Validation, Bias-Variance Tradeoff
* Linear Models, Model Inference
* Supervised Learning
* Natural Language Processing
* Time Series Analysis
* Kernel Methods, Mixture Models and EM, Marginal Likelihood
* Latent Factor Models, Latent Variables, SVD, PCA, Eigenvectors, Probabilistic PCA
* Ensembling, Bayesian Model Averaging, Boosting, Tree-based Models, Conditional Mixture Models
* Deep Learning
* CNNs, RNNs
* Autoencoders, VAEs, and GANs
* Time-Series Models, Auto-regressive NNs

The ability to use python is part of the grade. Students must demonstrate ability to setup data for learning, train, test, and evaluation using either python or R, but all assignment examples and solutions will be presented in python. The assignments include paper exercises designed to reinforce conceptual understanding. Quizzes and exams. A term project is required. A portfolio blog is required.

Communication

Communication between instructor and students is through

● Email via the Blackboard distribution list

● Announcements posted on Blackboard

● Notes posted on the Blackboard discussion board

● Private email exchanges

Course Structure

* Regularly test students on paper/algorithmic exercises
* Evaluate students’ implementation competency, using assignments that require coding on given datasets
* Evaluate ability to setup data, code, and execute using python language
* Exams
* Final project is typically asking and answering a “real world” question of interest using machine learning techniques

Course GitHub

The course GitHub (for all lectures, assignments and projects):

<https://github.com/nikbearbrown/INFO_7390>

nikbearbrown YouTube channel

Over the course of the semester I’ll be making and putting additional data science and machine learning related video’s on my YouTube channel.

<https://www.youtube.com/user/nikbearbrown>

The purpose of these videos is to put additional advanced content as well as supplemental content to provide additional coverage of the material in the course. Suggestions for topics for additional videos are always welcome.

Schedule

|  |  |  |
| --- | --- | --- |
| Week | Topic | Assignments |
| 1) Week 1 | Probability:  Counting  Understanding Probability Distributions  Random variables, quantiles, mean variance  Conditional probability, Bayes' theorem, base rate fallacy  Joint distributions, covariance, correlation, independence  Central limit theorem  Bias versus variance | Readings; Assignment 1 |
| 2) Week 2 | Statistics:  Bayesian inference with known priors, probability intervals  Conjugate priors  Bayesian inference with unknown priors  Frequentist significance tests and confidence intervals  Resampling methods: bootstrapping  Latent Factor Models, Latent Variables, SVD, PCA, Eigenvectors, Probabilistic PCA | Readings  HackerRank Online Quiz |
| 3) Week 3 | Unsupervised Learning (Clustering)  Selecting and Evaluating Models  Theory of Modeling  Validation  Cross-Validation  ROC curves  Exam | Readings; Project proposal  Assignment 2 |
| 4) Week 4 | Linear Models, Model Inference and Interpretation  Regression  Logistic Regression  Regularization | HackerRank Online Quiz |
| 5) Week 5 | Supervised Learning  Decision trees  k-Nearest Neighbors algorithm (k-NN)  Support Vector Machines | Readings; Assignment 3 |
| 6) Week 6 | Data Visualization |  |
| 7) Week 7 | Text Mining & Natural Language Processing  Word2Vec  GLOVE | Readings; Assignment 4  Exam 1 |
| 8) Week 8 | Neural Networks  Intro to Deep Learning  MLP | Readings; Project progress report. |
| 9) Week 9 | Deep Learning  CNN | Readings; Assignment 5  HackerRank Online Quiz |
| 10) Week 10 | Deep Learning  RNN |  |
| 11) Week 11 | Deep Learning  Autoencoders, VAEs, and GANs | Readings; Assignment 6  HackerRank Online Quiz |
| 12) Week 12 | Time Series Analysis  ARIMA models  Trend Analysis  Seasonal Models  Auto-regressive NNs |  |
| 13) Week 13 | Project Reports  Final Exam | Final Exam |

Teaching assistants

The Teaching assistants are:

TBA

Programming questions should first go to the TA’s. If they can’t answer them then the TA’s will forward the questions to the Professor.

Learning Assessment

Achievement of learning outcomes will be assessed and graded through:

● Quizzes

● Exams

● Completion of assignments involving scripting in R or python, and analysis of data

● Completion of a term paper asking and answering a “real world” question of interest using machine learning techniques

● Portfolio piece

Reaching out for help

A student can always reach out for help to the Professor, Nik Bear Brown [nikbearbrown@gmail.com](mailto:nik@ccs.neu.edu). In an online course, it’s important that a student reaches out early should he/she run into any issues.

Grading Policies

Students are evaluated based on their performance on assignments, performance on exams, and both the execution and presentation of a final project. If a particular grade is required in this class to satisfy any external criteria—including, but not limited to, employment opportunities, visa maintenance, scholarships, and financial aid—it is the student’s responsibility to earn that grade by working consistently throughout the semester. Grades will not be changed based on student need, nor will extra credit opportunities be provided to an individual student without being made available to the entire class.

Grading Rubric

The following breakdown will be used for determining the final course grade:

|  |  |
| --- | --- |
| Assignment | Percent of Total Grade |
| Assignments | 45% |
| Quizzes | 15% |
| Exams | 25% |
| Research Project\* | 10% |
| Portfolio | 5% |

\* Note that the assignments, presentations and drafts related to the research project go to that score rather than the programming assignments. I expect to use the following grading scale at the end of the semester. You should not expect a curve to be applied; but I reserve the right to use one.

|  |  |
| --- | --- |
| Score | Grade |
| 93 – 100 | A |
| 90 – 92 | A- |
| 88 – 89 | B+ |
| 83 – 87 | B |
| 80 – 82 | B- |
| 78 – 79 | C+ |
| 73 – 77 | C |
| 70 – 72 | C- |
| 60 – 69 | D |
| <60 | F |

Scores in-between grades. For example, 82.5 or 92.3 will be decided based on the exams.

\* Note the score is calculated using the grading rubric and IS NOT the average of the assignments that is displayed by BlackBoard.

Blackboard

You will submit your assignments via Blackboard *and* Github. Click the title of assignment (blackboard -> assignment -> <Title of Assignment>), to go to the submission page. You will know your score on an assignment, project or test via BlackBoard. BlackBoard only represents only the raw scores. Not normalized or curved grades. A jupyter notebook file ALONG with either a .DOC or .PDF rendering of that jupyter notebook file must be submitted with each assignment.

Multiple files must be zipped. No .RAR, .bz, .7z or other extensions.

Assignment file names MUST start with students last name then first name OR the groups name and include the class number and assignment number.

Assignment MUST estimate the percentage of code written by the student and that which came from external sources.

Assignment MUST specify a license at the bottom of each notebook turned in.

All code must adhere to a style guide and state which guide was used.

Due dates

Due dates for assignments are midnight on the date assigned.

Five percent (i.e. 5%) is deducted for each day an assignment is late. Solutions will be posted the following Monday. Assignments will receive NO CREDIT if submitted after the solutions are posted. Any extensions MUST be granted via e-mail and with a specific new due date.

Course Materials

*Required text (All free online)*

Some textbooks are all available for free to NEU students via SpringerLink ([http://link.Springer.com/](http://link.springer.com/)). You must access SpringerLink from an NEU IP address to have full access and/or download these books.

If you are off-campus, in order to access resources provided through the Northeastern library outside the network, you should use their bookmarklet to load any page through the proxy: <http://library.northeastern.edu/bookmarklet>

*Required Texts*

The *required* textbooks we will be using in this class are:

Materials on github

<https://github.com/nikbearbrown/INFO_7390>

*Recommended Texts*

An Introduction to Statistical Learning with Applications in R (2013)

Authors: Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani

Free online via SpringerLink (http://link.Springer.com/) <http://link.Springer.com/book/10.1007/978-1-4614-7138-7>

The Elements of Statistical Learning: Data Mining, Inference, and Prediction (2017)

Authors: Trevor Hastie, Robert Tibshirani and Jerome Friedman

Free online <https://web.stanford.edu/~hastie/ElemStatLearn/printings/ESLII_print12.pdf>

Beginning Python

From Novice to Professional

Authors: Magnus Lie Hetland 2017

ISBN: 978-1-4842-0029-2 (Print) 978-1-4842-0028-5

<https://link.Springer.com/book/10.1007/978-1-4842-0028-5>

Python Recipes Handbook

A Problem-Solution Approach

Authors: Joey Bernard 2016

ISBN: 978-1-4842-0242-5 (Print) 978-1-4842-0241-8

<https://link.Springer.com/book/10.1007/978-1-4842-0241-8>

Lean Python

Learn Just Enough Python to Build Useful Tools

Authors: Paul Gerrard 2016

ISBN: 978-1-4842-2384-0 (Print) 978-1-4842-2385-7

<https://link.Springer.com/book/10.1007/978-1-4842-2385-7>

Learn to Program with Python

Authors: Irv Kalb 2016

ISBN: 978-1-4842-1868-6 (Print) 978-1-4842-2172-3

<https://link.Springer.com/book/10.1007/978-1-4842-2172-3>

Deep Learning - Adaptive Computation and Machine Learning series by Ian Goodfellow, Yoshua Bengio, and Aaron Courville

<https://github.com/HFTrader/DeepLearningBook>

Beginning Python

From Novice to Professional

Authors: Magnus Lie Hetland 2017

ISBN: 978-1-4842-0029-2 (Print) 978-1-4842-0028-5

[https://link.Springer.com/book/10.1007/978-1-4842-0028-5](https://link.springer.com/book/10.1007/978-1-4842-0028-5)

Deep Learning with Python

A Hands-on Introduction

Authors: Nikhil Ketkar 2017

ISBN: 978-1-4842-2765-7 (Print) 978-1-4842-2766-4

[https://link.Springer.com/book/10.1007/978-1-4842-2766-4](https://link.springer.com/book/10.1007/978-1-4842-2766-4)

Pro Python Best Practices

Debugging, Testing and Maintenance

Authors: Kristian Rother 2017

ISBN: 978-1-4842-2240-9 (Print) 978-1-4842-2241-6 (Online)

[https://link.Springer.com/book/10.1007/978-1-4842-2241-6](https://link.springer.com/book/10.1007/978-1-4842-2241-6)

Mastering Machine Learning with Python in Six Steps

A Practical Implementation Guide to Predictive Data Analytics Using Python

Authors: Manohar Swamynathan 2017

ISBN: 978-1-4842-2865-4 (Print) 978-1-4842-2866-1

[https://link.Springer.com/book/10.1007/978-1-4842-2866-1](https://link.springer.com/book/10.1007/978-1-4842-2866-1)

Introduction to Data Science

A Python Approach to Concepts, Techniques and Applications

Authors: Laura Igual, Santi Seguí 2017

ISBN: 978-3-319-50016-4 (Print) 978-3-319-50017-1

[https://link.Springer.com/book/10.1007/978-3-319-50017-1](https://link.springer.com/book/10.1007/978-3-319-50017-1)

Python Recipes Handbook

A Problem-Solution Approach

Authors: Joey Bernard 2016

ISBN: 978-1-4842-0242-5 (Print) 978-1-4842-0241-8

[https://link.Springer.com/book/10.1007/978-1-4842-0241-8](https://link.springer.com/book/10.1007/978-1-4842-0241-8)

Lean Python

Learn Just Enough Python to Build Useful Tools

Authors: Paul Gerrard 2016

ISBN: 978-1-4842-2384-0 (Print) 978-1-4842-2385-7

[https://link.Springer.com/book/10.1007/978-1-4842-2385-7](https://link.springer.com/book/10.1007/978-1-4842-2385-7)

Learn to Program with Python

Authors: Irv Kalb 2016

ISBN: 978-1-4842-1868-6 (Print) 978-1-4842-2172-3

[https://link.Springer.com/book/10.1007/978-1-4842-2172-3](https://link.springer.com/book/10.1007/978-1-4842-2172-3)

Big Data Made Easy

A Working Guide to the Complete Hadoop Toolset

Authors: Michael Frampton 2015

ISBN: 978-1-4842-0095-7 (Print) 978-1-4842-0094-0

[https://link.Springer.com/book/10.1007/978-1-4842-0094-0](https://link.springer.com/book/10.1007/978-1-4842-0094-0)

Software

python Anaconda

* <https://www.continuum.io/anaconda-overview>

Python Tutorials

Dive into Python <http://diveintopython.org>

Python 101 – Beginning Python <http://www.rexx.com/~dkuhlman/python_101/python_101.html>

The Official Python Tutorial <http://www.python.org/doc/current/tut/tut.html>

The Python Quick Reference <http://rgruet.free.fr/PQR2.3.html>

Python Fundamentals Training – Classes <http://www.youtube.com/watch?v=rKzZEtxIX14>

Python 2.7 Tutorial Derek Banas· <http://www.youtube.com/watch?v=UQi-L-_chcc>

Python Programming Tutorial - thenewboston <http://www.youtube.com/watch?v=4Mf0h3HphEA>

Google Python Class <http://www.youtube.com/watch?v=tKTZoB2Vjuk>

Nice free CS/python book <https://www.cs.hmc.edu/csforall/index.html>

datacamp.com <https://www.datacamp.com/tracks/python-developer>

Deep Learning Tutorials

MIT 6.S191: Introduction to Deep Learning <http://introtodeeplearning.com/>

Stanford Winter Quarter 2016 class: CS231n: Convolutional Neural Networks for Visual Recognition <https://youtu.be/NfnWJUyUJYU>

Deep Learning - Adaptive Computation and Machine Learning series by Ian Goodfellow, Yoshua Bengio, and Aaron Courville

<https://github.com/HFTrader/DeepLearningBook>

Participation Policy

Participation in discussions is an important aspect on the class. It is important that both students and instructional staff help foster an environment in which students feel safe asking questions, posing their opinions, and sharing their work for critique. If at any time you feel this environment is being threatened—by other students, the TA, or the professor—speak up and make your concerns heard. If you feel uncomfortable broaching this topic with the professor, you should feel free to voice your concerns to the Dean’s office.

Collaboration Policies

Students are strongly encouraged to collaborate through discussing strategies for completing assignments, talking about the readings before class, and studying for the exams. However, all work that you turn in to me with your name on it must be in your own words or coded in your own style. Directly copied code or text from any other source MUST be cited. In any case, you must write up your solutions, in your own words. Furthermore, if you did collaborate on any problem, you must clearly list all of the collaborators in your submission. Handing in the same work for more than one course without explicit permission is forbidden.

Feel free to discuss general strategies, but any written work or code should be your own, in your own words/style. If you have collaborated on ideas leading up to the final solution, give each other credit on what you turn in, clearly labeling who contributed what ideas. Individuals should be able to explain the function of every aspect of group-produced work. Not understanding what plagiarism is does not constitute an excuse for committing it. You should familiarize yourself with the University’s policies on academic dishonesty at the beginning of the semester. If you have any doubts whatsoever about whether you are breaking the rules – ask!

Any submitted work violating the collaboration policies WILL BE GIVEN A ZERO even if “by mistake.” Multiple mistakes *will be sent to OSCCR for disciplinary review.*

To reiterate: **plagiarism and cheating are strictly forbidden. No excuses, no exceptions*.*** *All incidents of plagiarism and cheating will be sent to OSCCR for disciplinary review.*

Assignment Late Policy

Assignments are due by 11:59pm on the due date marked on the schedule. Late assignments will receive a 5% deduction per day that they are late, including weekend days. It is your responsibility to determine whether or not it is worth spending the extra time on an assignment vs. turning in incomplete work for partial credit without penalty. Any exceptions to this policy (e.g. long-term illness or family emergencies) must be approved by the professor.

Five percent (i.e. 5%) is deducted for each day an assignment is late. Assignments will receive NO CREDIT if submitted after the solutions are posted. Any extensions MUST be granted via e-mail and with a specific new due date.

Only ONE extension will be granted per semester.

Student Resources  
 **Special Accommodations/ADA:**In accordance with the Americans with Disabilities Act (ADA 1990), Northeastern University seeks to provide equal access to its programs, services, and activities. If you will need accommodations in this class, please contact the Disability Resource Center (www.northeastern.edu/drc/) *as soon as possible* to make appropriate arrangements, and please provide the course instructors with any necessary documentation. The University requires that you provide documentation of your disabilities to the DRC so that they may identify what accommodations are required, and arrange with the instructor to provide those on your behalf, as needed.

**Academic Integrity:** All students must adhere to the university’s Academic Integrity Policy, which can be found on the website of the Office of Student Conduct and Conflict Resolution (OSCCR), at <http://www.northeastern.edu/osccr/academicintegrity/index.html>. Please be particularly aware of the policy regarding plagiarism. As you probably know, plagiarism involves *representing anyone else’s words or ideas as your own*. It doesn’t matter where you got these ideas—from a book, on the web, from a fellow-student, from your mother. It doesn’t matter whether you quote the source directly or paraphrase it; if you are not the originator of the words or ideas, *you must state clearly and specifically where they came from*. Please consult an instructor if you have any confusion or concerns when preparing any of the assignments so that together. You can also consult the guide “Avoiding Plagiarism” on the NU Library Website at <http://www.lib.neu.edu/online_research/help/avoiding_plagiarism/>. If an academic integrity concern arises, one of the instructors will speak with you about it; if the discussion does not resolve the concern, we will refer the matter to OSCCR.

**Writing Center:** The Northeastern University Writing Center, housed in the Department of English within the College of Social Sciences and Humanities, is open to any member of the Northeastern community and exists to help any level writer, from any academic discipline, become a better writer.  You can book face-to-face, online, or same day appointments in two locations: 412 Holmes Hall and 136 Snell Library (behind Argo Tea).  For more information or to book an appointment, please visit <http://www.northeastern.edu/writingcenter>/.