# Data Visualization Methods

## course schedule

## final project

---

**resources**

**Statistical summaries and visualizations of parts of a dataset (groups, factors, categories)**

Collections of best data visualizations projects:

**http://infosthetics.com/**

**http://www.visualcomplexity.com/vc/**

Best visualizations lists linked here:
**http://manovich.net/index.php/exhibitions/selfiecity**

---

**Semester:**
Spring 2016

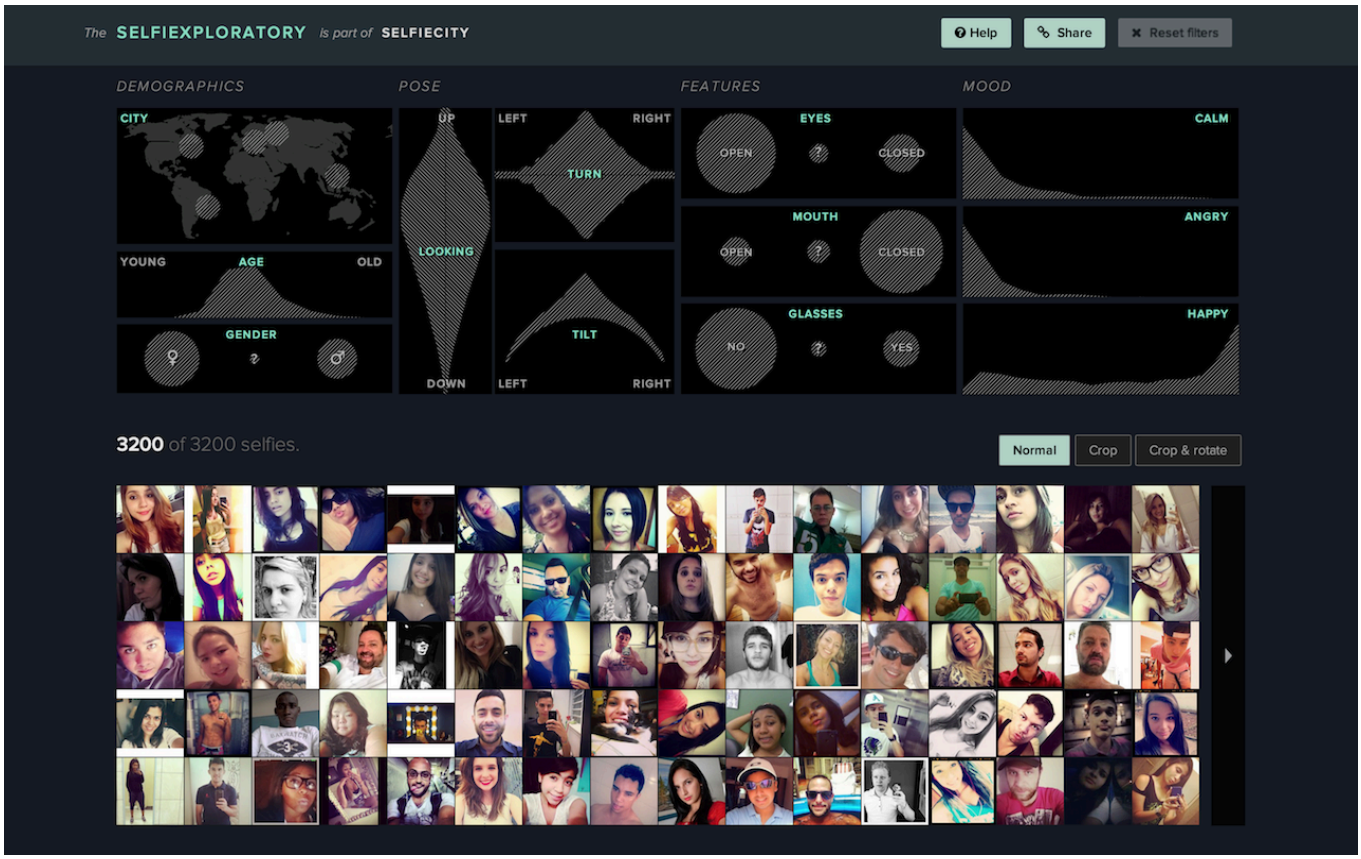
**School:**
The Graduate Center, CUNY (City University of New York)
365 5th Avenue, New York City.

**Instructor:**
Lev Manovich
Professor, PhD Program in Computer Science, The Graduate Center, CUNY.
Director, Software Studies Initiative.

**Format:** MA / PhD seminar.

**Meeting day and time:**
Mondays, 11:45 am - 1:45 pm

**Originally scheduled room: 4433**




Screenshot from http://selfiecity.net/ (2014). The project won Gold Award in Best Visualization Project of the Year category in Information is Beautiful competition in 2014.

---

## Course description

> "The next big idea in language, history and the arts? Data."
> New York Times, November 16, 2010.
>
> "Statistics may be regarded as (i.) the study of populations, (ii.) as the study of variation, (iii.) as the study of methods of the reduction of data." Ronald A. Fisher. STATISTICAL METHODS FOR RESEARCH WORKERS. (1925) [link]

## Introduction

This course covers (1) how to visualize data; (2) preparing and analysing datasets; (3) elements of graphic design; (4) working with large social media datasets

1| We will study and practice common **visualization techniques** for a single and multiple variables, quantitative and categorical data, spatial and temporal data, networks, and image collections.

2| Both in academic research and in industry, visualization and working with data go together. Some visualization techniques can't be used before the data is transformed using methods from modern statistics or data science. Additionally, real-life datasets often need to be cleaned and organized in proper formats before they can be visualized. Therefore, we will also devote significant time to learn basics of **data cleaning** and **data analysis**.

3| Students will also be introduced to basic **principles of modern design** as they apply to design of static, animated and interactive visualizations, data-centric publications, maps, and other common types of data design. The principles cover use of form, proportion, color, composition, design grids, basics of typography, hierarchical organization of information, systematic use of design variables, rhythm.

Students will complete a number of practical assignments to understand and start mastering principles and techniques being introduced in class. The class time will be divided in three parts – 1/3 for instructor presentations, 1/3 for discussions of outstanding visualizations, design projects and readings, and 1/3 for critique of student work. Selected historical and theoretical readings will be used to introduce students to

the histories of visualization and modern design and to help them start thinking critically about the common practices of these fields, and their use in commercial, non-profit, and scientific settings. In this way, the class aims to both teach students solid practical skills and critical reflexive attitude towards the material.

Since this class is offered within MA program and is related to Digital Humanities courses at The Graduate Center, we will also devote some times to working with **large cultural datasets** and **social media data**. The availability of massive social and cultural data sets (including social media and digitized cultural artifacts) and progress in computing opened up new possibilities for the study of societies and cultures. The fields of social computing, cultunomics, digital humanities and cultural analytics that developed in the end of 2000s started to explore some of these possibilities.

Today the analysis of large datasets from social networks and media sharing sites is carried out by hundreds of thousands of thousands of computer scientists working in the industry and in the academy. Google, Facebook, Twitter, Instagram, Amazon, Twitter, Echonest are just a few of many thousands of companies that rely on the analysis of social media, user-generated content, connections between users, and also content of web sites, large bodies of texts, music tracks, images, and other types of media and online activities.

The "big data turn" also already affected many fields in humanities (digital humanities, history, literary studies, art history, film studies, archeology, etc.), social sciences (e.g., computational sociology), and professional fields such as journalism, arts administration, city governance, and urban planning

But even if you don't plan now to analyze social or cultural data in the future, this class can be still very relevant. The abilities to analyze, visualize and reason about large data - and to understand and use others research that does this - are quickly becoming essential for 21st century researchers regardless of their fields.

For examples of recent projects from Software Studies Initiative (Software Studies Initiative), that analyze online content, visit this page.

## Rationale

Data visualization is increasingly important today in more and more fields. Its growing popularity corresponds to important cultural and technological shifts in our societies –

adoption of data-centric analysis research and arguments across dozens of new areas, and also arrival of massive data sets. Data visualization techniques allow people to use perception and cognition to see patterns in data, and communicate and form research hypotheses. The goal of this course is to introduce students to fundamentals of data visualization and relevant design principles. Students will learn the basic data visualization techniques, when and how to use them, how to design visualizations that best exploit human visual perception, and how to visualize various types of data (quantitative, categorical, spatial, temporal, networks).

**Learning Goals/Outcomes**

The key goals of this course are to learn how to use modern visualization techniques to help analysis and understanding of data, how to prepare and analyze data sets using selected statistical and data science methods, how to use principles of design in creating effective and engaging visualizations, and how to approach visualization of various data types.

- Students will be able to understand data visualization medium theoretically and historically in relation to other major visual and communication media, past and present.

- They will learn how to use the basic modern visualization techniques.

- They will learn basic techniques for cleaning data

- They will learn summary statistics and selected techniques from data science commonly used before data can be visualized.

- They will learn how to create interactive web visualizations.

- They will also learn basic principles of modern graphic design relevant for design of visualizations, and interactive data projects.

- They will learn to reason about visualizations and data patterns, and critique visualizations.

- They will learn how to approach design process, present initial multiple proposals and refine the chosen design though iterations.

**Assessments**

a. Class participation: students are expected to participate in discussions of the assigned material.

b. Practical assignments: students will complete a number of homeworks.

c. Final project: creation of a larger scale visualization project and a short paper that discusses the datasets, the visualization techniques and the findings.

**This short online class is similar to the approach we will use in learning explorative data analysis:**

https://www.class-central.com/mooc/1478/udacity-data-analysis-with-r

## Course requirements and grading:

1) Active participation in class discussions (which requires doing all assigned readings on time): %20

2) Practical homeworks: %60

3) Final project: %20

## Readings and lecture notes:

Homework to be done before each class is **listed below**

Because our class meets only once a week, I will not be able to go over every topic listed for every class. For the topics which we will not cover class, I have linked lecture notes and other linked material. Therefore, in addition to the assigned readings and sites to view in homeworks, you should also go through **lecture notes and linked material for each class**. You should do that **after** each class. Feel free to research any subjects which interest you in more detail.

If you are already familiar with any of the readings, projects, concepts, or data analysis/visualization techniques covered in any of the homework - skip them.

If you don't have computer science background to understand details some of the readings, try at least to get the key ideas - by reading an introduction and a conclusion of an assigned or recommended computer science article.

### Recommended Textbooks:

Some of the chapters of the textbooks listed below will be assigned during the semester (these chapters will be made available as PDFs to students in this class):

1) If you have computer science or equivalent background - data mining in R:
Vanchang Zhao. R and Data Mining: Examples and Case Studies. Elsevier, 2012.

2) If you have very little technical background - very gentle introduction to working with data and analyzing it in R:
Jeffrey Stanton, Introduction to Data Science, 2013.

3) This textbook specifically focuses on analyzing major "humanities"data types using R:
Taylor Arnold and Lauren. Humanities Data in R: Exploring Networks, Geospatial Data, Images, and Text. Springler, 2015.

More choices - some of the free data science books online:
http://codecondo.com/9-free-books-for-learning-data-mining-data-analysis/

---

# COURSE SCHEDULE:

[likely to change during the semester]

**2 Class introduction; examples of data visualization**

homework for class 3

**3 Introduction to R; working with data tables ( R )**

R class demo (class 3)

homework for class 4

**4 Classic visualization techniques for 1 and 2 variables ( R )**

Class 4 notes and R demo

homework for class 5

**5 Basic descriptive statistical measures; creating and visualizing data summaries and group summaries (in R). History of statistics.**

Class 5 notes

Calculating and Visualizing Descriptive Statistics in R

Class 6 homework

**6 Guest lecture and demo -** DAMIANO CERRONE from Spin Unit:
**analysis and visualization of spatial data.**

Class 7 homework

**7 Guest lecture -** Alise Tifetale from GC CUNY and software studies initiative.

**8 Correlation and linear regression Visualization of correlations between many variables. Changing continuous variables into categorical variables** (cut). **Plotting multiple time series** (long vs. wide data). **( R ).**

Class 8 notes

**9 Cleaning data ( Excel / Google Sheets). Basic principles of modern graphic design for visualization. Changing formats of classical visualization techniques to achieve fresh look.**

**10 Creating interactive web visualizations (Google Charts; Plot.ly with R).**

**11 Artistic visualization**

**12 Elements of contemporary data analysis: features, multi-dimensional feature space; distance matrix, heatmaps.**

**13 Dimension reduction (PCA, MDS); use of dimension reduction methods for visualization of multi - variable datasets.**

**14 Analyzing and visualizing data in time; visualizing image collections**

---

# RESOURCES:

# My highly selective list of BEST tools and resources for learning visualization, data analysis, and working with social media data:

# Start here if you completely new to this - the very basics:

http://schoolofdata.org/handbook/courses/tell-me-a-story/

http://schoolofdata.org/handbook/courses/data-to-diagrams/

# Get some (big) data to analyze and visualize:

http://www.smartdatacollective.com/bernardmarr/235366/big-data-20-free-big-data-sources-everyone-should-know

http://www.kdnuggets.com/2011/02/free-public-datasets.html

Hundreds of datasets for every country from quandl.com - for example: https://www.quandl.com/collections/society/internet-users-by-country

# Working with data - best books and online texts:

[Warning: this list is biased towards R, because I love it.]

Best general overview of working with data for non-technical audiences (its written for journalists but many parts are quite general):

**data journalism handbook**

Best textbook that teaches you data analysis (using R) - for people with very little technical background:

Jeffrey Stanton, **Introduction to Data Science.**

For more advanced students - computer science text teaching you data analysis using R:

Vanchang Zhao, **R and Data Mining: Examples and Case Studies**. Elsevier, 2012.

Analysis of (literary) texts in R - written for digital humanities audience, very gentle and gradual:

http://www.springer.com/statistics/computational+statistics/book/978-3-319-03163-7

Text mining in R - fast, diving right in:

Hands-On Data Science with R Text Mining

More free data science books:

http://codecondo.com/9-free-books-for-learning-data-mining-data-analysis/

# Data cleaning:

http://schoolofdata.org/courses/#IntroDataCleaning

# Online courses in Data Science:

Some of the online courses about Data Science

Another list of online courses about Data Science

**This online class is similar to the approach of taken in this class (i.e. this syllabus):**

https://www.class-central.com/mooc/1478/udacity-data-analysis-with-r

## Digital humanities - online resources:

lists of tools and other resources:

The CUNY Digital Humanities Resource Guide

# Best visualization tools:

Yes, there dozens of software tools, but these are used most widely today:

**http://flowingdata.com/2016/03/08/what-i-use-to-visualize-data/**

**Leader in  data visualization software, no programming (requires PC or Windows emulator on Mac)**

**http://www.tableau.com/**

Interactive web visualization, without or with programming:

**Google charts**

**More sophisticated interactive web visualization using programming:**

**d3: http://d3js.org/**

**Creating online interactive maps, no programming**

**CartoDB** or Mapbox

**Basic interactive web visualization (interfaces with R), can be used without or with programming**

**plot.ly**

# If you want to learn d3:

**Free book:**
http://chimera.labs.oreilly.com/books/1230000000345/index.html

# Learning and using R:

**free introductory R class from code school:**
https://www.codeschool.com/courses/try-r

**many R courses from datacomp ($25/month):**
https://www.datacamp.com/

**variety of ways to explore and visualize data:**
**Chapter 3** - Vanchang Zhao, R and Data Mining: Examples and Case Studies (Elsevier, 2012).

**Best tutorials for basic and advanced visualization using R:**
http://flowingdata.com/category/tutorials/

**Creating interactive web visualizations using R:**
https://plot.ly/

# If you want to use Python:

http://pbpython.com/visualization-tools-1.html

http://radar.oreilly.com/2013/03/python-data-tools-just-keep-getting-better.html

---

# homework for class 3

**1) Do this online mini-course:**

http://schoolofdata.org/handbook/courses/data-to-diagrams/

**2) Read the following section from Data Journalism Handbook:**

Using visualizations to Tell Stories

Designing with Data

Different Charts Tell Different Stories

**3) Check these organizations/programs which research issues around data & society + prepare future researchers:**

http://www.datakind.org/

http://schoolofdata.org/

http://www.datasociety.net/ (NYC) - great events calendar

Civic Hall (NYC)

http://www.law.nyu.edu/centers/ili (NYC)

Tow Center for Digital Journalism (NYC)

---

# homework for class 4

Start learning R:

**For students ready to jump into R:**

A) **go through and try all commands in the following parts** of the official *An Introduction to R*:

1 Introduction and preliminaries
2 Simple manipulations; numbers and vectors
3 Objects, their modes and attributes
7 Reading data from files

B) After that, **became familiar with the commands** described in the following parts of *Quick R* website (my favorite R resource):

http://www.statmethods.net/interface/index.html
http://www.statmethods.net/input/contents.html
http://www.statmethods.net/management/index.html

**Alternative - for students who prefer a more slow gradual introduction to R and basics of working with data:**

Jeffrey Stanton, An Introduction to Data Science. 2012 (full book available online): Go through **Chapter 1 - Chapter 8**.

**More R learning resources:**

Introduction to R class (free):

https://www.datacamp.com/courses/free-introduction-to-r

http://mran.revolutionanalytics.com/documents/getting-started/

http://www.introductoryr.co.uk/R_Resources_for_Beginners.html

---

# R class demo - class meeting 3

## Datasets used in this demo:

1 data assembled for On Broadway project in our lab - Broadway street in manhattan (13 miles) broken into 713 rectangles, most data from 2/2014-7/2014

- nyc

2 data assembled for Selfiecity project in our lab - as sample of 120,000 images shared in six global cities during one week, 12/2013

- xx

## Using R scripts

**script example demonstrating some commands in the demo below -**
link

## Reading data into R:

http://www.statmethods.net/input/index.html

Example - reading a data file in tab delimited format:

select folder containing the data:
Misc > Change working directory

list files in the directory:
list.files()
dir()

read the tab delimited data files into R:
xx <- read.delim("**120K_Instagram_images_data.txt**")
nyc = read.delim("broadway-crosstab.LM.all.txt")

read comma delimited data file into R:
tags = read.delim("top_10_tags_neighborhood.csv")

### Guide to reading data into R:

http://www.computerworld.com/article/2497164/business-intelligence-beginner-s-guide-to-r-get-your-data-into-r.html

### Save R data into a tab delimited file:

write.table(xx.tokyo, file="tokyo.txt", row.names = FALSE, quote = FALSE, sep="\t")

### Load .Rda file into R:

In addition to using data in standard .txt and .csv files, R can also store data in its own native format - **.rda**

loading data in .rda format into R workspace:

load("/Users/levmanovich/Documents/**instagram_park_ave_2_stream_May_01_Jun_01.Rda**")

You can also load .rda file using top menu:
File > Open Document

After you read .Rda file, you should find how does the corresponding object is named in R:

ls()

**Write data to .Rda file:**

For example, we want to sample the .Rda file and then write a new file to the hard drive:

nyc.sample <- nyc[sample(1:nrow(nyc), 40000, replace=FALSE),]

save(nyc.sample,file="nyc.May.sample.Rda")

## Working with data:

examine data objects in R workspace:

ls()

delete an object “tags” from R workspace:

rm("tags")

**Examining a data object:**

dim(xx)
str(xx)
colnames(xx)

**Using head and tail commands:**

```
head(xx)
tail(xx)
head(xx, n=20)
head(xx$username, n=40)
```

**Drop data columns you will not use:**

```
> colnames(xx)
 [1] "just_filename"             "instagram_id"
 [3] "updated"                   "updated_trans_from_UT"
 [5] "updated_trans_from_UT_value" "datetime_number"
 [7] "hour"                      "date"
 [9] "username"                  "city"


xx2 = xx[c(1:2,5:8,9)]

colnames(xx2)
[1] "just_filename"             "instagram_id"
[3] "updated_trans_from_UT_value" "datetime_number"
[5] "hour"                      "date"
[7] "username"
```

**create random sample:**

http://www.statmethods.net/management/subset.html

```
xx.sample <- xx[sample(1:nrow(xx), 5000, replace=FALSE),]

dim(xx)
dim(xx.sample)
```

**count the number of records for each factor (=categorical variable):**

```
table(xx$city)
table(xx.sample$city)
table(xx$hour)
```

**plot this:**
barplot(table(xx$hour))

**count the number of unique values in a data object** (120K dataset):

length(unique(xxx$username))

## sort:

http://www.statmethods.net/management/sorting.html

head(nyc)
head(nyc[order(nyc$numPix..Instagram),])
newdata = nyc[order(nyc$numPix..Instagram),]

tail(sort(table(xx$username)))
tail(sort(table(xx$username)), n=50)
barplot(tail(sort(table(xx$username)), n=50))

## subset:

http://www.statmethods.net/management/subset.html

**selecting part of the data based on conditions:**

table(nyc$Neighborhood)
nyc.Mid = nyc[which(nyc$Neighborhood=="Midtown (34th-42nd)"),]

## Summarize data by factor:

There are many ways to do this in R - here are just some:

http://stackoverflow.com/questions/9847054/how-to-get-summary-statistics-by-group

http://stats.stackexchange.com/questions/8225/how-to-summarize-data-by-group-in-r

http://www.statmethods.net/management/aggregate.html

**Count number of unique factor within another factor - one method:**

for example, we want to find how many images each user shared:
number of (xx$instagram_id) per each (xx$username):

```
mmm <- aggregate(instagram_id ~ username, data=xx, FUN=function(x) length(unique(x)))
```

using this result - for example:
barplot(sort(mmm$instagram_id))

sort by numbers of images per user:
mmm.sort <- mmm[order(mmm$instagram_id),]

## Add a new column with values based on values in other column(s):

```
kki2$maidan_only <- with(kki2, ifelse(kki$maidan>0 & kki$other=="0", 1, 0))
```

## Representing dates and times in R:

http://www.noamross.net/blog/2014/2/10/using-times-and-dates-in-r---presentation-code.html

**Convert Excel date and time format to number in R:**

```
xx$date_time_num <- as.numeric(as.POSIXlt(xx$updated, origin="1899-12-30"))
```

example:
"2014-02-19 02:01:10" becomes  1392794517

# Class 4 notes and R demo

# basic visualization concepts and methods:

**data set** in Google docs used for demonstration in this section

If a data table uses rows to represent separate objects and columns to represent their properties, "variables" correspond to columns.

The columns can be also called characteristics or "features" (in data science).

Basic visualization techniques can be divided in this ways:

**visualizations of one variable: bar plot, line plot, histogram, pie plot**

**visualizations of two variables: scatter plot, heatmap**

**visualizations of multiple variables: scatter plot matrix, mosaic plot, parallel plot**, etc.

**Note:** You can think of a basic **map** as a scatterplot of two data columns: latitude and longitude.

**Note:** a special case for a scatter plot is a situation when **one variable which is recorded at regular intervals**. For example, we can record temperature at 1 hour intervals. Or we can record child's height at 1 year intervals. In such a case, we can plot this one column using bar plot or line plot, without using the second column that indicates time intervals.

This data is also called **time series**.

**Plotting full data vs. aggregated data:**

Visualizations can be made using full data in column(s), or **aggregated** data. An alternative term for aggregated data is **summarized**. Typically we summarize data using some categories that already exist in the data, or we create new ones (for example, histogram).

Typical **ways of summarizing data:** count (numbers of cases in each category) or use basic descriptive statistics (mean, median, standard deviation, variance, etc.)

### Continuous vs. Discrete Data:

Discrete data - integers, categories

Continuous data - numbers with proportions (floating point numbers)

Continuous data is plotted using histograms or scatter plots.

Continuous data can be made discrete by aggregating it (will be covered later).

### Ordering one dimensional data before plotting:

For one variable, sort the data in ascending or descending order before plotting as bar plot / line plot - unless the data already has a particular logical order, which should be preserved.

### Visualization proportions:

If you show data which is changing over time, make the time/sequence axis **longer** than the other axis.

If you plot two variables and none of them is time, make your scatterplot **square**.

# basic visualization in R:

http://www.statmethods.net/graphs/index.html

Basic visualization techniques in R for one variable: **bar plot, histogram, pie chart,**

**dot plot.**

Basic visualization technique in R for two variables: **scatter plot**

**Visualization commands in ggplot2 package:**

http://docs.ggplot2.org/current/

## Optional - study on your own if you want to look ahead: advanced visualization in R (we will cover this material in later classes):

**Plot using facets with ggplot2:**

http://docs.ggplot2.org/current/facet_grid.html

http://docs.ggplot2.org/current/facet_wrap.html

**Scatter plot methods for plotting very large data sets:**

http://stackoverflow.com/questions/7714677/r-scatterplot-with-too-many-points

**SmoothScatter (x, y) - smoothed scatter plot**

**Another method for 2D plots of large data (bucketing):**
https://medium.com/data-lab/74b9f41509b

**Heatmaps:**

[this example uses van Gogh data set]

```
xx <- read.delim("van_gogh_data.txt" )
vangogh_dist <- dist(scale(xx[,2:3]))   // using average brightness and saturation of images
```

heatmap(as.matrix(vangogh_dist), col=rainbow(16))

heatmap without automatic re-ordering:
heatmap(as.matrix(dist(scale(xx[,2:3])), Rowv=NA, Colv=NA))

plot using grayscale:
image(as.matrix(dist(scale(xx[,2:3]))), col=gray(0:16/16))

**Visualization of large datasets with tabplot:**

http://cran.r-project.org/web/packages/tabplot/vignettes/tabplot-vignette.html

**Plotting World and Country maps in R:**

https://www.students.ncl.ac.uk/keith.newman/r/maps-in-r

**Inspiration: variety of ways to organize data for maps:**

http://youarehere.cc/#/maps/by-city

**Scatterplot matrix:**

http://little-book-of-r-for-multivariate-analysis.readthedocs.org/en/latest/src/multivariateanalysis.html#plotting-multivariate-data

**Network visualization:**

http://web.stanford.edu/~messing/CreateBasicNetVis.html

**Social Network visualization with R and Dephi:**

http://kateto.net/2014/04/facebook-data-collection-and-photo-network-visualization-with-gephi-and-r/

# Data visualization - selected resources:

**Examples of well designed 1D and 2D data visualizations using R:**

**ggplot2**

**https://plot.ly/learn/**

**Some of the best recent visualizations:**

http://blog.udacity.com/2015/01/15-data-visualizations-will-blow-mind.html

http://flowingdata.com/2014/12/19/the-best-data-visualization-projects-of-2014-2/

see "best of" lists linked here:
**http://manovich.net/index.php/exhibitions/selfiecity**

**Classic collections of most interesting data visualizations:**

**http://infosthetics.com/**

**http://www.visualcomplexity.com/vc/**

**Graduate programs in data visualization in U.S.:**

PARSONS:
http://www.newschool.edu/parsons/ms-data-visualization/

Northeastern:
http://www.northeastern.edu/camd/artdesign/academic-programs/mfa-in-information-design-and-visualization/

DUKE - MA in Historical and Cultural Data Visualization:
http://www.dukewired.org/ma/
Graduate Center - currently developing a proposal for M.S. in data analysis and visualization.

**Very good tutorials which include presentations of basic visualizations techniques in R:**

http://flowingdata.com/category/tutorials/

**Most popular data visualization tools:**

Leading platform for interactive web visualization:
http://d3js.org/

Standard professional data visualization software:
Tableau: http://www.tableausoftware.com/public/how-it-work

Interactive web visualization using R:
http://www.r-bloggers.com/interactive-visualizations-with-r-a-minireview/
http://shiny.rstudio.com/gallery/

**More visualization tools and software:**

"The 36 best tools for data visualization" (May 2014):
http://www.creativebloq.com/design-tools/data-visualization-712402

# Class 4 demo -visualizing a sample dataset using R

The data set used for this demo is 13,208 images and video shared by 6,165 Instagram users in central part of Kiev (Ukraine) during 2014 Ukraine Revolution. The images were shared during 2/17-2/22, 2014. These images were tagged with 5,845 unique tags. The data and images were downloaded using Instagram API.

**Project that uses this data:**

**http://www.the-everyday.net/**

You can see some of the visualizations similar to the ones I will show in the demo in our published project:

http://www.the-everyday.net/p/the-extraordinary-and-everyday.html

**Data file:**

Kiev-feb17-feb22-1row-per-image_QTIP.txt

R script for this demo

**1. (optional) examine the sample data set in Excel or Google Charts.**

**2. Load the data file in R:**

```
list.files()
xx <- read.delim("Kiev-feb17-feb22-1row-per-image_QTIP.txt")
```

**3. Examine the structure of the dataset:**

```
dim(xx)
str(xx)
colnames(xx)
head(xx)
```

**4. Examine and visualize number of images by time:**

```
table(xx$date.y)
barplot(table(xx$date.y))
barplot(table(xx$date.y), las=2)

hist(xx$numeric.date.y)
hist(xx$numeric.date.y, breaks=50)
hist(xx$numeric.date.y, breaks=200)
hist(xx$numeric.date.y, breaks=200, border="grey", col="grey")
```

**5. Visualize location data:**

```
plot(xx$lat, xx$lon)
```

```
plot(xx$lat, xx$lon, pch=".")
```

**6. Optional: visualize location data using CartoDB**

http://cdb.io/1lOvIhI

**7. "Slice" data by one of the geographic coordinates and dates:**

After you explore the whole dataset, typically you want to compare its different parts. Note that there lots of ways to divide a dataset into groups and visualize them. Some are already implied by the structure of the data, but you can always invent others.

```
hist(xx$lat)
hist(xx$lat, breaks=100)

kiev.feb17 <- xx[which(xx$date.y == "2014-02-17"),]
kiev.feb18_22 <- xx[which(xx$date.y != "2014-02-17"),]

dim(kiev.feb17)
dim(kiev.feb18_22)

hist(kiev.feb17$lat, breaks=100)
hist(kiev.feb18_22$lat, breaks=100)
```

**8. Create and visualize summary statistics by groups**

Looking at the use of particular tags:

```
table(xx$maidan)
table(xx$Kiev)
barplot(table(xx$maidan))
barplot(table(xx$Kiev))

length(unique(xx$username))

sort(table(xx$username))
plot(tail(sort(table(xx$username)), n=100))
plot(tail(sort(table(xx$username)), n=1000), type="l")
plot(tail(sort(table(xx$username)), n=1000), type="h")
```

Count number of unique factor within another factor - for example, number of unique users per day:

```
mm <- aggregate(username ~ date.y , data=xx, FUN=function(x) length(unique(x)))
```

mm

**9. Create a new object containing a part of the dataset**

```
xx.maidan <- xx[which(xx$maidan >0),]
```

**10. Compare volumes of images with diff. characteristics over time**

```
// create a window to hold two plots
op <- par(mfrow = c(2, 1))

// create the two plots
hist(xx$numeric.date.y, breaks=100, border="grey", col="grey")
hist(xx.maidan$numeric.date.y, breaks=100, border="grey", col="grey")

// make the plots use the same values for Y
op <- par(mfrow = c(2, 1))
hist(xx$numeric.date.y, breaks=100, border="grey", col="grey", ylim=c(0,200))
hist(xx.maidan$numeric.date.y, breaks=50, border="grey", col="grey", ylim=c(0,200))
```

**11. Create multiple plots using ggplot2 package**

**documentation: ggplot2**

```
library(ggplot2)
ggplot(xx, aes(Value, Hue)) + geom_point(alpha = 1/20) + facet_grid(. ~ date.y)
ggplot(xx, aes(Value, Hue)) + geom_point(alpha = 1/20) + facet_wrap(~ date.y)
```

## Demo 2

using Van Gogh dataset: link

```
vg = read.delim("van_gogh_additional_measurements.txt")

ggplot(vg, aes(factor(Year))) + geom_bar()

ggplot(vg, aes(factor(Year))) + geom_bar() + coord_polar()

ggplot(vg, aes(factor(Month))) + geom_bar() + coord_polar()

ggplot(vg, aes(factor(Month))) + geom_bar() + facet_wrap(~Year)
```

```
ggplot(vg, aes(factor(Year_Month))) + geom_bar()

ggplot(vg, aes(factor(Year_Month))) + geom_bar() + coord_polar()

ggplot(vg, aes(x=Year_Month, y=Shape_Count)) + geom_point()
```

---

# Homework for class 5

## 1| practical assignment:

You will receive email from Dropbox telling you where to upload your file.

Goal: use R to visualize relations between selected variables in van Gogh data. Use built-in visualization commands and/or with ggplot2.

van Gogh data file for this homework (this file version has genres data).

The variables to visualize:

1) Year, Month
2) one of GENRE variables, image_proportions
3) Label_Place, season

Note: For (1) and (3) you can visualize differences in the numbers of paintings in relation to the year and month, and place and season variables. Or you can visualize patterns in any other third variable (so not only images count).

After you create 3 visualizations you are happy with, combine them into one PDF and submit it. (the PDF should be < 15 MB).

**Name the PDF file as follows:**
lastname_firstname_ assingnment_1.pdf

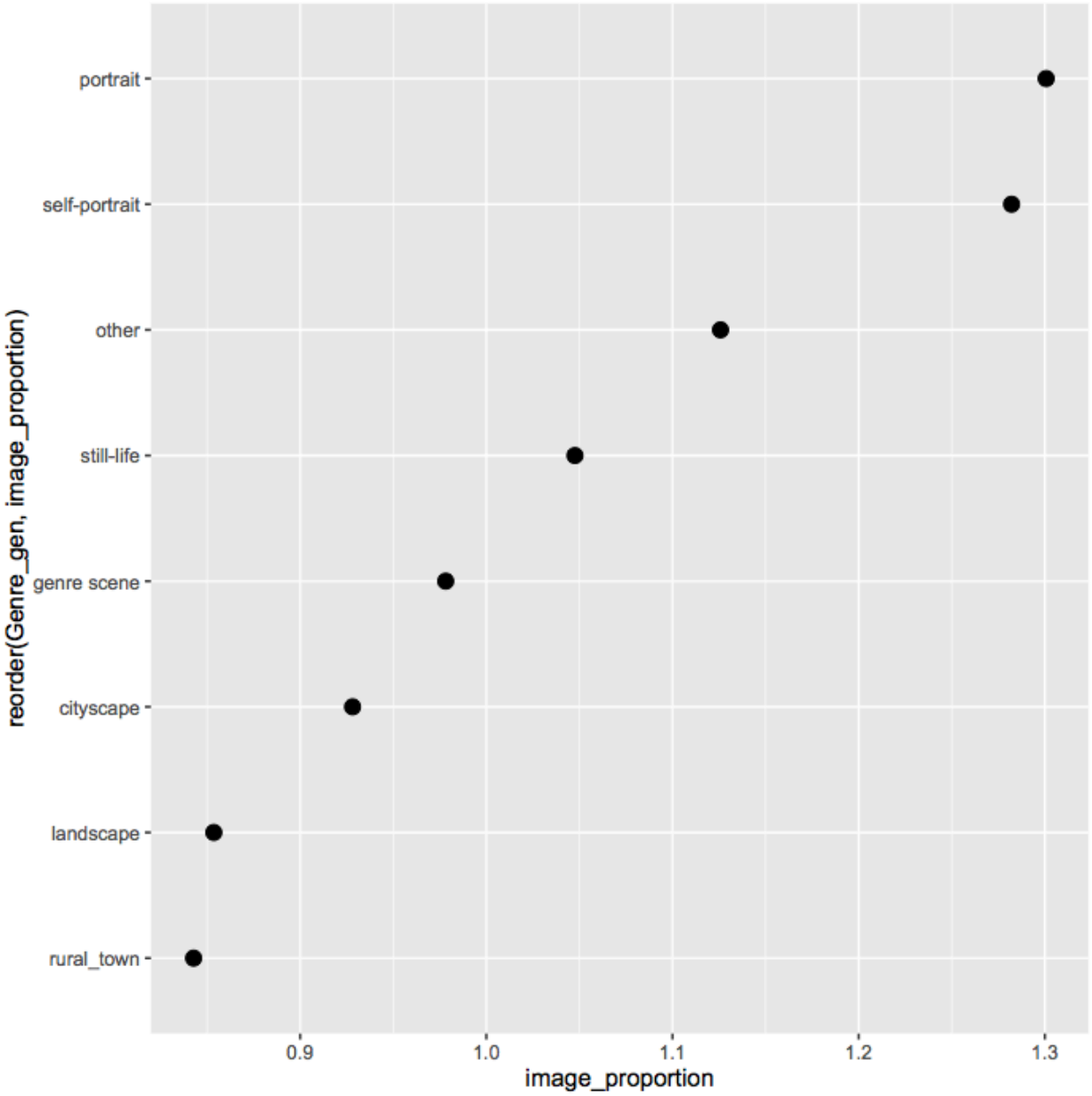
---

**The following are three examples of possible visualizations for this homework:**

ggplot(mm, aes(reorder(Genre_gen,image_proportion),image_proportion)) +
geom_point(size=3) + coord_flip()

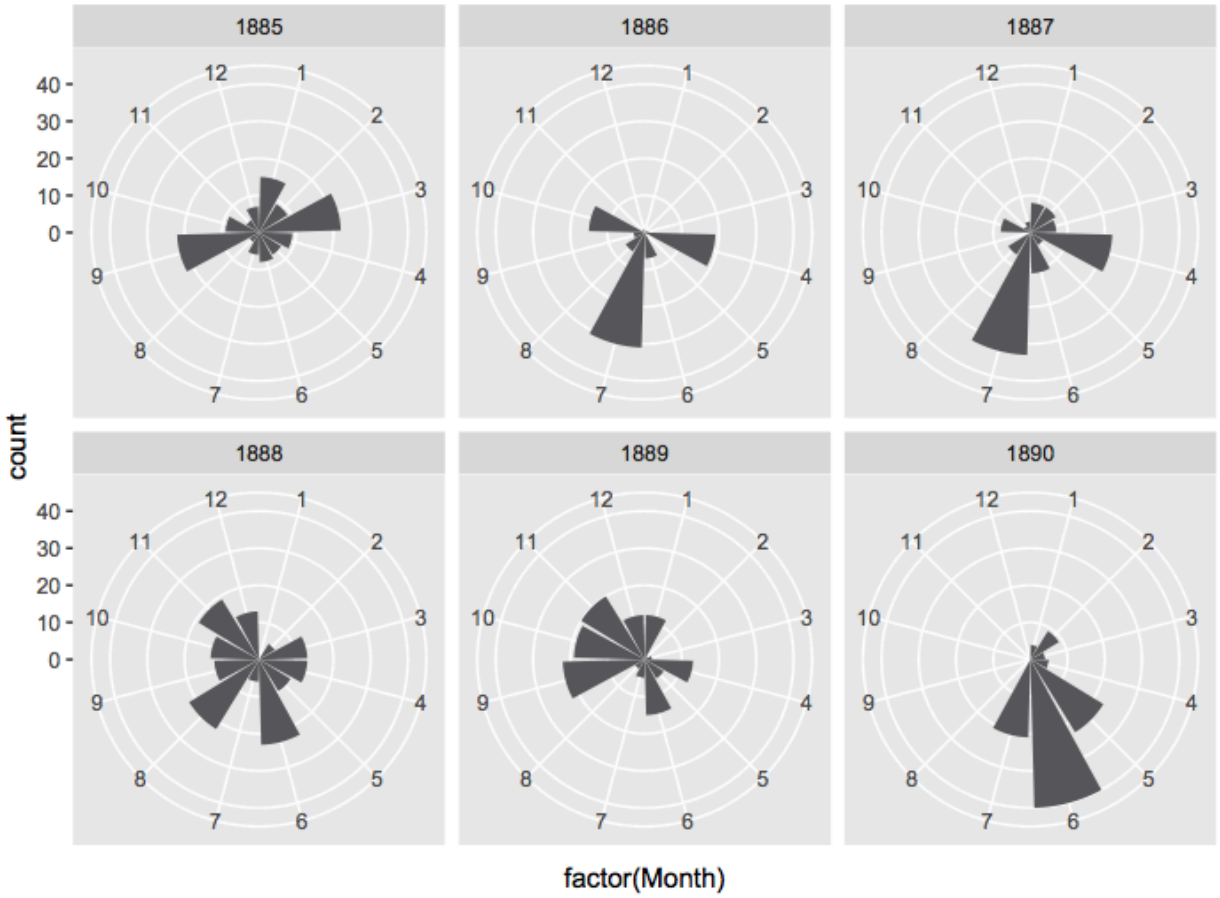
ggplot(xx, aes(factor(Month))) + geom_bar() + coord_polar() + facet_wrap(~ Year)

ggplot(xx, aes(factor(Season))) + geom_bar() + coord_polar() + facet_wrap(~ Label_Place)



example of a visualization showing relations between average (mean) image proportion and Genre category


example of a visualization showing number of paintings in relation to Place and Season




example of a visualization showing number of paintings in relation to Year and Month

## 2| Check these well-known examples of using standard 1D and 2D data visualization techniques with big data - and how to write about big social data for non-technical audiences:

**Google n-Gram viewer:**

http://ngrams.googlelabs.com

More details and the paper: http://www.culturomics.org/

Media coverage:
http://www.nytimes.com/2013/12/08/technology/in-a-scoreboard-of-words-a-c

ultural-guide.html?pagewanted=all&_r=0

Further developments:

http://www.theatlantic.com/technology/archive/2013/10/googles-ngram-viewer-goes-wild/280601/

http://larryferlazzo.edublogs.org/2014/07/24/ny-times-creates-their-own-version-of-googles-ngram-viewer/

**Nicholas Felton: Annual Reports:**

http://feltron.com

**Stephen Wolfram: The Personal Analytics of My Life -**

http://blog.stephenwolfram.com/2012/03/the-personal-analytics-of-my-life/

**OK Cupid blog:**

http://blog.okcupid.com/

## 3| Readings: historical development of statistics and social sciences and contemporary "social physics"

**Required readings:**

**Development of statistics in the 18th-19th century and the idea of "social physics":**

**Philip Ball. *Chapter 3: The Law of Large Numbers* from Critical Mass. 2006.**
The PDF of the chapter will be made available to students.

**Big Data about human behavior and new "social physics":**

http://www.technologyreview.com/review/526561/the-limits-of-social-engineering/

Evgeny Morozov, Every Little Byte Counts, NYT, 5/14/2014.

**And also, to help you make more interesting data visualizations - visualization and emotion:**

Fernanda Viegas and Martin Wattenberg. How to make data look sexy. 4/19/2013

---

# Class 5: Basic descriptive statistical measures in R. Visualizing summarized data in R, continued. Histories of data visualization and statistics.

## 1. Classical concepts and techniques of descriptive statistics

"Statistics may be regarded as (i.) the study of **populations**, (ii.) as the study of **variation**, (iii.) as the study of methods of the **reduction of data**." Ronald A. Fisher. STATISTICAL METHODS FOR RESEARCH WORKERS. (1925) [link]

**Descriptive statistics:**

https://en.wikipedia.org/wiki/Descriptive_statistics

http://onlinestatbook.com/2/introduction/descriptive.html

**Descriptive statistics: Central tendency measure**

https://en.wikipedia.org/wiki/Central_tendency

key measures of central tendency: mean, median, five number summary

**Descriptive statistics: Dispersion measures**

https://en.wikipedia.org/wiki/Statistical_dispersion

key measures of dispersion: variance, standard deviation

**Historical development of descriptive statistics and its use in emerging social sciences:**

Pages from Durkheim's *Suicide* demonstrating his use of aggregate statistics:

https://ia801407.us.archive.org/23/items/lesuicidetudede00durkgoog/lesuicidetudede00durkgoog.pdf

about the book - https://en.wikipedia.org/wiki/Suicide_(book)

---

# Calculating and Visualizing Descriptive Statistics in R:

## Descriptive statistics in R using built-in commands:

http://www.statmethods.net/stats/descriptives.html

# Calculating common descriptive statistics in R:

First, read sample data table into R:

```
xx = read.delim("van_Gogh_genres.txt")
hist(xx$image_proportion, n=40)
```

**Calculating single descriptive statistics statistics measures - continuous variables :**

```
mean(xx$image_proportion)
median(xx$image_proportion)
sd(xx$image_proportion)
```

**Calculating many descriptive statistics together - continuous variables :**

```
summary(xx$image_proportion)
fivenum(xx$image_proportion)
```

**Summarizing discrete (categorical) data:**

Typically we want to count how many cases we have in **one categorical variable**:

```
table(xx$Genre_gen)
```

Or **two** variables:

```
table(xx$Genre_gen, xx$Year)
```

Or **three** variables, and so on:

```
table(xx$Genre_gen, xx$Year, xx$Season)
```

**Visualizing summaries of categorical variables:**

- For a **single variable** - bar plot, point plot, or line plot (point or line plots are the same as bar plots but they show data using points or connected lines);

```
barplot(table(xx$Genre_gen))
plot(table(xx$Genre_gen), type="l")
plot(table(xx$Genre_gen), type="p")
```

- These plots often do not print all labels by default - to force them **to print all labels**, use lax=2 option:

```
plot(table(xx$Genre_gen), type="p", las=2)
```

- You can also rotate barplot using horiz=TRUE option:

```
barplot(table(xx$Genre_gen), las=2, horiz=TRUE)
```

For **two categorical variables:**

- Grouped bar plot:

```
xx.table= table(xx$Label_Place, xx$Season)
barplot(xx.table, beside=TRUE)
```

More examples and options:

http://www.statmethods.net/graphs/bar.html

You can also use (less common) **mosaic plot**:

```
plot(table(xx$Genre_gen, xx$Year))
```

# Statistical summaries and visualizations of parts of a dataset (groups, factors, categories):

See this script:
**GC_script_data_summary.R**

## Counting how many factors are in another factor:

This is a kind of summary appropriate for categorical data - instead of using descriptive statistics such as mean and sd, we instead count number of items in each category:

```
// count how many genres appear in van Gogh paintings in each place
aa = colSums( xtabs( ~ Genre_gen + Label_Place , xx ) !=0 )
```

// different way to do the same
aa = with(xx, tapply(Season, Label_Place, function(x) length(unique(x))))
**Counting numbers of rows using two groups:**

aggregate(xx **by** = list(xx$Genre_gen, xx$Label_Place), FUN = length)

## Summarizing continuous data by group:

There are many ways to do this in R - here are some:

http://www.statmethods.net/management/aggregate.html

http://stackoverflow.com/questions/9847054/how-to-get-summary-statistics-by-group

http://stats.stackexchange.com/questions/8225/how-to-summarize-data-by-group-in-r

**Using tapply():**

tapply(xx$image_proportion, xx$Genre_gen, mean)

A guide to "apply" functions in R:

http://stackoverflow.com/questions/3505701/r-grouping-functions-sapply-vs-lapply-vs-apply-vs-tapply-vs-by-vs-aggrega

**tapply** - "*For when you want to apply a function to **subsets** of a vector and the subsets are defined by some other vector, usually a factor.*"

**Using aggregate()**

aggregate() does the same as tapply but it produces a **data frame** which is easier to further analyze and visualize:

This format **uses categories in *one* variable to aggregate all other variables:**

```
attach(mtcars)

aggregate(mtcars, by=list(cyl), FUN=mean)

aggregate(mtcars, by=list(cyl), FUN=fivenum)
```

This format **uses categories in *two* variable to aggregate all other variables:**

```
aggregate(mtcars, by=list(cyl,vs), FUN=mean)
```

This format **uses categories in ONE variable to aggregate another SINGLE variable:**

```
aggregate(instagram_id ~ username, data=xx, FUN=sum)
```

We can define **our own function for aggregation** - for example, to count number of cases in a categorical variable (in this case, we are counting how many genres van Gogh painted in in each place he lived):

```
aggregate(Genre ~ Label_Place, data=xx, FUN=function(x) length(unique(x)))
```

## Using ggplot2 built-in statistics commands:

The following comes from:

http://www.dummies.com/how-to/content/how-to-plot-summarized-data-in-a-ggplot2-in-r.html

""One very convenient feature of ggplot2 is its range of functions to summarize your R data in the plot. This means that you often don't have to pre-summarize your data."

| Stat | Description | Default Geom |
|---|---|---|
| `stat_bin()` | Counts the number of observations in bins. | `geom_bar()` |
| `stat_smooth()` | Creates a smooth line. | `geom_line()` |
| `stat_sum()` | Adds values. | `geom_point()` |

| | | |
|---|---|---|
| `stat_identity()` | No summary. Plots data as is. | `geom_point()` |
| `stat_boxplot()` | Summarizes data for a box-and-whisker plot. | `geom_boxplot()` |

**How to tell ggplot2 to leave your data unsummarized**

"Sometimes you don't want ggplot2 to summarize your data in the plot. This usually happens when your data is already pre-summarized or when each line of your data frame has to be plotted separately.

In these cases, you want to tell ggplot2 to do nothing at all, and the stat to do this is stat_identity()."

## Plotting data summaries by group:

**Make sorted bar plot of summary statistics by group:**

```
barplot(sort(tapply(xx$image_proportion, xx$Genre_gen, sd)))
```

Same but plotting every label:

```
barplot(tapply(xx$image_proportion, xx$Genre_gen, sd), las=2)
```

**Plot statistics by group using ggplot2:**

Requires two steps:

1) `mm=aggregate(image_proportion ~ Genre_gen, data=xx, FUN=function(x) mean(x))`
2) `ggplot(mm, aes(reorder(Genre_gen,image_proportion), image_proportion)) + geom_point(size=3) + coord_flip()`

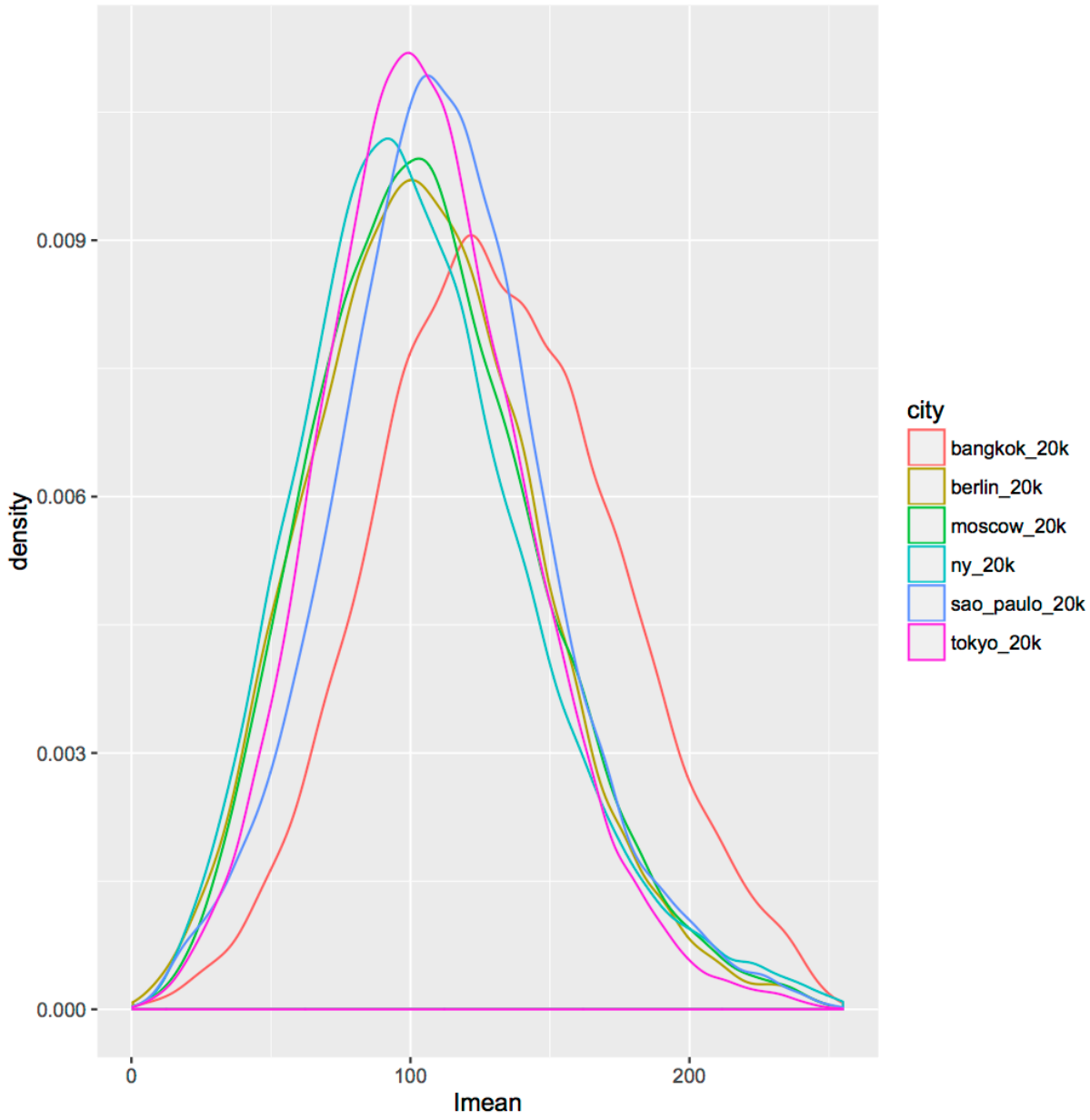
**Make line plots of distributions of one variable in every group:**

To do this, you need to convert your data frame from its standard format (called “wide data” in R) - where each variable in its column - to a “long format.”

```
// read data into R
x = read.delim("broadway-crosstab.LM.all.txt")

// make a copy of the data frame leaving only columns containing variables you want to plot
xx = x[c(1,8:10)]

// load reshape2 library
library(reshape2)

// use melt() to convert this data into long format
xxx = melt(xx,id="Id_num")

// plot using ggplot()
ggplot(xxx, aes(x=Id_num,y=value,colour=variable,group=variable)) + geom_line()
```

**Line plot of distributions of data in each group using geom_density()**

```
xx = read.delim("van_Gogh_genres.txt")

ggplot(xx, aes(x=brightness_median, colour=Genre_gen,group=Genre_gen)) + geom_density()
```




**More resources - visualising distributions of data and data parts using ggplot2:**

http://www.r-bloggers.com/ggplot2-cheatsheet-for-visualizing-distributions/

http://www.fdawg.org/FDAWG/Tutorials/ggplot2.html

### P.S. Combining multiple ggplot2 graphs together:

http://www.cookbook-r.com/Graphs/Multiple_graphs_on_one_page_(ggplot2)/

---

# Homework for Class 6

**1) Practice summarizing and visualizing statistics of data parts (using R)**

Go through commands and explanations in *Notes for Class 5* (above) and keep practicing summarizing and visualizing statistics of parts of a dataset. You can also use History of our class R session I emailed you (van Gogh dataset used there is the same you got for your homework).

Note that you can use many built-in datasets in R for practicing this:

list all available datasets:
> data()

to see contents of any particular dataset just type its name in R:
> Titanic
> mtcars

**2) Readings - urban research using contemporary data sources (Spin Unit):**

http://www.spinunit.eu/urban-meta-morphology/

**3) View innovative mapping projects:**

http://infosthetics.com/cgi-bin/mt/mt-search.cgi?search=map&IncludeBlogs=1&limit=20

**3) Optional - practice making maps using R:**

http://www.r-bloggers.com/minimalist-maps/

http://www.arilamstein.com/blog/2016/01/25/mapping-us-religion-adherence-county-r/

http://www.r-bloggers.com/create-your-own-hexamaps/

http://www.r-bloggers.com/visualising-thefts-using-heatmaps-in-ggplot2/

http://www.computerworld.com/article/3038270/data-analytics/create-maps-in-r-in-10-fairly-easy-steps.html

**Great tutorials on making maps in R:**

http://bcb.dfci.harvard.edu/~aedin/courses/R/CDC/maps.html

**Using ggmap package:**

https://journal.r-project.org/archive/2013-1/kahle-wickham.pdf

More mapping tutorials:

http://www.r-bloggers.com/search/maps/page/2/

---

# Homework for Class 7

## If you have no previous knowledge of descriptive statistics:

> **do this free online course** (or use any other online resource to become familiar with basic techniques and practice them in R):
> https://www.khanacademy.org/math/probability/descriptive-statistics
>
> **Go through chapters 5-7:**
>
> https://ischool.syr.edu/media/documents/2012/3/DataScienceBook1_1.pdf

## Everybody: Practice data visualization using ggplot2 -

http://www.r-bloggers.com/ggplot2-cheatsheet-for-visualizing-distributions/
http://www.fdawg.org/FDAWG/Tutorials/ggplot2.html

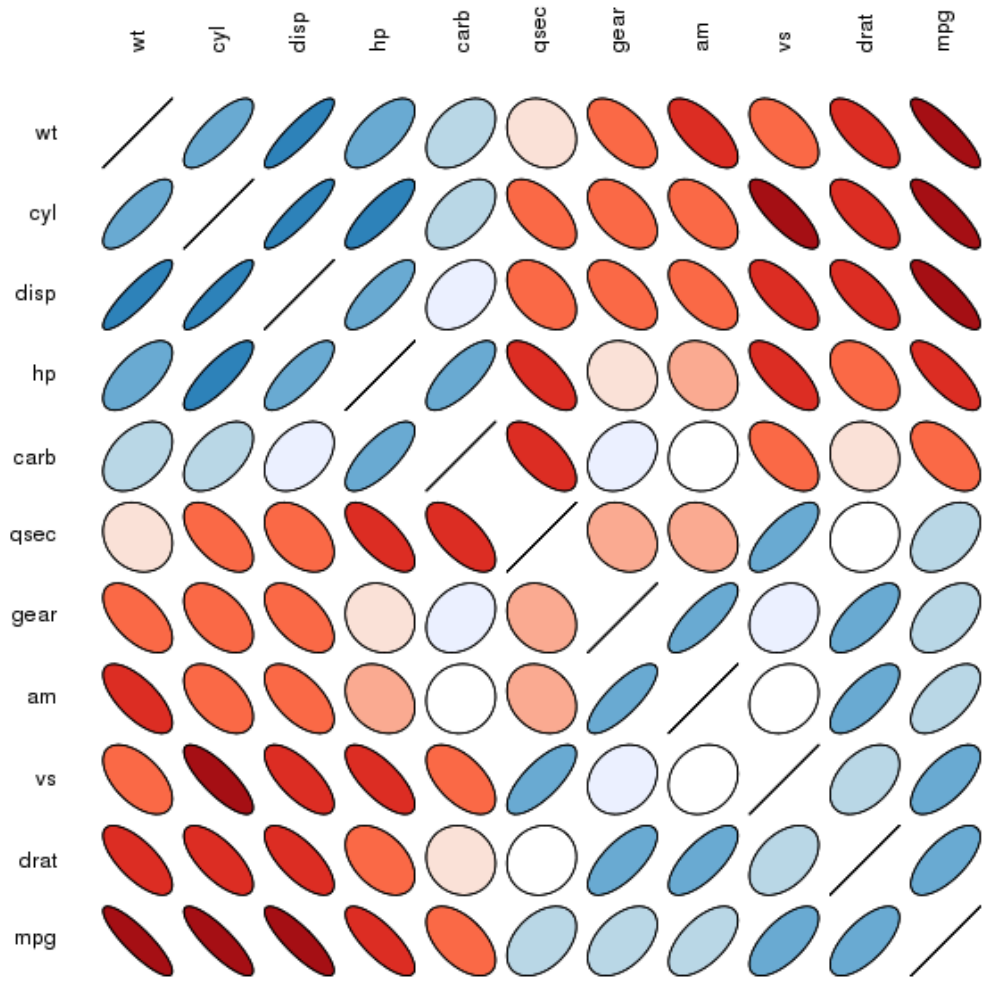
**Everybody: readings about history of statistics -**

1) https://en.wikipedia.org/wiki/History_of_statistics#Development_of_modern_statistics

2) Ian Hacking, Introduction to The Taming of Chance. Cambridge University Press, 1990.

---

# Class 7:
# Correlation and linear regression (using R). Visualization of correlations between many variables. Fitting trend lines. Plotting multiple time series (long vs. wide data in R). Changing continuous variables into categorical variables (cut command in R).




Above: Example of correlation matrix ellipses

## More history of statistics and social thought:

**1830s: Discoveries that physical characteristics of people fit "normal distribution" - "average human"**
E. B. Taylor, Quentelet on the Science of Man, *Popular Science*, Volume 1 (May 1872).

**19th century: variation, correlation, regression:**
http://en.wikipedia.org/wiki/Francis_Galton#Variance_and_standard_deviation
http://en.wikipedia.org/wiki/Francis_Galton#Correlation_and_regression

**Summary of statistics as discipline, 1925:**
Ronald Fisher, Introductory chapter from Statistical Methods for Research Workers, 1925.

## Correlation; visualizing correlations in R:

Correlation analysis is a set of statistical concepts and techniques. The basic command to calculate correlation in R is *cor:*

cor(data$x, data$y)

In our class we will be only concerned with use of **correlation in visualization**.

If we only have **two variables**, their scatter plot gives us a visual intuition about their correlation:

http://study.com/academy/lesson/scatter-plot-and-correlation-definition-example-analysis.html

What about correlations between a **number of variables** - how can we visualize this? Solving this problem leads to a few visually interesting and informative visualization techniques:

**Visualize correlations using heatmaps:**

http://www.r-bloggers.com/using-r-correlation-heatmap-with-ggplot2/

http://www.r-bloggers.com/using-r-correlation-heatmap-take-2/

**Other techniques - including correlation matrix ellipses:**

http://stackoverflow.com/questions/5453336/plot-correlation-matrix-into-a-graph

## Regression; fitting lines and curves in R:

Regression analysis is a set of statistical concepts and techniques. It s a way to represent

relations between two or more variables using a mathematical model. The model allows us to predict values of one variable (often called “response variable” or “dependent variable”) from values of another variable (often called “predictor” or “independent variable”):

In the simplest case, we have only one independent variable (x) and one dependent variable (y). If our model has this form, this is called “linear regression” (its visualized as a straight line going through the data):

$Y = a + bX + e$

Explanations of simple linear regression:

https://onlinecourses.science.psu.edu/stat501/node/251

Doing linear regression in R:

http://www.r-bloggers.com/r-tutorial-series-simple-linear-regression/

This is the simplest case. We can also use multiple variables to predict one response variables; and also use multiple variables to predict multiple response variables, etc.

In our class we will only be concerned with **use of regression in data visualization**. In this case, regression is used as a practical technique to show the trend in the data using a line or a curve.

In general, the terms “regression line” and “fit line” are often used interchangeably.

However, in the context of visualization, the terms “fit line,” “fit curve,” or “smoother” (in ggplot2) are used in this context more often than regression.

Here is a tutorial:
http://www.theanalysisfactor.com/linear-models-r-plotting-regression-lines/

Ways to do this using ggplot2:
http://docs.ggplot2.org/0.9.3.1/stat_smooth.html

**Now let's do this using our own dataset:**

First read the data:
x = read.delim("**van_gogh_data.txt**")

**Standard method: plot all points and draw trend line:**

```
ggplot(x, aes(Year_Month, Brightness_Median)) +  geom_point() + stat_smooth(method = "lm")
```

**Alternative Method: first summarize data and then plot and draw trendline :**

```
xx = aggregate(Brightness_Median ~ Year, data = x, mean)
ggplot(xx, aes(Year, Brightness_Median)) +  geom_point() + stat_smooth(method = "lm")
```

**Instead of a line, lets fit a curve:**

```
ggplot(x, aes(Year_Month, Brightness_Median)) +  geom_point() + stat_smooth(method = "loess")
```

**more options for fitting curves:**
http://www.ats.ucla.edu/stat/r/faq/smooths.htm

## Plotting multiple time series - long vs. wide data in R:

Imagine a dataset which has one variable indicating points in a sequence (for example, minutes or hours, etc.), and a number of variables which have values at each point. For example, we measure temperature, wind strength, pressure every hour in some location. This kind of data is often called "multiple time series." How can we plot this with ggplot2?

To do this, you need first to **convert your data from "wide" to "long format."**

**Converting data from wide to long formats:**

http://seananderson.ca/2013/10/19/reshape.html

http://www.r-bloggers.com/converting-a-dataset-from-wide-to-long/

**Plotting data in long format:**

Simple examples:

http://stackoverflow.com/questions/13324004/plotting-multiple-time-series-in-ggplot

http://www.sixhat.net/how-to-plot-multpile-data-series-with-ggplot.html

More advanced examples:

https://plot.ly/ggplot2/time-series/

## Example of plotting multiple lines using our own dataset:

```
x = read.delim("broadway-crosstab.LM.all.txt")
xx = x[c(1,8:10)]
xxx = melt(xx,id="Id_num")
ggplot(xxx, aes(x=Id_num,y=value,colour=variable,group=variable)) + geom_line()
```

## Changing continuous variables into categorical variables (*cut* command):

Sometimes it is useful to change a continuous variable into a categorical variable. This in fact is what histogram command does automatically. **R *cut* command** is more general, and gives you a number of options for "cutting" the data:

http://www.r-bloggers.com/r-function-of-the-day-cut/

http://stackoverflow.com/questions/5746544/r-cut-by-defined-interval

(**see a practical example below**)

# More R and ggplot2 techniques - work through these examples:

## ggplot - using facets:

using our van gogh dataset:

ggplot(x, aes(Month)) + geom_bar() + facet_wrap(~ Year)

ggplot(x, aes(Image_Width, Image_Height)) + geom_point(size=10, alpha=0.5) + facet_wrap(~ Year)

## ggplot2 - Scatter Plots options:

http://www.cookbook-r.com/Graphs/Scatterplots_(ggplot2)/

**Scatter plot with transparency:**
```
ggplot(mtcars, aes(wt, mpg)) + geom_point(alpha = 1/10)
```

or:
```
ggplot(mtcars, aes(wt, mpg)) + geom_point(alpha = 0.05)
```

Note: alpha can be any number between 0 and 1.

This is how you **control points size**:
```
ggplot(mtcars, aes(wt, mpg)) + geom_point(size=0.5)
```

Now we **combine size and alpha parameters:**
```
ggplot(mtcars, aes(wt, mpg)) + geom_point(alpha = 0.05, size=0.5)
```

More options in scatter plots:
http://docs.ggplot2.org/0.9.3/geom_point.html

## Adjusting plot margins:

The following uses part of the data about MoMA collection. MoMA released their data on GitHub in Fall 2015. Helen Wall who was taking my class in Fall 2015 semester started to clean the data, so we will use the partly cleaned dataset she created. Eventually she cleaned all the data and published analysis of patterns in this dataset":
https://medium.com/@WallHelen/120kmoma-ae298a2a57b7#.uoats5ilo

I have emailed you this data: **Artworks_Helen_sel_cols.txt**

```
// read the data
x = read.delim("Artworks_Helen_sel_cols.txt" )
```

```
// found how many rows have missing values
bb = complete.cases(xx)
table(bb)
```

**Adjusting size of labels and making sure all labels are shown:**
(las=2, cex.names=0.8)

```
// plot number of works per MoMA department
barplot(sort(table(xx$Department)), las=2, cex.names=0.8)

// check margins for plots
par("mai")

// increase bottom margin
par(mai=c(3.00,0.82,0.82,0.42))

// redo the plot
barplot(sort(table(xx$Department)), las=2, cex.names=0.8)
```

// we notice in the plot that one of the categories has no name - these are the works which do not have department in the data file

```
// to check this in the data, we use length(unique(x))
length(unique(xx$Department))

// we will remove the rows with missing values to make a plot of this column
xx2 <- xx[-which(xx$Department == ""), ]

// check the new data frame
length(unique(xx2$Department))

// redo the plot
par(mai=c(3.00,0.82,0.82,0.42))
barplot(sort(table(xx2$Department)), las=2, cex.names=0.8)

// add the title
barplot(sort(table(xx3$Department)), las=2, cex.names=0.8, main="Number of artworks in
MoMA Departments")
```

**Changing continuous variables into categorical variables (cut):**

```
// now let us make a plot of artworks dates
hist(xx$Merged_Dates)

// plot reveals that some artworks have very early dates in the data file - lets check these
numbers:
fivenum(xx$Merged_Dates)
head(sort(xx$Merged_Dates)

// remove one record with 1600 date
xx3 = xx[-which(xx$Merged_Dates == "1600"), ]

// redo the graph and adjust the parameters
hist(xx3$Merged_Dates)
hist(xx3$Merged_Dates, n=40)

// let's make exact bins at 1 year intervals
xx4 = xx3
xx4$years = cut(as.numeric(as.character(xx4$Merged_Dates)), breaks = seq(1760, 2020),
include.lowest=TRUE)
table(xx4$years)
barplot(table(xx4$years))
barplot(table(xx4$years), xaxt = "n")
```

// lets make exact bins at 5 year intervals
xx4$five = cut(as.numeric(as.character(xx4$Merged_Dates)), breaks = seq(1760, 2020, by=5), include.lowest=TRUE)
table(xx4$five)
barplot(table(xx4$five), las=2, cex.names=0.5)
barplot(table(xx4$five), las=2, cex.names=0.5, horiz=TRUE)

**Counting instances of one set of categories inside another set of categories (review):**

Review - some of the ways to do this in R:

http://stackoverflow.com/questions/16604380/how-to-count-levels-of-a-factor-in-a-data-frame-grouped-by-another-value-of-tha

http://stackoverflow.com/questions/15280362/in-r-how-do-i-count-the-occurrences-of-a-factor-in-several-columns-and-group-by

// **New:** count how many artists names are in each department
// in R this is called counting how many factors are in another factor
**aa = colSums( xtabs( ~ Artist + Department , xx ) !=0 )**

// different way to do the same you already know
**aa = with(xx, tapply(Artist, Department, function(x) length(unique(x))))**

// plot this result as pie chart
pie(aa)

// adjust margins and plot again
par(mai=c(2, 2, 2, 2))
pie(aa)

// **New:** Now let's count how many works each artist has in MoMA collection -
using **data tables** package - this method is appropriate if your data has many rows

// to learn how to use data tables:

// **https://www.datacamp.com/courses/data-table-data-manipulation-r-tutorial**

// **http://blog.datacamp.com/data-table-r-tutorial/**

```
xx5 = xx[,c(2,8)]

library(data.table)
data_t = data.table(xx5)
setkey(data_t, Artist)
xx6 = data_t[, table(Artist)]

tail(sort(xx6))
par(mai=c(0.82,2.00,0.82,0.42))
barplot(tail(sort(xx6), n=20), las=2, cex.names=0.6, horiz="true")
```

---

# Final Project

**Deadline: June 1, 2016 at noon.**

Create a short visual essay about the topic of their choice - so you can use your educational background and interests. The topic should be of interest to general audiences as opposed to narrow professional audiences.

The essay should include a
**Few visualizations of some relevant dataset(s) you find or create.**

The essay should include some discussions/explanation of patterns in the visualizations.

You can (I encourage you) include photos, video, maps, etc.

So you can think about your essay as “data journalism” piece - but you are limited by journalism’s rules. You can talk about the past, present or future.

**Text length**: between 800 and 1200 words.

**Format** can be anything: Google doc, Word doc, PDF, a webpage, a long blog post, etc. Visualizations can be static, animated or interactive (which is easy to do using Google Docs).

Here are some **examples of such essays** - they some use sophisticated interactive visualizations, and I don't expect you to produce something like this, but the overall structure - presenting some story using a number of visualizations - is what you should also use. (Note that these essays are longer than what you need to write).

Ex-student who took this class:
https://medium.com/@WallHelen/120kmoma-peak-years-80b9c55fc734#.a2p4cu7su

http://www.nytimes.com/interactive/2014/12/12/upshot/where-men-arent-working-map.html ?

http://qz.com/465820/how-brand-new-words-are-spreading-across-america/

http://www.nytimes.com/interactive/2014/09/19/travel/reif-larsen-norway.html

http://www.nytimes.com/interactive/2014/12/23/us/gender-gaps-stanford-94.html

http://blog.okcupid.com/index.php/race-attraction-2009-2014/

http://blog.okcupid.com/index.php/the-best-questions-for-first-dates/

You can **create your own data**. For example, let's say you want to count and plot types of objects and proportions of these types that appear across a number of Instagram photos in "flat lay" genre. Or maybe you want to spend time in a cafe and record activities and their numbers (working on laptop, chatting on a phone, talking to another person, etc.)

Or you can use **existing dataset(s) available online** about some subjects.

Here are **examples of public datasets** online:

Economics data:

http://www.bls.gov/cps/cpsaat11.htm

http://www.nber.org/data/

Museums data:

https://github.com/cooperhewitt/collection

https://github.com/MuseumofModernArt/collection

City data:

https://www.citibikenyc.com/system-data

https://nycopendata.socrata.com/

Social media data:

https://snap.stanford.edu/data/

**Lists of of  datasets:**

http://www.kdnuggets.com/datasets/index.html

https://github.com/caesar0301/awesome-public-datasets