MSDS 7331 – Machine Learning I
Course Syllabus

Instructor: Jake Drew, Southern Methodist University
Email: jdrew@mail.smu.edu
Course Github Page: https://github.com/jakemdrew/DataMiningNotebooks

Optional Reading Texts:

- *Introduction to Data Mining* by Tan, Steinbach, and Kumar
  - First Edition, ISBN-10: 0321321367
  - Most information I present will come from an array of disparate sources. While many theoretical elements come from this text, it is best used for review of video content and examples.

- *Python Machine Learning* by Sebastian Raschka
  - First Edition, ISBN: 978-1-78355-513-0
  - https://www.amazon.com/gp/product/1783555130/ref=dbs_a_def_rwt_bibl_vppi_i1

- *The Elements of Statistical Learning: Data Mining, Inference, and* Prediction by Trevor Hastie, Robert Tibshirani, and Jerome Friedman
  - Second Edition, ISBN: 978-0-387-84857-0, e-ISBN: 978-0-387-84858-7
  - Protected Link to eBook: https://www.dropbox.com/s/5ychxomsgl085as/statistical_learning_book.pdf?dl=0

- *Applied Predictive Modeling* by Max Kuhn and Kjell Johnson
  - ISBN: 978-1-4614-6848-6, e-ISBN: 978-1-4614-6849-3
  - Protected Link to eBook: https://www.dropbox.com/s/oe34ijztit7jg9l/springer%20book.pdf?dl=0

- *Mining of Massive Datasets* by Jure Leskovec, Anand Rajaraman, and Jeffrey D. Ullman
  - Link to Free text: https://www.dropbox.com/s/owd3jjf6t0jfaee/MassiveDatasets.pdf?dl=0

This class introduces the processes of managing, exploring, visualizing, and acting on large amounts of data. This course provides an introduction to data-mining techniques (classification, regression, association mining, clustering, and user/item recommendation) used in analytics. All material covered will be reinforced through hands-on experience using state-of-the art tools to design and execute data mining processes. Class examples will come from the programming language Python (and some examples from R). Pre-requisite courses for this class include basic statistics and probability, linear algebra, calculus, and introductory algorithm analysis (or desire to learn quickly). Experience with databases is helpful but not required.

Assignments will require the python programming language but can also be completed in R or SAS (mostly). Assignments are due in the format of "reports" that include code, visualizations, and summarize all analysis with text or video. **Please note that these assignments are quite comprehensive—student feedback has revealed that they typically take more preparation time than previous courses in data science you may have taken.** An example report and the graded feedback for the report will also be given prior to the submission of the first assignment. This is by design because the assignments are the major components of the course grade.

Learning Outcomes

This course is constructed to help students design and use machine learning and data mining techniques. Students will hone their abilities to analyze data, visualize and explain data, and predict outcomes using various learning algorithms. Various techniques for mining rules and building production ready recommendation systems will be discussed. Tools for working with massively data parallel systems will be discussed, as well as techniques for working with massive data sets out of core. While this course is designed for breadth of topics, some topics will be discussed in depth (such as support vector machines). Finally, students will learn to communicate ideas about these technical areas effectively.

Topics covered include:

- Data analysis in python (scikit-learn, pandas, dato, and jupyter)
- Visualization using matplotlib, seaborn, and plotly
- Feature dimension reduction and feature manipulation
- Introduction to common features used in Data Mining
- Linear and non-parametric regression
- Classification techniques (logistic regression, neural networks, SVMs, nearest neighbors, decision trees, ensemble methods)
- Gradient-based optimization techniques (batch, mini-batch, and stochastic gradients)
- Clustering techniques (k-means, DBSCAN, spectral clustering)
- Recommender systems (collaborative filtering, latent factors, matrix factorization) • Association rule mining
- Parallelism complications are also presented

Assignments

Assignment Guidelines

- Due to the resubmission policy, no late work is accepted.
- All work must be submitted in file format on 2DS by the due date.
- Links to work (github, etc.) submitted on 2DS are not acceptable.
- Please have multiple team members submit work to ensure there are no potential issues with submission failures or missing files.
- As a last resort, you may email your submission to ensure it is marked by the due date. However, at least one team member will eventually have to upload the file on 2DS to receive a grade.
- It is your team's responsibility to ensure that all members submit the same version of a file. It is the grader's choice as to which file will be selected for grading.
- In almost all cases, deductions will be applied to each team member's individual grade.

Group Assignments

- All work in this class is team based.
- You must form a team of 3-4 students or you will be assigned to a team before Class 3.
- Failure to work in the final groups selected for any lab assignment will result in a **30-point deduction** from your individual lab grade.
- You may only change groups in the event that another student in your class is willing to switch groups with you.

Lab Assignments

Periodically, lab assignments will be submitted electronically. Lab assignments can be completed in teams of two or three. Late labs will not be accepted and are not eligible for resubmission (see next section). Three lab assignments are due throughout the course and one "mini-lab assignment" worth fewer points (see next section for grading).

Lab assignments should be turned in as one of the three following options:

1. Rendered Jupyter notebooks (*.ipynb files), Most preferred method
2. Embedded HTML (*.htm or *.html) rendering of the Jupyter notebooks (with all images either in a zipped directory with the master HTML file or embedded in the HTML). Please use "zip" to compress directories (*i.e.*, try to avoid *.tar or *.rar because they are harder to view in the online system)
3. PDF documents in report format with all visualizations, code, and analysis included

Lab assignments are posted at the beginning of the course. Most assignments are turned in during a week where formal video/lecture does not take place (except for the mini-lab project). Use this extra time to complete extended or time-consuming analyses of the data. There is a high expectation for these assignments. Comment all code and explain reasoning in detail in the document. An example graded assignment will be given before the first lab is due.

Live-Session Assignments

Five times during the course we will hold a “Flipped Lecture Assignment” where videos are posted/watched before the live session and we will use class time to complete a live coding assignment. The specifications for the assignment will be given at the start of session and the assignment will then be turned in at the end of session. Students will work in assigned teams and turn in the live-session assignments at the end of the session. Come prepared to work! Live session assignments will be given in the form of a Jupyter notebook with blank spaces for questions and blank spaces for code to be filled in. The live session assignment is completed in three parts, as follows:

1. An initial notebook with a few questions will be given out. This notebook should be answered before coming to class the day of the assignment.
2. The instructor will work out the problems with you during the first 15 minutes of class.
3. The remainder of the assignment will then be handed out and teams can work on the assignment until the end of class.

As a team, you will answer the questions to the best of your ability. Live-session assignments should be turned in as rendered Jupyter Notebooks (exactly like the preferred method for lab assignments). These assignments are meant to be formative and facilitate one-on-one interactions with the instructor. Ask questions often and solicit feedback on given answers from the instructor!

If working as a team, all team members will need to be present to receive a grade. An absence during a “live-session assignment” cannot be made up after the assignment has been given in session. However, certain arrangements (in restricted circumstances) can be made beforehand to take the assignment early. For example, known cleared absences like conference/work-related travel, travel related to university sanctioned extracurricular activities, etc. Exceptions will be made for unexpected circumstances such as death/hospitalization of a family member.

Grading

Team coding is highly encouraged in this course. **All assignments can be completed as a team.** For team portions of assignments, one grade will be issued to the entire team, not individually (that is, the team will sink or swim together for that portion of the assignment). This helps to prevent any issues cropping up about copying/cheating from individuals in the class. Teams can change between assignments. Please handle team conflicts within your team in an appropriate manner. For example, in the case where an individual is not contributing to the team effort, that member can be told that he/she will no longer be part of a given team (well in advance of the assignment). This should be used only where appropriate and in a manner that is cohesive with team inclusiveness.

Teams will be evaluated based upon their lab assignments and live session assignments, as follows:

- Lab assignments:
    - 70% of grade (3 labs @ 20% each, 1 mini-lab @ 10%)
        - Lab One: Visualization and Data Preprocessing
        - Mini-Lab: Logistic Regression and SVMs

- Lab Two: Classification
- Lab Three: Clustering, Association Rules, or Recommenders

- Live-Session Assignments:
  - 30% of grade (5 at 6% each)
    - Linear Regression in Python
    - Decision Trees and Splitting Criteria
    - KNN, Naive Bayes, and Document Classification
    - Clustering Spatial Data
    - Association Rules in the Titanic

Lab Assignments Grading Schema: In each lab assignment, specific deliverables are asked and should be completed to the best of your ability. Each deliverable will be worth a certain percentage of the grade and you will be graded in terms of the quality of your analysis. Comment code so that it is readable and, where possible, immediately understandable. The sum total of these deliverables will be 90% of the points possible for each assignment. If you complete all the project deliverables satisfactorily you should expect a grade of 90%.

The remaining 10% of the points are reserved for **exceptional work that is above and beyond in one or more elements of the analysis.** The instructor will provide some ideas regarding exceptional work for each lab assignment, but students have free reign to provide whatever additional analysis they prefer. Grading of the exceptional work does not follow a strict rubric. It is subjective to the instructor and at the instructor's discretion.

Live-Session Assignment Grading Schema: These assignments are completed under time pressure. I understand this and will grade them accordingly. It is not my wish for these assignments to be "evaluations" of your learning, but, instead, be tools for motivating you to understand the material. **Use these assignments to ask questions to the instructor and get one-on-one feedback!** Because of time pressure, I will drop the lowest live-session assignment grade and replace it with the average of the remaining four. This allowance will not be given for live-session assignments that are not completed (for example, as the result of an unexcused absence). In other words, a 0% cannot be replaced or dropped.

Resubmissions: *Live-session assignments cannot be resubmitted under any circumstances***.** However, this is a challenging class taken at an advanced pace. Because of this, lab assignments can be resubmitted if you are unhappy with your performance. Resubmissions have some strict stipulations:

1. Resubmissions are due one week after receiving your grade and instructor feedback.
2. In a resubmission, you should fix the area(s) that the instructor pointed out were deficient. For each deliverable you can recover up to 90% of the points for that deliverable in a resubmission. Do not change deliverables that you already have received 90% or more of the points for as you cannot receive any more credit for these deliverables.
3. You should visibly highlight parts of the submission that are new and summarize the changes at the top of the resubmitted document. Changes to the document that are not documented at the top will not be regraded.
4. Resubmissions are not eligible for "exceptional work." Any points given in the original assignment for exceptional work will be removed from resubmitted assignment. This is

done to disincentive students from performing exceptional work at the expense of the deliverables in the assignment.

5. Because of stipulations 2 and 4 the typical maximum grade for resubmissions is 80%. However, it is possible to get above 80%. This happens when you have received full credit for some deliverables in the original submission and you do not change them.
6. The final lab assignment cannot be regraded because there is no time to resubmit/regrade in the final week of the course.
7. The instructor reserves the right to not allow resubmission for assignments that are so poor or so convoluted that it appears a "reasonable effort" was not made. Labs turned in late are also not eligible for resubmission.

Because of the strict stipulations in resubmitting, it is often not advisable to resubmit assignments. However, I believe it can be a very valuable learning experience to try at something you failed at originally. It is my belief that not allowing resubmission after feedback is given tells you "this assignment wasn't that important anyway." The stipulations above are meant to guard against students trying to "game" the resubmit system (not penalize resubmitting). Even still, the best way to get a good lab grade is to effectively complete the assignment in the first submission.

Curves: A dividing line for converting percentages to letter grades must be taken in order to be in line with an academic institution's grading schema. The dividing lines for this course are listed below. Even so, please do not measure your performance in this class by the letter grade you receive. Your potential employers might care very little that you got a B or C in this class if you can knowledgeably talk about data mining systems. Also, I am happy to write recommendation letters for those that performed well in the class (regardless of your grade—performance is different than a letter grade).

A	93.000 – 100
A-	90.000 – 92.999
B+	87.000 – 89.999
B	83.000 – 86.999
B-	80.000 – 82.999
C+	77.000 – 79.999
C	73.000 – 76.999
C-	70.000 – 72.999
D+	67.000 – 69.999
D	63.000 – 66.999
D-	60.000 – 62.999
F	Else

Absences

Class session attendance is required. Students with three absences or less, who actively participate in class, will not receive any deductions for his/her absence. Starting with the fourth absence, 2% points from the final grade will be deducted for each absence (over the initial three absences). Please note: Rarely are these measures needed!

Cheating

Cheating of any kind such as plagiarism or direct copying is strictly prohibited and against the SMU honor code. However, collaboration is strongly encouraged. All lab assignments can be done as a group and turned in as a group.

A note on cheating: You can and should look online for others that have analyzed data in Jupyter notebooks. However, if you start with their code or use it, you should 100% make it clear where you got it and what you changed (some of the best performers in this class have taken this approach). However, the reference to other work should be included **directly next to the code or visualization** that you are using (that is, including a list of references at bottom of the document and not making it clear where you got parts of code/images/etc. will be considered cheating and plagiarism). Don't try to make it such that the cheating is "plausibly deniable." That is not a defense for behaving unethically and could result in expulsion from the SMU program. When quoting directly from another source, please place that text in quotes and cite the quote immediately. Failure to place text in quotes from other sources will be considered cheating.

I do not mess around here. I have seen a lot of questionable copying of code/text without attribution in Computer Science in recent years. Please attribute where it came from! Many times, this copying is the result of not allocating enough time to complete the assignment, or a fear that you cannot get a good grade if you let me know that you are struggling with basic concepts. **It is better to try and fail in these circumstances than to turn in unattributed code.** In the event of plagiarism or cheating, I will turn the matter over to SMU's honor code committee, which could result in expulsion from the University. That is a daunting consequence, especially when I allow resubmissions for the course.

Another note on cheating: I understand small mistakes. Like having a small "similar" snippet of code that you forgot to copy the link to in your assignment (especially when your past performance indicates this was just an "oops, I forgot!"). I am not in the business of failing people for minor missteps, nor would I want to be. These measures are in place for egregious or obvious violations. Most of the assignment in this course, because they are contemporary issues, are hard to cheat.

Disability Accommodations

Students needing academic accommodations for a disability must first be registered with Disability Accommodations & Success Strategies (DASS) to verify the disability and to establish eligibility for accommodations. Students may call 214-768-1470 or visit http://www.smu.edu/alec/dass.asp to begin the process. Once registered, students should then schedule an appointment with the professor to make appropriate arrangements.

Religious Observance

Religiously observant students wishing to be absent on holidays that require missing class should notify their professors in writing at the beginning of the semester, and should discuss with them, in advance, acceptable ways of making up any work missed because of the absence. (See University Policy No. 1.9.)

Excused Absences for University Extracurricular Activities

Students participating in an officially sanctioned, scheduled University extracurricular activity should be given the opportunity to make up class assignments or other graded assignments missed as a result of their participation. It is the responsibility of the student to make arrangements with the instructor prior to any missed scheduled examination or other missed assignment for making up the work. (See the University Undergraduate Catalog for details.)

Please note that this syllabus is subject to change. Any changes to the syllabus will be announced.