Introduction to Programming and Predictive Analytics for Business

IFI 8410; Spring 2022

Wednesdays, 6:00 – 8:30 p.m. [online]

Instructor:
Dr. Meng Zhao
Institute for Insight, Robinson College of Business
Office: Room 304
Email: mengzhao@gsu.edu

WebEx information:

Link: https://gsumeetings.webex.com/meet/mengzhao
Dial in: +1-415-655-0002
Access code: 120 628 8748

Office hours via WebEx:
Fridays, 4:00 – 5:00 p.m.
Also available by appointment

Prerequisites: None.

Catalog Course Description: This course introduces students to the science of business analytics and covers foundational material needed to use and apply analytic techniques to real-world business challenges. Students will learn to identify the appropriate tools to collect, analyze, and visualize data and utilize data in decision making. Examples will illustrate the use of predictive analytics to solve business problems such as reducing customer churn, customer segmentation, predicting market demand, forecasting stock prices, etc. Both structured and unstructured data will be used throughout the course.

Student Learning Outcomes:
By the end of the semester students will be able to:

- Frame a business problem using predictive analytics;
- Ascertain how and when to use regression, classification, and cluster methods to address business issues;
- Choose and implement the right packages in Python to build predictive models;
- Effectively interpret, visualize, and present the results of analyses.

Required and Recommended Texts and Downloads:

- Required downloads:
  - Python and Jupyter Notebook via Anaconda distribution: https://www.anaconda.com/products/individual
- Required readings assigned and provided by instructor
- Recommended texts:
  - Thomas W. Miller, *Modeling Techniques in Predictive Analytics With Python and R: A Guide to Data Science* (Pearson 2019)
  - Wes McKinney, *Python for Data Analysis* (O'Reilly, 2nd edition)

Assessment:

Item	*Percentage*
Homework assignments – individual	15%
Homework assignments – group	15%
Mid-semester project presentation	25%
Final project presentation	35%
Peer evaluation (weighted)	10%

Grading Scale:

A+	A	A -	B+	B	B-	C+	C	C-	D	F
97.0%-100%	91.0 – 96.9 %	89.5 – 90.9%	87.0 – 89.4%	83.0 – 86.9%	79.5 – 82.9%	77.0 – 79.4%	72.0 – 76.9%	69.5 – 71.9%	60.0 – 69.4%	Below 59.9%

Course Design

Course Structure: This course introduces students to a variety of methods and topics in programming and predictive analytics while also providing extensive hands-on opportunities to work with data. Most code will be in Python; we may work with parallel code in R in some weeks to expose students to both programming environments.

Students will learn new methods and techniques each week via recorded lectures, example code, and instructor-provided data sets. Working in teams and meeting weekly with the instructor, students will then apply these methods to a semester-long project using either 1) a data set that students provide, or 2) an instructor-provided data set.

On most weeks, there will be an individual or group homework assignment. Other weeks will be set aside for project work in teams with opportunities for check-ins and guidance by the instructor.

Teams will present their projects in-progress in the middle of the semester and their final project at the end. All teams will watch and give feedback on all other teams' presentations.

Team Management: Early in the semester, teams will form. If there are problems during the semester, the following methods will be used:

Terminating team members: As in any organization, there may be people in your group who are not willing or able to perform to the level of excellence demanded by the team. The process used to improve team member performance and/or to terminate a team member's membership in the team will involve the following steps:

- Discuss the poor performance with the individual and the standards he or she is expected to meet. As a team, document the discussion including all members' agreed-upon understanding of the standards of performance and the individual's shortfall from those standards. The document should describe what the individual must do to meet the team's standards and the time frame in which the individual will come up to the standards. This agreement should be signed by all team members, and a copy should be sent to the instructors.
- If the agreement is not met, the team, including the individual in question, will schedule a meeting with the faculty. The team will bring a copy of the contract to the meeting for the faculty and will discuss the individual's performance with the faculty. The individual will be terminated or given a final chance to improve his or her performance during that meeting and within a given time frame.
- If the performance does not improve within the time frame, the individual will be terminated from the team.
- If the individual is terminated, the individual may seek to join another team. Alternatively, he or she must complete all course work in its entirety by himself or herself from that point forward.

Resigning from a team: A student may resign from a team and switch to a different one. The work that was done while a team member is the property of both the team and the individual so all can use the work product. Faculty will facilitate the placement of the resigning person on a different team.

Peer evaluation: At the end of the semester, each student must complete an evaluation of his or her teammates'

participation in the group. Peer evaluation results are worth 10% of the final course grade. Students must complete this evaluation in order to receive any points for their own group participation grades. Students may not decrease a team member's peer evaluation score without first having addressed deficiencies in that team member's performance directly with that team member, including via the process described above.

Course Policies

iCollege: The syllabus, recorded lectures and slides, code, data sets, and other course material will be posted on iCollege. Students will also use iCollege to access course announcements, upload assignments, and view grades. We will use Piazza to ask/answer questions. iCollege and Piazza are linked to your GSU student email account. If you do not check that account regularly, forward messages and notifications to an account that you do check.

WebEx: Team, all-class, and office hour meetings will be conducted via WebEx. All cameras must be on during WebEx meetings and microphones should stay on mute unless you are speaking.

Instructor and TA Communication: The instructor and TA will be available for student questions and support during weekly office hours. Outside of office hours, students should first use Piazza (via iCollege) to post questions. If you have not received an answer from your classmates via Piazza, *only then* should you contact the instructor. Allow one full business day for a response.

Late Work and Make-Up Policy: Any request to submit late or make-up work must be accompanied by sufficient documentation of the reason for the request. It is at the instructor's discretion whether to accept late or make-up work. Not having enough time to complete the assignment or last-minute computer problems will not, alone, excuse late work. Requests for extensions or make-up work should be made prior to the missed deadline or class, to the extent possible. There are no opportunities for additional projects or extra credit work.

University Policies

Ethics and Academic Honesty: GSU takes issues of ethics and academic honesty very seriously. Students are expected to recognize and uphold standards of intellectual and academic integrity in all work. The University policy on academic dishonesty is spelled out in Section 1350 of the Graduate Catalog. Lack of knowledge is not an acceptable defense to any charge of academic dishonesty. Violations will result, at a minimum, in a zero for the assignment and can result in expulsion from the university. The following are instances of academic dishonesty:

- Plagiarism
- Cheating on examinations
- Unauthorized collaboration with others
- Falsification of materials
- Multiple submissions (i.e., submitting the same work for credit in more than one class)

Accommodations: Students who wish to request accommodation for a disability may do so by registering with the Access and Accommodation Center. Students may only be accommodated upon issuance by the Access and Accommodation Center of a signed Accommodation Plan and are responsible for providing a copy of that plan to instructors of all classes in which accommodations are sought.

Student Course Evaluations: Your constructive assessment of this course plays an indispensable role in shaping education at Georgia State. Upon completing the course, please take time to fill out the online course evaluation.

Course Schedule and Topics

General plan for each week:

- Before class time, students are responsible for watching the recorded lecture, reviewing readings and resources, learning the method(s), and running the example code on the relevant data set.

- **During class time**, each team will have a separate scheduled meeting time via WebEx with the instructor to go over the new method(s) and code, address questions, and collaborate on the team's ongoing project. There will be at least **three all-class meetings** via WebEx, at the beginning, middle, and end of the semester, during class time in place of scheduled team meetings.
- **Between classes**, students are responsible for completing the homework assignments (group and/or individual) and advancing work on the semester-long team project. The instructor and TA will be available for assistance during virtual office hours and students are also encouraged to assist one another via Groupme.

The course syllabus provides a general plan for the course; deviations may be necessary.

Code and slides posted (Fridays)	**Before class**	**During class** (Wednesdays)	**Between classes** (All homework due via iCollege by next class start time)
Week 1: Course introduction; Data assessment; Data ethics and bias			
1/10	Readings and resources: • Miller excerpt Data: company_reviews Data source documentation: https://www.kaggle.com/fireball684/hackerearthericsson	1/12 Introduction to the course and instructors. Introduction to predictive analytics; Python, Jupyter Notebook, and R; data ethics and bias; variable types. Walk-through iCollege. Create LinkedIn profile, mark yourself as open to work and connect.	Team formation and team meeting scheduling [via email from instructor] HW 1 [individual]: Install Python and Jupyter Notebook via the Anaconda distribution Write a 1-2 page report in which you: • Identify the variable type for each column; • Describe any data ethics or bias issues associated with the data, drawing on the Kaggle source documentation, and how you might correct or address them; and • Describe 2-5 research questions you might ask and answer using this data set.
Week 2: Data exploration and visualization in Python			
1/14	Week 2 code and slides Data: company_reviews	1/19 Verify Python and Jupyter Notebook installation during team meetings. Review data exploration and visualization methods.	HW 2 [individual]: Using the company reviews data set, write a 1-2 page report in which you: • Generate summary statistics for all continuous and categorical variables; • Generate a bar chart, different from the example used in the instructions;

			• Generate a scatter plot, different from the example used in the instructions; and • Interpret all results. HW 2 [group]: Turn in one report per group covering: • Team meeting and communication plan • Choice of a data set from students' own data, subject to instructor approval, or instructor-provided options
Week 3: Working with variables: dummies and binning			
1/21	Week 3 code and slides Data: • company_reviews for the week 3 code • your own team's data set for the homework	1/26 Data manipulation: • Dummies • Binning • Other data wrangling • Saving your work – "pickling"	HW 3 [group]: Turn in a Jupyter Notebook (upload the .ipynb file to iCollege), one per team, in which you: • Ingest your team's data set. • Generate summary statistics for all continuous and categorical variables. • Choose a continuous variable and bin it. Explain in the comments (#) why you have chosen to bin the variable in the way that you did. • Choose a categorical variable and convert it to dummies. • Convert a text categorical variable to a numerical categorical variable, or vice versa if you don't have a text categorical variable. • Use groupby, crosstabs, and bar plots to explore differences across groupings within your data. Run at least one groupby function and at least one crosstab function and plot the results.
Week 4: Linear and logistic regression for interpretation			
1/28	Week 4 code and slides. Data: • company_reviews for the week 4 code	2/2 Linear and logistic regression for interpretation	HW 4 [group]: Turn in a Colab document or Jupyter Notebook, one per group, in which you: • Ingest your team's data set (from .csv or a saved set of

	• your own team's data set for the homework		objects from a previous Python session) • Run a linear OLS regression using a continuous variable as the dependent variable and 2-5 other variables as independent variables. Convert variables to the required form as needed. • Run a logistic regression using a binary variable as the dependent variable and 2-5 other variables as independent variables. Convert variables to the required form as needed. Convert the log odds coefficients to marginal effects. • Include as #comments in a cell in your notebook your group's answers to the following questions for **each regression**: Which results are statistically significant? Focusing on those statistically significant results, what do the regression tables tell you about the change in *y* associated with a one-unit increase in the *x*'s? (remember to interpret the marginal effects for the logistic regression results, not the log odds)
Week 5: Linear and logistic regression for prediction; Performance measures			
2/4	Week 5 code and slides. Data: • company_reviews for the week 5 code • your own team's data set for the homework	2/9 Linear and logistic regression for prediction Mean squared prediction error Confusion matrix and performance evaluation	HW 5 [individual]: Turn in a Colab document or Jupyter Notebook, one per student, in which you: • Ingest your team's data set (from .csv or a saved set of objects from a previous Python session) • Create training and test sets. • Run a linear OLS regression using a continuous variable as the dependent variable and set of independent variables of your choosing. Convert variables to the required form as needed.

			• Run the code to eyeball and visualize the actual versus predicted *y*'s. • Calculate mean squared prediction error. • Run a logistic regression using a binary variable as the dependent variable and a set of independent variables of your choosing. Convert variables to the required form as needed, and re-create training and test sets if necessary. • Run the code to eyeball the actual versus predicted *y*'s. • Generate a confusion matrix, precision score, recall score, and F1 score. • Include as #comments in a cell in your notebook your assessment of how well your two models did in predicting the values of *y.* How might you achieve even better predictive performance?
Week 6: Linear and logistic regression for prediction; Variable selection and overfitting			
2/11	Week 6 code and slides. Data: • company_reviews for the week 6 code • your own team's data set for the homework	2/16 Linear and logistic regression special issues Variable selection Overfitting	HW 6 [individual]: Turn in a Colab document or Jupyter Notebook, one per student, in which you: • Ingest your team's data set (from .csv or a saved set of objects from a previous Python session) • Using your team's dependent variable of choice and *either* linear or logistic regression, run at least 2 automated variable selection processes. • Compare the independent variables that are selected from the different approaches, along with the regression statistics. • Combining the automated results with your own subject matter knowledge and intuition, choose a final set of independent variables (x's).

			• Re-run your regression for interpretation (generate an output table) and prediction (generate prediction performance measures). • Include as #comments in a cell in your notebook your assessment of how well your model performed for both interpretation and prediction using your new set of x's. Note: This HW is due on 3/2 (week 8), by 6:00 p.m. For class on 2/23, prepare your group's mid-semester team project presentation.
Week 7: Mid-semester team project presentations			
2/18		2/23 Team presentations 6:00 p.m.	
Week 8: Decision trees and random forest			
2/25	Week 8 code and slides. Data: • company_reviews for the week 8 code • your own team's data set for the homework	3/2 Decision trees, random forest, algorithm horseraces Visualization Variable importance	HW 7 [group] Turn in a Colab document or Jupyter Notebook, one per group, in which you: • Ingest your team's data set (from .csv or a saved set of objects from a previous Python session) • Using your team's dependent variable of choice and the **best set of x's** that you have identified from your previous weeks' work, run *either* a linear or logistic regression and compute the mean squared prediction error (for linear) or F1 score (for logistic). • Then run the appropriate decision tree or random forest code for your continuous or categorical independent variable, and compute the mean squared prediction error (for continuous) or F1 score (for binary). • Compare all three scores and determine which model is the best predictor for your data. Which model won the horse race?

			• Include your observations as #comments in a cell in your notebook.
Week 9: K-nearest neighbor; targeted predictions			
3/4	Week 9 code and slides. Data: • company_reviews for the week 9 code • your own team's data set for the homework	3/9 KNN Targeted predictions	HW 8 [group] Turn in a Colab document or Jupyter Notebook, one per group, in which you: • Ingest your team's data set (from .csv or a saved set of objects from a previous Python session) • Use your team's dependent variable of interest and re-run the algorithm horserace from last week, but now add KNN. • Use regression coefficients and p-values, variable importance, and/or targeted predictions to identify the most influential *x*'s in your model. • Include your observations about the best performing predictive model and the most important independent variables as #comments in a cell in your notebook.
Week 10: K-means clustering			
3/18	Week 10 code and slides. Data: • processed.cleveland.data for the week 10 code • your own team's data set for the homework	3/23 k-means clustering	HW 9 [group] Turn in a Colab document or Jupyter Notebook, one per group, in which you: • Ingest your team's data set (from .csv or a saved set of objects from a previous Python session) • Isolate all continuous variables in your data set. • Determine the optimal k and run k-means clustering on your continuous variables. • Add your cluster numbers back to your original data set and use plots and exploratory data analysis to explore cluster characteristics. • Include your observations about clustering as #comments in a cell in your notebook. Did clustering produce anything

			interesting or useful for your group?

Week 11: Text analysis			
3/25	Week 11 code and slides. Data: • company_reviews for the week 11 code • your own team's data set for the homework or wine_reviews.csv if your data set has no text	3/30 Text analysis	HW 10 [group] Turn in a Colab document or Jupyter Notebook, one per group, in which you: • Ingest your team's data set (from .csv or a saved set of objects from a previous Python session) OR wine_reviews.csv if your data set has no text. • The wine_reviews.csv file has text in the "description" column and a binary dependent variable in the "score_category" column. • Clean and preprocess your text. • Generate text frequency tables and word clouds to compare two subsets of your data (score_category) for wine_reviews; choose subsets in your own data. • Run a logistic regression, decision tree, and random forest and assess your model's performance. • Include your observations about your analyses as #comments in a cell in your notebook.
Week 12: Project preparation			
	Final slide template.	4/6 Numpy and Linear Algebra Operations Pandas data frames, loading data, data-table manipulation **OR Skip for optional team meetings.**	Prepare for final presentation.
Week 13: Project preparation			
	Final slide template.	4/13 No new material; extensions per team. **Optional team meetings.**	Prepare for final presentation.
Week 14: Final team project presentations			
		4/20 **Team presentations @ 6:00 p.m.**	
Peer evaluation due 4/27 @ 6:00 p.m.			