Wrapping Up and Next Steps

Ivan Corneillet

Data Scientist

Learning Objectives

After this lesson, you should be able to:

- Summarize the models we learned in this class and select the appropriate model for the problem at hand
- As next steps, which models you should learn on your own

Announcements and Exit Tickets

Today

Here's what's happening today:

- Announcements and Exit Tickets
- Alumni Panel
- What models did we learn in this class? Which one should I choose? Pros and Cons?
- What else did we learn in this class?
- What's next?
- Exit Tickets

Wrapping Up and Next Steps

What models did we learn in this class? Which one should I choose? Pros and Cons?

What models did we learn in this class?

Regression

- Linear Regression
- k-Nearest Neighbors
- Regression Decision Trees and Random Forests
- AR, MA, ARMA, and ARIMA

Classification

- Logistic Regression
- k-Nearest Neighbors
- Classification Decision Trees and Random Forests

What models did we learn in this class? We've also made a foray in unsupervised learning:

- Principal Component Analysis (a.k.a., PCA) for dimensionality reduction as a pre-processing step for other machine learning algorithms
- k-Means Clustering

What model should I use? Ask yourself the following questions:

- Do I have an output or not?
  - If yes, you need to use a supervised learning technique (really the focus of this class); otherwise, you will use an unsupervised technique (e.g., k-Means for clustering)
- Assuming you have a supervised learning problem, is your output a quantitative variable or qualitative
  - If it is quantitative you will use one of the regression methods; otherwise you will use a classification algorithm

What model should I use? Is your goal interpretation or prediction?

- Interpretative regression models are:
  - Linear Regression
  - Simple Regression Decision Trees
- Predictive regression models are:
  - k-Nearest Neighbors
  - Regression Random Forests
  - AR, MA, ARMA, and ARIMA. Low order of models are relatively interpretable but higher order are not. This is mainly why they are usually only used for prediction

What model should I use? Is your goal interpretation or prediction? (cont.)

- Interpretative classification models are:
  - Logistic Regression
  - Simple Classification Decision Trees
- Predictive classification models are:
  - Classification Random Forests

Activity | Knowledge Check

EXERCISE

DIRECTIONS (20 minutes)

1. For the following machine learning algorithms that we studied in this class, list the possible pros and cons of each
    a) Linear Regression
    b) k-Nearest Neighbors
    c) Logistic Regression
    d) Simple Decision Trees
    e) Random Forests
    f) AR, MA, ARMA, and ARIMA

DELIVERABLES

Answers to the above questions

Pros and Cons of Linear Regression

- Pros
  - Intuitive
  - Very interpretable
  - Easy to compute predictions
  - No need to standardize your data
- Cons
  - Assumes linear association among variables
  - Assumes normally distributed residuals
  - Outliers can easily affect coefficients

Pros and Cons of k-Nearest Neighbors

- Pros
  - Intuitive
  - Very easy to compute
  - Easily capture non-linearity
- Cons
  - Not interpretable
  - Cannot be used if you have sparse data and feature space with dimension of 4 or more
  - Need to standardize your data

Pros and Cons of Logistic Regression

- Pros
  - Fit is fast; faster than Random Forests
  - Output is a (posterior) probability which is easy to interpret
- Cons
  - Limited to binary classification (but *sklearn* provides a multiclass implementation; use ensemble under the hood)

Pros and Cons of Simple Decision Trees

- Pros
  - Intuitive
  - Very interpretable
  - Easy to compute predictions
  - No need to standardize your data
- Cons
  - Low predictable power

Pros and Cons of Random Forests

- Pros
  - Good predictive power
  - Easy to compute predictions
  - No need to standardize your data
- Cons
  - Not interpretable

Pros and Cons of AR, MA, ARMA, and ARIMA

- Pros
  - AR – good for smoothing patterns
  - MA – good for tackling shocks
  - ARMA – good for smoothing patterns and tackling shocks
  - ARIMA – good for smoothing patterns and tackling shocks; also takes care of linear trends in the model

- Cons
  - Not that interpretable (except when p and q are small)

What else did we learn in this class?

What else did we learn in this class? (cont.)

- Over the course, we improved our Python fluency
  - *pandas* DataFrames and other Python data structures (e.g., dictionaries)
  - Write basic functions to simplify our life and avoid code duplication (e.g., transforming variables on the training set then on the testing set)

What else did we learn in this class? (cont.)

- We are no longer afraid of statistics! You should feel at home now with:
  - (Two-Tail) Hypothesis Testing
  - Normal, Student's t-, and F-distributions
  - t-values and p-values

What else did we learn in this class?

- We discussed how important it was to tidying up data
  - Tidying data is one of the most fruitful skill you can learn as a data scientist. It will save you hours of time and make your data much easier to visualize, manipulate, and model

What else did we learn in this class? (cont.)

- The importance of validating your models and the k-fold cross-validation techniques
  - Divide your dataset into a train and a test sets. Train with training Data and Test it with test data
  - Divides your train set into chunks. Then train your model on all groups but one, and then test it on the one chunk left out. Repeat on all groups. This way, you are not wasting any data. Especially useful when you have a small dataset

What else did we learn in this class? (cont.)

- Git/GitHub
  - GitHub has become such a staple amongst the open-source development community that many developers have begun considering it a replacement for a conventional resume and some employers require applications to provide a link to and have an active contributing GitHub account in order to qualify for a job

What’s next?

What's next?

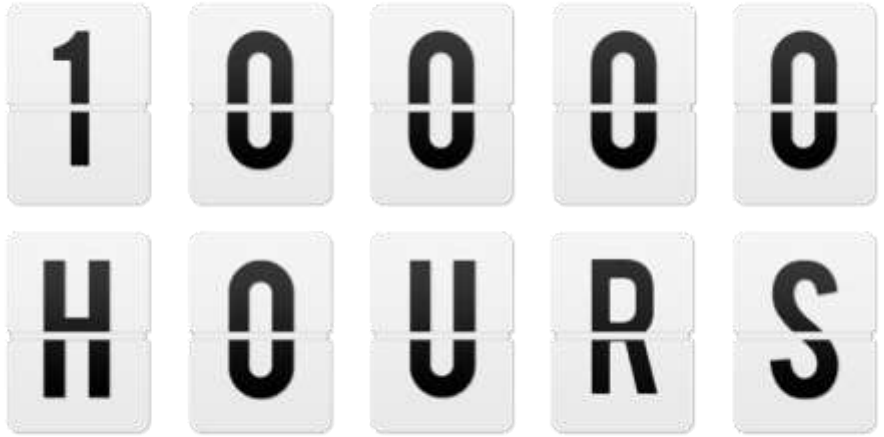
- A lot!
- This course was just an introduction to data science
- We focused on learning just a handful on models but learning them well. There are of course many more…

In short term, consider spending time learning or doing a deep dive on the following machine algorithms:

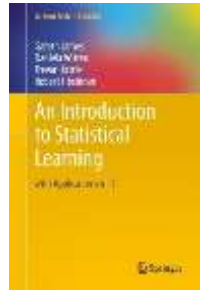
- Regularization (for linear regression and logistic regression)
- Boosting (e.g., on Regression/Classification Decision Trees)
- Naive Bayes (classification)
- Support Vector Machines (a.k.a., SVM) (classification)
- Ensemble Learning

Get your Hands Dirty, a.k.a., Practice, Practice, Practice…

- Kaggle (http://www.kaggle.com) competitions are a great way to practice everything we've learned in this class. And it's fun too!
  - Azi, Jeremiah, and Ivan are spending way too much time in this site…
  - You can compete by yourself but your can also team up with your fellow GA classmates!
- If Kaggle is not your thing, you should consider joining a study group if you haven't done so already



Longer term, consider the following resources



- An Introduction to Statistical Learning: with Applications in R (by James et al.). The e-book is available free-of-charge here

- A MOOC (Massive Open Online Courses) called Statistical Learning covering the book above is usually offered by Stanford also free-of-charge once a year during the winter. (now self-paced!) Check it out here

- For a more advanced treatment of these topics, check out The Elements of Statistical Learning: Data Mining, Inference, and Prediction (by Hastie et al.). And yes, the e-book is also free… (here)

Next Class

Final Project Presentations

Exit Ticket

Don’t forget to fill out your exit ticket here