Introduction to Data Science

Daniel Gutierrez, Data Scientist

Los Angeles, Calif.

Course Outcomes

- Examine the process of data munging, aka data wrangling, aka data transformation
- Present a variety of commonly used techniques to add to your data science toolbox
- You'll be able to draw upon these methods for future projects

Lesson Objectives

- Learn “data munging” aka data wrangling, data transformation
- Be able to complete data munging tasks as required by the needs of the data science process
- Build up your data science toolbox with a variety of data munging techniques
- Use your new R programming skills to transform raw data into a form usable by machine learning algorithms

Data Munging

- Feature engineering
- Creating a data pipeline

Data Munging

- Revising variable names
- Sometimes a raw data set may have variable names that are inconvenient for use in R
- During data munging, you can write specialized code to fix up variable names
- Each case may be different, but the techniques will be similar
- Use regular expressions with functions like `strsplit()`

Data Munging

- Creating new variables
- Sometimes a raw data file doesn’t contain all the variables you need. You can calculate new variables based on the ones available

Data Munging

- Discretizing numeric variables
- Discrete “ranges” of values sometimes more convenient for machine learning than continuous values.
- A `salary` field more useful as a series of 6 value ranges: $0-$10,000, $10,001-$25,000 and so on
- Use `cut()` function in R

Data Munging

- Data and time handling
- Date and time values are a potentially thorny issue when performing data munging in R
- There are many representations of data and time values in data sets
- Your R code must recognize and parse date and time values to get them into R date and time classes
- We'll use the `lubridate` package for this purpose

Data Munging

- Creating binary categorical variables
- When using certain machine learning algorithms, it is more convenient to have a categorical variable (known as a factor in R) represented as multiple binary variables (`TRUE` or `FALSE`)
- We can use `sapply()` to create a matrix containing the binary values.

Data Munging

- Merging data sets
- You may receive two or more data sets of similar structure that you need to combine together
- We can used the `merge()` function
- As with SQL, we can join data sets in the following ways: inner join, outer join, left outer join and right outer join

Data Munging

- Ordering data sets
- When examining a data set you may notice natural ordering of the data instead of a random order
- It is beneficial to order a data frame containing the data especially when browsing through records
- We can use the `order()` function for this purpose
- You can order a data frame by one or more variables

Data Munging

- Reshaping data sets
- Sometimes the data you receive is “misshapen,” i.e. the data is all there just in an inconvenient format or structure.
- You’ll need to reshape the data to make it easier to use downstream
- We can use the `melt()` function from the `reshape2` package

Data Munging

- Data manipulation using the `dplyr` package
- `dplyr` is a valuable tool for the data munging process
- It provides the means for filtering, selecting, restructuring, and aggregating tabular data in R
- `dplyr` is a large package with many functions. A whole book would be needed to explain its many features. We'll see some examples in the associated Code module

Data Munging

- Missing data is a recurring theme in data sets
- Incomplete records with variables missing or in error
- Need a plan for dealing with missing values
- Can discard incomplete records, or *impute* to infer missing values based on data from other records
- Using the `impute()` function from the `e1071` package

Data Munging

- Feature scaling
- Ranges of quantitative variables can vary, e.g. number of bedrooms is 3, while square footage is 1,800
- Difference in magnitude due to measurement units
- Feature scaling allows us to normalize values
- Why? One reason is some classifiers calculate a distance metric between data points
- Use the `scale()` function in base R `stats`

Code module

- WEEK 6-1 Code module – revising variable names
- WEEK 6-2 Code module – creating new variables
- WEEK 6-3 Code module – discretize numerics
- WEEK 6-4 Code module – date handling with `lubridate`
- WEEK 6-5 Code module – binary categorical vars
- WEEK 6-6 Code module – merge data sets
- WEEK 6-7 Code module – sort data sets
- WEEK 6-8 Code module – reshape data sets
- WEEK 6-9 Code module – data manipulation with `dplyr`
- WEEK 6-10 Code module – handling missing data
- WEEK 6-11 Code module – feature scaling

Summary

- Built the beginnings of your data munging tool box
- We saw examples of a number of useful and commonly used methods
- These techniques represent a small fraction of all the data munging methods you'll need
- From this point forward, you should work to add more tools