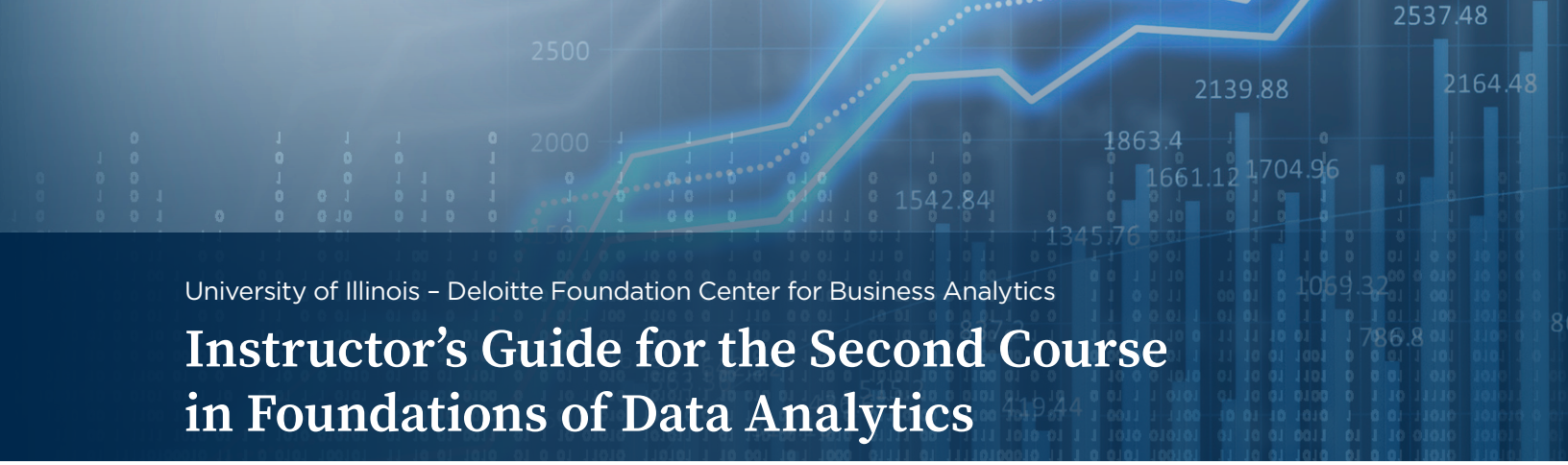

University of Illinois – Deloitte Foundation Center for Business Analytics

Instructor's Guide for the Second Course in Foundations of Data Analytics

Course Description

This course will build a practical foundation for machine learning by teaching students basic tools and techniques that can scale to large computational systems and massive data sets. This course follows the first course in the Foundations of Data Analytics series to teach students to draw inferences from large, complex, and diverse data. This course contains eight modules that introduce machine learning with a focus on business applications. This course will introduce the scikit-learn and Statsmodels Python modules, while also demonstrating new applications of the NumPy, Pandas, Matplotlib, and Seaborn modules.

Course Organization

This course is organized into eight modules. Each module consists of multiple lessons, which each contain an associated video explaining the lesson content, one or more external readings, and included course Jupyter notebooks. Each module also includes a quiz (or assessment) that tests basic mastery of the lesson contents and a programming assignment that tests synthesis of the lesson contents, especially into the broader module and course content.

Typically, a module is covered in one week, although this pace can be faster or slower based on student needs. In addition, content can be selectively used from lessons and modules to supplement an existing course. The module quiz is traditionally given through the course LMS in a timed fashion (e.g., thirty minutes). The assignments are completed on the course JupyterHub server, and are collected at the assignment deadline and processed by an autograder.

ILLINOIS
Gies College of Business

Special Notes

Software Versions

This course employs the Python based Scikit Learn module to build and test machine learning models. This open source software module is undergoing continual development, and occasionally new versions of the software will cause the code in the course notebooks to throw warnings and potentially cause error conditions that might prevent a notebook from working. The notebooks in this course release have been tested with the following versions of Python and associated Python library (note, there is a special notebook called test-version.ipynb in the modified directory that will display your installed software versions for comparison).

```
3.6.7 | packaged by conda-forge | (default, Nov 21 2018, 02:32:25)
[GCC 4.8.2 20140120 (Red Hat 4.8.2-15)]

Numpy version         = 1.15.4
Pandas version        = 0.23.4
Scikit Learn version  = 0.20.2
Matplotlib version    = 0.9.0
Seaborn version       = 3.0.2
Statsmodels version   = 0.9.0
```

Data Storage

The machine learning notebooks in this course are designed to use data files stored locally; however, if the data file is not found, the file will be pulled automatically from an internet archive to the local file system. The notebooks are hardcoded to look for the data files in the data directory, which is expected to exist in the home directory of the current user. For example, on a Mac OSX computer with a username Joe , the directory /Users/Joe/data . This directory will need to be created for the notebooks to successfully download and store the data files.

Course Contents

Module 1: Introduction to Machine Learning

This module provides the basis for the rest of the course by introducing the basic concepts behind machine learning, and, specifically, how to perform machine learning by using Python and the scikit-learn machine learning module. First, you will learn how machine learning and artificial intelligence are disrupting businesses. Next, you will learn about the basic types of machine learning and how to leverage these algorithms in a Python script. Third, you will learn how linear regression can be considered a machine learning problem with parameters that must be determined computationally by minimizing a *cost* function. Finally, you will learn about neighbor-based algorithms, including the knearest neighbor algorithm, which can be used for both classification and regression tasks.

Learning Objectives

By the end of this module, you should be able to:

- Articulate the different types of machine learning algorithms and provide examples where each type might be applied within the accounting profession
- Describe the challenges in cleaning and pre-processing data
- Apply the scikit-learn Python module to perform basic machine learning tasks
- Understand the importance of a cost function and the importance of minimizing this function
- Perform general linear regression by using the scikit-learn module
- Apply the k-nearest neighbor algorithm for both classification and regression tasks

Module 2: Fundamental Algorithms

This module introduces several of the most important machine learning algorithms: logistic regression, decision trees, and support vector machine. Of these three algorithms, the first, logistic regression, is a classification algorithm (despite its name). The other two, however, can be used for either classification or regression tasks. Thus, this module will dive deeper into the concept of machine classification, where algorithms learn from existing, labeled data to classify new, unseen data into specific categories; and, the concept of machine regression, where algorithms learn a model from data to make predictions for new, unseen data. While these algorithms all differ in their mathematical underpinnings, they are often used for classifying numerical, text, and image data or performing regression in a variety of domains. This module will also review different techniques for quantifying the performance of a classification and regression algorithms and also how to deal with imbalanced training data.

Learning Objectives

By the end of this module, you should be able to:

- Articulate the basic process of machine classification and describe how it can be applied to a specific predictive challenge,
- Articulate the basic concept of machine regression and describe how it can be applied to a specific predictive challenge
- Describe the basic concepts behind the logistic regression, decision tree, and support vector machine algorithms
- Apply the logistic regression, decision tree, and support vector machine algorithms to classification challenges by writing Python analysis scripts that use the scikit-learn library
- Apply the decision tree and support vector machine algorithms to regression challenges by writing Python analysis scripts that use the scikit-learn library,
- Quantify the performance of a machine classification and machine regression analysis
- Work effectively with imbalanced training classes

Module 3: Practical Concepts in Machine Learning

This module introduces several important and practical concepts in machine learning. First, you will learn about the challenges inherent in applying data analytics (and machine learning in particular) to real world data sets. This also introduces several methodologies that you may encounter in the future that dictate how to approach, tackle, and deploy data analytic solutions. Next, you will learn about a powerful technique to combine the predictions from many weak learners to make a better prediction via a process known as ensemble learning. Specifically, this module will introduce two of the most popular ensemble learning techniques: bagging and boosting, and it will demonstrate how to employ them in a Python data analytics script. Finally, the concept of a machine learning pipeline is introduced, which encapsulates the process of creating, deploying, and reusing machine learning models.

Learning Objectives

By the end of this module, you should be able to:

- Articulate the differences between performing data analytics in this course and the real-world,
- Understand the basic concepts behind ensemble learning, bagging, and boosting,
- Develop a data analytics script that employs ensemble learning in a Python script by using the scikit-learn module
- Construct a machine learning pipeline by using the scikit-learn module

Module 4: Overfitting and Regularization

This module introduces the concept of regularization, problems it can cause in machine learning analyses, and techniques to overcome it. First, the basic concept of overfitting is presented along with ways to identify its occurrence. Next, the technique of cross-validation is introduced, which can mitigate the likelihood that overfitting can occur. Next, the use of cross-validation to identify the optimal parameters for a machine learning algorithm trained on a given data set is presented. Finally, the concept of regularization, where an additional penalty term is applied when determining the best machine learning model parameters, is introduced and demonstrated for different regression and classification algorithms.

Learning Objectives

By the end of this module, you should be able to:

- Articulate the issue of overfitting and explain how it can negatively impact a machine learning analysis,
- Explain the bias-variance tradeoff and its relationship to overfitting,
- Employ different cross-validation techniques to mitigate the impact of overfitting,
- Use cross-validation to select optimal model parameters when determining the best machine learning model
- Employ regularization to generate improved machine learning models

Module 5: Fundamental Probabilistic Algorithms

This module starts by discussing practical machine learning workflows that are deployed in production environments, which emphasize the big picture view of machine learning. Next this module introduces two additional fundamental algorithms: naive Bayes and Gaussian Processes. These algorithms both have foundations in probability theory but operate under very different assumptions. Naive Bayes is generally used for classification tasks, while Gaussian Processes are generally used for regression tasks. This module also discusses practical issues in constructing machine learning workflows.

Learning Objectives

By the end of this module, you should be able to:

- Describe the steps required to build a production machine learning workflow
- Describe the basic concepts behind the naive Bayes and Gaussian process algorithms
- Apply the naive Bayes and Gaussian process algorithms to machine learning challenges by writing Python analysis scripts that use the scikit-learn library

Module 6: Feature Engineering

This module introduces an important concept in machine learning: the selection of the actual features that will be used by a machine learning algorithm. Along with data cleaning, this step in the data analytics process is extremely important, yet it is often overlooked as a method for improving the overall performance of an analysis. This module beings with a discussion of ethics in machine learning, in large part because the selection of features can have (sometimes) non-obvious impacts on the final performance of an algorithm. This can be important when machine learning is applied to data in a regulated industry or when the improper application of an algorithm might lead to discrimination. The rest of this module introduces different techniques for either selecting the best features in a data set, or the construction of new features from the existing set of features.

Learning Objectives

By the end of this module, you should be able to:

- Articulate how machine learning might make *ethical* mistakes
- Indicate how machine learning models can be constructed to avoid making *ethical* mistakes
- Understand the importance of including ethics in artificial intelligence that will increasingly affect our everyday lives
- Perform feature selection in data analytics scripts written in Python that use the scikit-learn module
- Perform dimensional reduction by using the scikit-learn module
- Apply manifold learning to reduce the number of features in a large data set

Module 7: Introduction to Clustering

This module introduces clustering, where data points are assigned to larger groups of points based on some specific property, such as spatial distance or the local density of points. While humans often find clusters visually with ease in a given data sets, computationally the problem is more challenging. This module starts by exploring the basic ideas behind this unsupervised learning technique, as well as different areas where clustering can be used by businesses. Next, one of the most popular clustering techniques, K-means, is introduced. Next the density-based DB-SCAN technique is introduced. This module concludes by introducing the mixture models technique for probabilistically assigning points to clusters.

Learning Objectives

By the end of this module, you should be able to:

- Articulate that basic concepts behind clustering and provide examples where it is commonly used by businesses
- Understand the basic concepts of the K-means, DB-SCAN, and mixture model clustering techniques
- Employ these algorithms within a Python data analytics script by using the scikit-learn module

Module 8: Introduction to Anomaly Detection

This module introduces the concept of an anomaly, or outlier, and different techniques for identifying these unusual data points. First, the general concept of an anomaly is discussed and demonstrated in the business community via the detection of fraud, which in general should be an anomaly when compared to normal customers or operations. Next, statistical techniques for identifying outliers are introduced, which often involve simple descriptive statistics that can highlight data that are sufficiently far from the *norm* for a given data set. Finally, machine learning techniques, which can either classify outliers or identify points in low density (or outside normal clusters) areas as potential outliers, are reviewed

Learning Objectives

By the end of this module, you should be able to:

- Articulate the basic concepts behind outlier detection
- Provide examples where outlier detection is important to accountancy and business in general,
- Apply statistical outlier detection techniques in a Python data analytics script
- Apply machine learning based outlier detection techniques by using the scikit-learn library