Data visualisation course

The best way to kick off an understanding of data visualisation is to watch talks by the experts in the field. Here is a small compilation of interesting talks on data visualisation:

- Hans Rosling: No more boring data
- Aaron Koblin: Artfully visualizing our humanity
- David McCandless: The beauty of data visualization
- Thomas Goetz: It's time to redesign medical data

Next, browse some of the most beautiful data visualisations created:

- Visualisations by the New York Times
- What are some of the most innovative infographics on the Internet?
- Visual Complexity: A collection of beautiful data visualisations

Here are a few specific examples you may want to look at:

- In Investing, It's When You Start and When You Finish
- Obama's 2012 Budget Proposal: How 3.7 Trillion is Spent
- Dissecting the Midterm Exit Polls: how various groups voted for House candidates

There is a lot of buzz around data visualisation and data science today. Here are some quotes:

- ...the sexy job in the next 10 years will be statisticians.: Hal Varian, the chief economist at Google.
- Data is the Next Intel Inside: Tim O'Reilly, founder of O'Reilly media
- Data Scientist: The Hottest Job You Haven't Heard Of
- Why Your IT Department Needs Data Scientists
- The 6 Hottest New Jobs in IT
- The three sexy skills of data geeks (statistics, programming and visualisation)

Here's evidence in the growing focus on analytics:

- Searches for the word "analytics" on Google

Hopefully, these will inspire you to become a data scientist!

Also watch:

- Welcome video for this course
- Introductory slidecast
- Pictures through Numbers

Introduction

Let's start with an understanding of what data science is.

- What is Data Science (PDF, O'Reilly). This is the article that popularised the term Data Science, and in many ways, is considered the original definitive source.
- What is Data Science (Google Docs, Harlan Harris). This is another extensive presentation on what constitutes data science.

This course is an introduction to data science. You will learn three things:

- How to programmatically process data
- How to analyse it
- How to visualise it

This course will ask you to perform a series of tasks, supported by reading material. The tasks are designed for two real life roles:

- **An analyst at a media firm** preparing a report movies' ratings. The report needs to identify the ratings of movies by genre.
- **An analyst at an Indian mutual fund** automating the equity research platform. The analyst must use historical data to find the best performing stocks, and the best predictors of a stock's price.

If you have any queries regarding the course, you can post them to the Google Groups datavis2011. This will be the primary means of discussion and communication regarding this course.

Prerequisites

We assume you're familiar with:

- **Python**. If not, read 3 chapters of Dive into Python and all of Google's Python Class
- **HTML**. If not, read a quick introduction to HTML

On Windows, you may find the following software helpful. (On UNIX, you may already have Python, and you won't need Cygwin.)

- ActivePython
- Cygwin
- lxml
- NumPy
- SciPy

Interaction and Submissions

A Google Groups has been created for this course at groups.google.com/group/datavis2011/. Please join this group. You may ask any questions, share feedback, or discuss ideas related to data visualisation with the faculty and students on this forum.

This Google Group will be the primary means of interaction. Any emails sent to datavis@googlegroups.com *from your registered ID* will be posted on this forum.

Your completed tasks may be posted to any public repository, such as

- GitHub
- BitBucket
- Google Code
- etc.

Please let your instructors know the URL of your repository.

A. Learning Python

This is an optional module to help you brush up on Python. In this module, you will learn how to handle data structures in Python.

Tasks

1. Print all multiples of 3 under 20.
2. A palindrome is a string that reads the same forwards and backwards (level, for example.) Write a function that checks if a string is palindromic. Use that to print all palindomic numbers under 1000.
3. Write a python program that prints number of characters, words and lines in a file. For example, when run on diamond.txt:
4. Diamond has remarkable

5. optical characteristics.
6. Because of its extremely
7. rigid lattice, it can be
8. contaminated by very few
9. types of impurities,
10. such as boron and
11. nitrogen. Combined with
12. wide transparency, this
13. results in the clear,
14. colorless appearance of
15. most natural diamonds.

... it should print the number of lines, words and characters in the file.

1. Given a files salaries.csv with this structure:
2. City,Job,Salary
3. Delhi,Doctors,500
4. Delhi,Lawyers,400
5. Delhi,Plumbers,100
6. London,Doctors,800
7. London,Lawyers,700
8. London,Plumbers,300
9. Tokyo,Doctors,900
10. Tokyo,Lawyers,800
11. Tokyo,Plumbers,400
12. ...

... sort cities by descending order of lawyer salary.

1. Given the same salaries.csv file above, print the median salary of each profession.

Reading

- Dive into Python
- Google's Python Class
- For more interesting problems to solve, try Project Euler

B. Basic Statistics

TODO. Include predictive analytics.

1. Handling big data

The first step in processing data is getting that data. Your best case scenario is where you are given that data. But quite often, the data you need is not given -- or even if it is, it's not in the format you want.

In this module, you will learn:

- How to scrape data from external sources
- How to parse and transform it into a format you need

Tutorial

For a step-by-step walkthrough of how to scrape data from a site, visit Scraping IMD

Tasks

1. Download the IMDb Movie Ratings and:
    1. find the 20 most popular movies with a rank more than 8.0
    2. find the 20 best rated movies with over 40,000 votes in the 2000s (year >= 2000)
2. For each company in the list of equities on the BSE,
    1. get the last 1 month's closing price for each of these. Here is a sample CSV output
    2. find a way of parellalizing the downloads
    3. identify the top 10 gainers (%) across groups over the downloaded period.
    Note: The gain is the closing price on the last day of the period, divided by the closing price on the first day of the period, minus 1.

Reading

- Tutorial on scraping

- Parsing HTML using lxml
- Why use CSV
- Data Analysis on the Command Line

Optional reading

- Scraping using ScraperWiki
- Unix text processing
- Advanced BASH scripting
- Parallel tasks
- Threaded downloads

2. Analysis

Once you have the data, you need to analyse it. A background in statistics is helpful, but we won't be needing that.

A lot of insight is found in finding averages and trends by segment. Answering "Which group is the best?" is half the work of most analysts.

In this module, you will learn:

- Summarisation by segment
- Prediction using Linear Regression

Tasks

1. Using the IMDb Movie Rating Data:
2. convert it into a CSV file
3. find the average rank of the 10 most popular movies between 2000-2009 (inclusive)
4. find the year in the 1900s when the average rank increased the most, compared to the previous year.
   (Ignore movies with votes < 1000)
5. find the expected average rank for 2013 using linear regression. How good is this regression?
   (Ignore movies with votes < 1000.)
6. find the correlation between rank and votes for each year in the 1900s.
   By how much did the correlation coefficient grow each year? How good is this regression?

7. Using the downloaded stock data:
8. identify the stock most correlated with ICICI Bank's stockprice

Reading material

For these tasks, you can write either your own code, or use SciPy with its built-in statistical libraries.

- Processing CSV files with Python
- Collections: useful data structures
- Numpy: basic statistical processing
- Intro to Linear Regression (video)
- How good is the regression? (video)
- SciPy linear regression library

3. Vector graphics

There are many ways of presenting visuals. One common way is to use image files. But this suffers from two problems:

- It is static. It can neither move, nor allow the user to interact
- It isn't scalable. When you zoom in, you lose resolution.

Vector graphics are way of overcoming both limitations. Watch this video on why vector formats.

We will be using SVG as our vector format, due to it's widespread support and ease of manipulation.

In this module, you will learn:

- How to draw in SVG
- How to style SVG and create graphs
- Tools to manipulate SVG

Tasks

As practice, read about paths / rectangles, and draw a sparkline / bar graph showing the data 23,80,92,62,98,7,9,56,19,68.

1. Create an SVG file that reproduces this bubble chart in SVG
2. Draw a scatterplot of rank vs votes for every movie with at least 10,000 votes. (x-axis=votes, y-axis=rank)

3. Draw a correlation matrix of any 30 stocks on the Sensex. This is a matrix with both axes holding the stock names. The colour of each cell is the correlation between that pair of stocks: red for -1 and green for -1.

Reading material

- SVG Essentials:
- SVG reference
- Vector drawing tools

Interlude: Design

A good aesthetic sense is key to creating visualisations. Colour, typography and layout can be used to great effect in increasing the impact of your design.

Here are some references that will be of great help in improving your sense of design.

General Design

- Philippe Stark on Design (video)
- Don Norman on Design (video)
- The Visual Display of Quantitative Information (book)
- Better designs (Google Docs presentation)

Visualisations

- Protovis examples: a wide variety of chart sets
- Infosthetics: a blog on data visualisation
- Flowing Data: a blog on data visualisation
- Stamen projects: Projects by Stamen Design, a leading visualisation firm
- Gallery of data visualisation

4. Templates

The key concept in data visualisation is *using code to generate visuals from data*. Why use code? Because it's repeatable

We use a *template* that will convert *data* into a HTML file -- typically embedding SVG within that.

There are many templating engines available. We'll be using one of the simplest: Tornado's templates. These offer considerable flexibility. You write what is nearly full Python code in the template.

Tasks

Using Tornado templates,

1. Draw bar graph of the number of movies by year since 1900
2. For each stock, draw a sparkline showing last month's closing price trend
3. Convert these files into PDF without using a browser

Reading material

- Template creation using Tornado
- PDF conversion tools: wkhtml2pdf or PhantomJS
- Essential visualisation resources: a collection of articles. 1 2 3 4 5 6 7 8

5. Gramener visualisation server

Gramener's visualisation server simplifies the creation of data visualisations. It comes with a library of pre-packaged visualisations that can be put together into complex dashboards.

A copy of the server will be provided to you at the start of this module. But in the meantime, you can browse the package documentation at learn.gramener.com/docs/.

Tasks

1. A full-page dashboard for each year of movies, showing the best rated, most popular movies and any other interesting statistics
2. A full-page stock summary for any stock, showing the stock's absolute and relative performance
3. Create a scatterplot of ratings vs votes for every movie with over 10,000 votes. On hover, show the movie name example

Reading material

- Gramener visualisation server documentation

6. Interactive visualisations

The output of these visualisations need not stay static. Using Javascript, the content can be made dynamic (changing over time) as well as interactive (changing based on user inputs).

In this module, you will learn how to structure templates that produce interactive visualisations, and design them to bring out the full meaning of the data -- without overwhelming the user.

TODO

7. Animations

TODO

8. Project

To be done in groups of two

- Public data visualisations
- New innovative chart modules

Additional reading material

Books

(Ordered from easy to tough)

1. Programming Collective Intelligence is a very readable introduction to machine learning and recommendation systems. It explains the algorithms using simply (but accurately) with Python code.
2. The Visual Display of Quantitative Information (book)
3. The Elements of Statistical Learning
4. R in a Nutshell is a gentle introduction to R.

Courses

- Ben Shneiderman, Information Visualization
- Stanford's CS229: Machine Learning uses Matlab and Octave
- MIT's 6.867: Machine Learning