Big Data Analysis

Lec02. Data Exploration and Visualisation with Python

W1 Recap: Data Science pipeline

1. Ask an interesting question:

- What is the goal?
- What would you do if you had all the data?
- What do you want to predict or estimate?

2. Get the data:

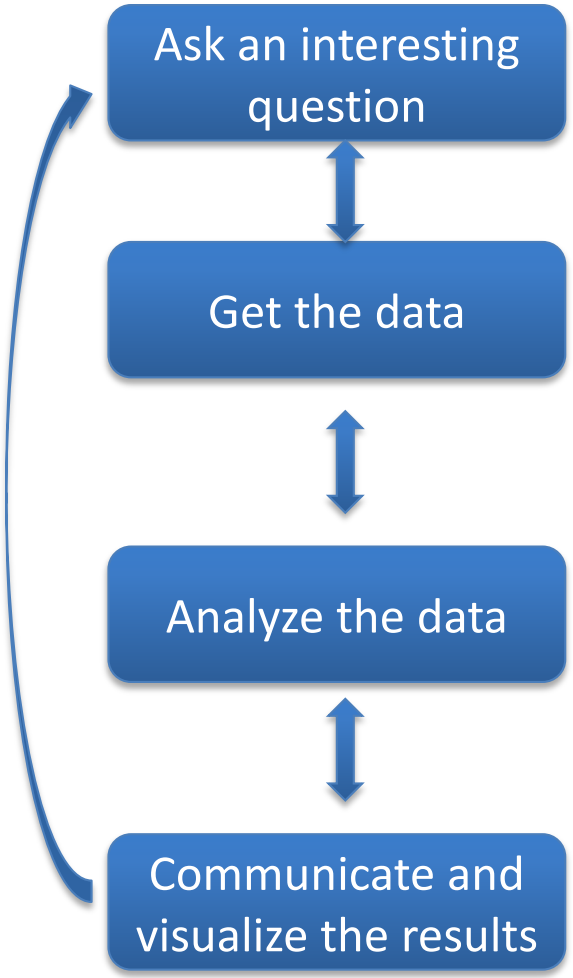
- How were the data sampled?
- Which data are relevant?
- Are there privacy issues?

3. Analyze the data:

- Are there anomalies?
- Are there patterns?
- Are there trends?

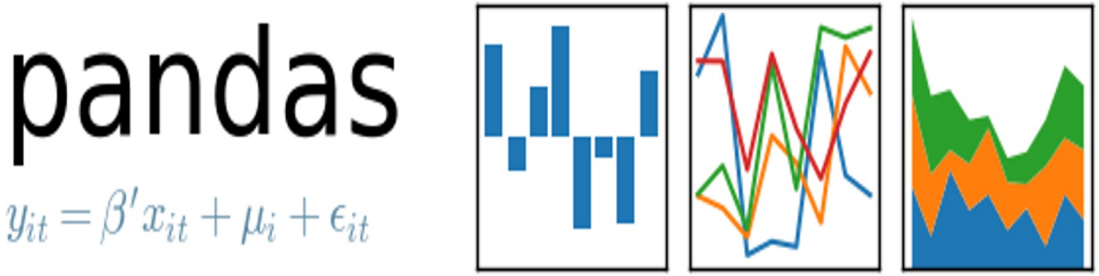
4. Communicate and visualize the results

- What did we learn?
- Do the results make sense?
- Can we tell a story?



W1 Recap: Python for Data Analytics

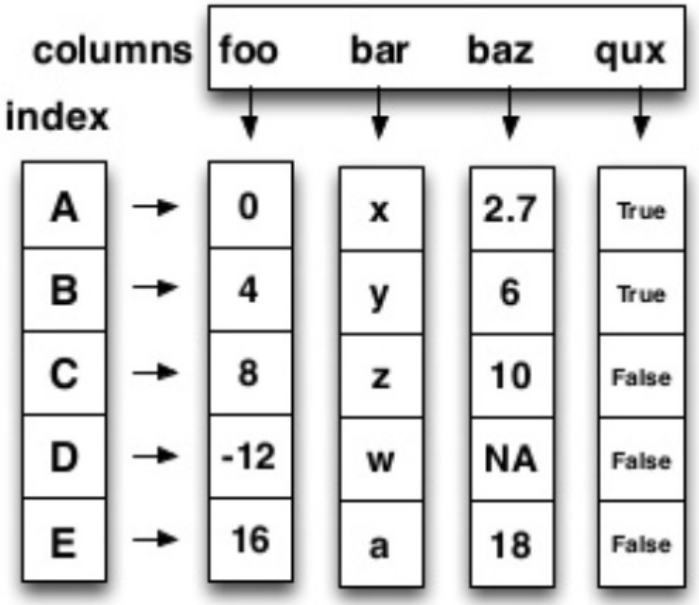
https://seaborn.pydata.org/



- Numpy: great for handling numbers, vectors, matrices
- Scipy: great for numerical optimizations
- Pandas: great for handling tabular/relational data
- Scikit Learn: great for data analytics techniques

W1 Recap: Data Storage with Pandas

- DataFrame: a table with named columns (like in the relational model)
  - Represented as a dictionary
    - columnName -> series
  - Each Series object represents a column
  - Each column can have a different type
  - Row and column indices
  - Size mutable: insert and delete columns



- Why Use DataFrames?
  - better for series with multiples attributes of different types.
  - Easy and efficient search elements by index.

W1 Recap: Clean “dirty” data

- Bad formats
- Missing data
- Erroneous data
- Irrelevant data
- Inconsistent data
- Malicious data
- Outliers

Course structure

W1. Data Processing with Python

W2. **Data Exploration with Python**

W3. Data Modeling with Pytyhon

W4. Data Analytics for Timeseries

W5. Holiday

W6-7. Data Analytics for Texts

W8. Data Analytics for Images

W9. Data Analytics for Graphs

W10-11. Data Analytics for Other Data

W12. Revision

Learning Outcomes

- At the end of this lecture students will be able to know:
  I. Data Exploration
  II. Data Visualisation
  III. Dimensionality Reduction
  IV. Data Sampling (OPTIONAL)

I. Data Exploration

Exploring Two Variables

- Investigating the relationship between different variables
  - Note: association does not imply causation
- Exploring two variables:
  - the explanatory variable (aka the independent variable) - the variable that claims to explain, predict or affect the response; and
  - the response variable (aka the dependent variable) - the outcome of the study.
- Examples:
  - Are there differences in test scores between males and females?
  - Can you predict a person's favorite type of music (classical, rock, jazz) based on his/her IQ level?
  - How is the number of calories in a hot dog related to (or affected by) the type of hot dog (beef, meat or poultry)? In other words, are there differences in the number of calories among the three types of hot dogs?

Exploring Two Variables (cont'd)

- C→Q: categorical to quantitative
- C→C: categorical to categorical
- Q→Q: quantitative to quantitative
- Q→C: not studied

		Response	
		Categorical	Quantitative
Explanatory	Categorical	C→C	C→Q
	Quantitative	Q→C	Q→Q

C→Q case

- C→Q: categorical to quantitative

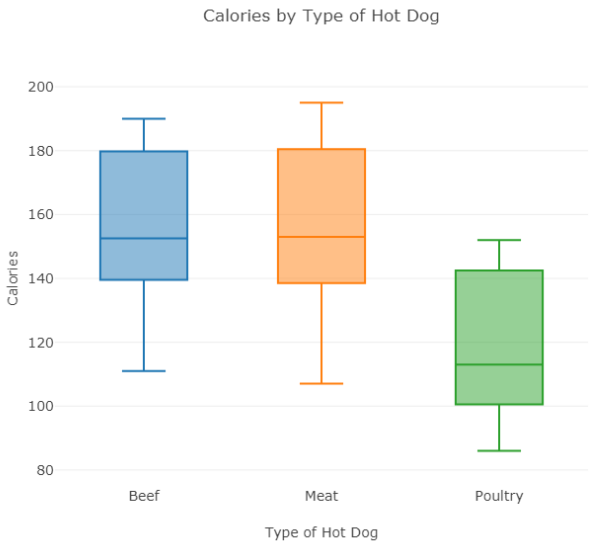
Explanatory → Type

Response → Calories

	Type	Calories
Brand 1	Beef	186
Brand 2	Poultry	129
Brand 3	Beef	181
Brand 4	Meat	173
.	.	.
.	.	.
.	.	.
Brand 54	Poultry	144

- How to present: box-and-whisker plot

Statistic	Beef	Meat	Poultry
min	111	107	86
Q1	139.5	138.5	100.5
Median	152.5	153	113
Q3	179.75	180.5	142.5
Max	190	195	152



From categorical to categorical

- C→C: categorical to categorical

Explanatory ↖ Response ↗

Student	Gender	Body Image
.	.	.
.	.	.
student 25	M	overweight
student 26	M	about right
student 27	F	underweight
student 28	F	about right
student 29	M	about right
.	.	.
.	.	.

- How to present:
  - Two-way table

		Body Image			
		About Right	Overweight	Underweight	**Total**
Gender	Female	560	163	37	760
	Male	295	72	73	440
	Total	855	235	110	1200

  - Compute conditional percents

		Body Image			
		About Right	Overweight	Underweight	**Total**
Gender	Female	560/760 = 73.7%	163/760 = 21.5%	37/760 = 4.9%	760/760 = 100%
	Male	? %	? %	? %	? %

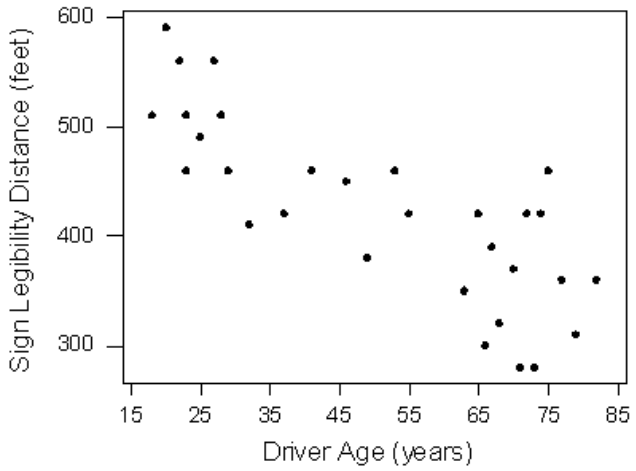
From quantitative to quantitative

- Q→Q: quantitative to quantitative
  - Formally, how close two variables have a linear/non-linear relationship between each other

Explanatory ↖ Response ↗

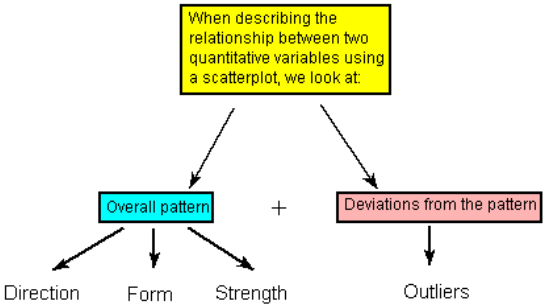
	Age	Distance
Driver 1	**18**	**510**
Driver 2	32	410
Driver 3	55	420
Driver 4	23	510
.	.	.
.	.	.
.	.	.
Driver 30	82	360

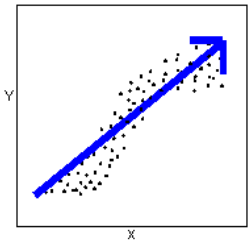
- How to present:
  - Scatter plot



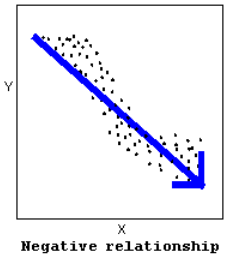
From quantitative to quantitative (cont'd)

- Q→Q: quantitative to quantitative
- How to interpret the scatter plot
  - Direction

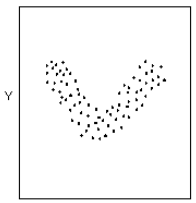




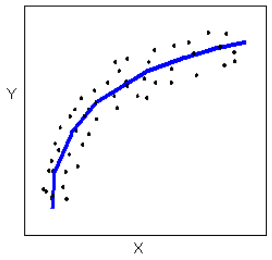
Positive relationship



Negative relationship



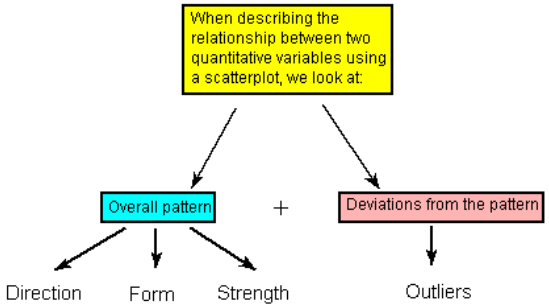
Neither positive nor negative

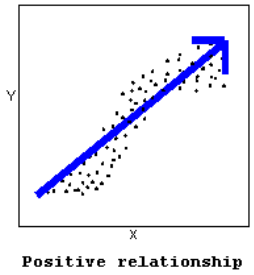


non-linear

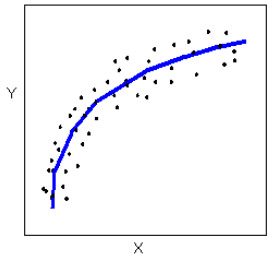
From quantitative to quantitative (cont'd)

- Q→Q: quantitative to quantitative
- How to interpret the scatter plot
  - Form





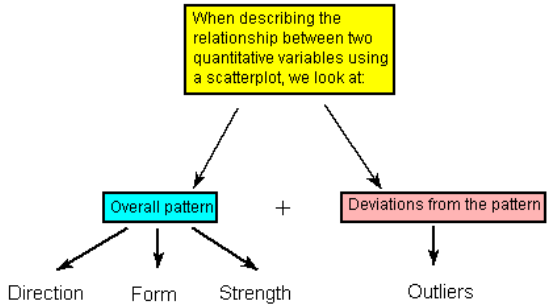
Linear

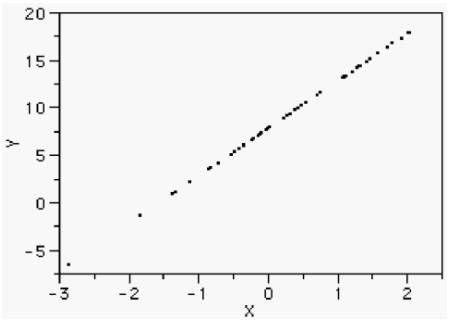


non-linear

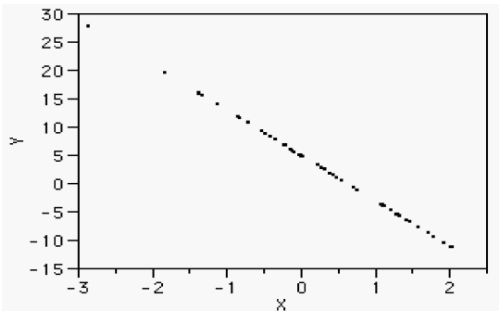
From quantitative to quantitative (cont'd)

- Q→Q: quantitative to quantitative
- How to interpret the scatter plot
  - Form
- If the form is linear
  - Measure correlation coefficient
  - $r = \frac{cov(X,Y)}{\sigma_X \sigma_Y}, -1 \leq r \leq 1$

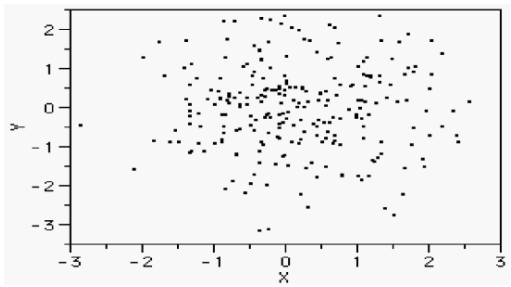




A perfect linear relationship r = 1



A perfect negative linear relationship r = -1

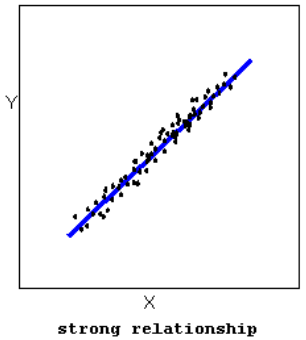


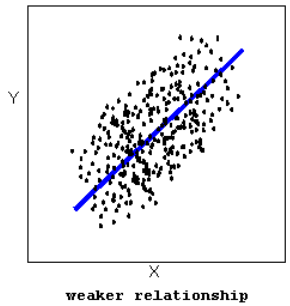
No relationship, r = 0

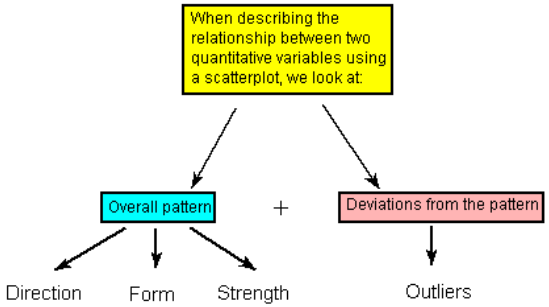
Properties of correlation coefficient:
https://lagunita.stanford.edu/courses/OLI/StatReasoning/Open/courseware/eda_er/_m5_linear/?child=first

From quantitative to quantitative (cont'd)

- Q→Q: quantitative to quantitative
- How to interpret the scatter plot
  - Strength

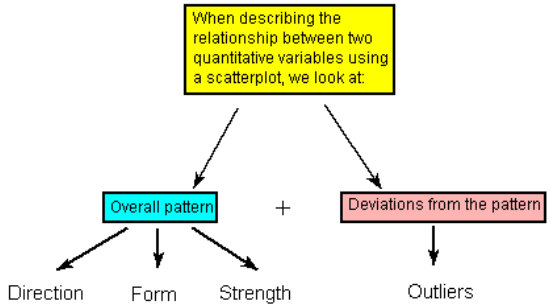


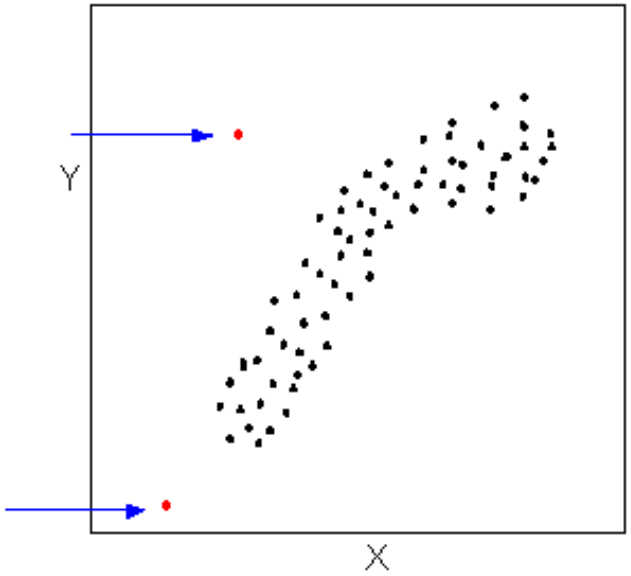




From quantitative to quantitative (cont'd)

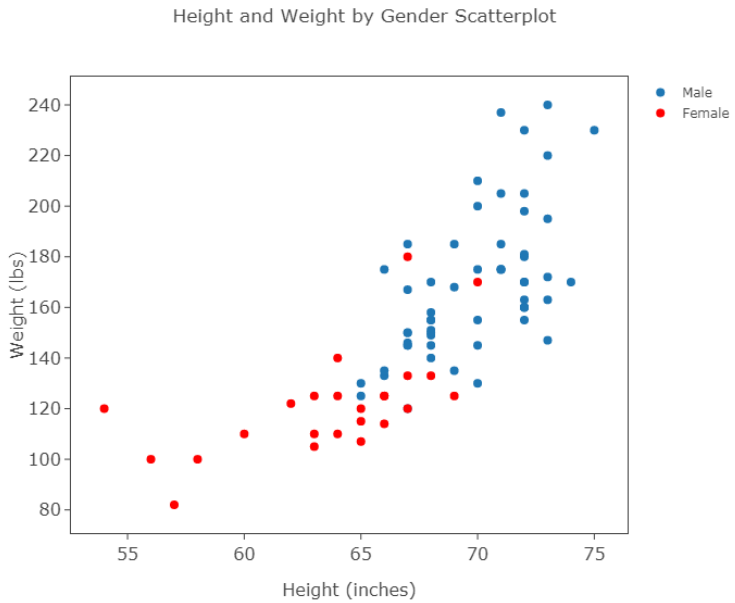
- Q→Q: quantitative to quantitative
- How to interpret the scatter plot
  - Outliers





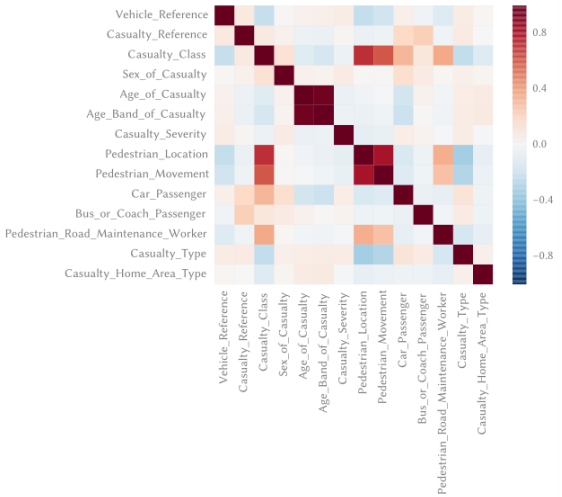
Exploring Multiple Variables

- Investigate the dependence between multiple variables at the same time
- How to present:
  - **Labeled scatter plot:** it may be reasonable to indicate different subgroups or categories within the data on the scatterplot, by labeling each subgroup differently



Exploring Multiple Variables (cont'd)

- Investigate the dependence between multiple variables at the same time
- How to present:
  - **Correlation matrix :**
    - output: a symmetric matrix where element m_{ij} is the correlation coefficient between variables i and j
    - note: diagonal elements are always 1
    - can be visualized graphically using a heatmap
    - allows you to see which variables in your data are informative



II. Data Visualisation

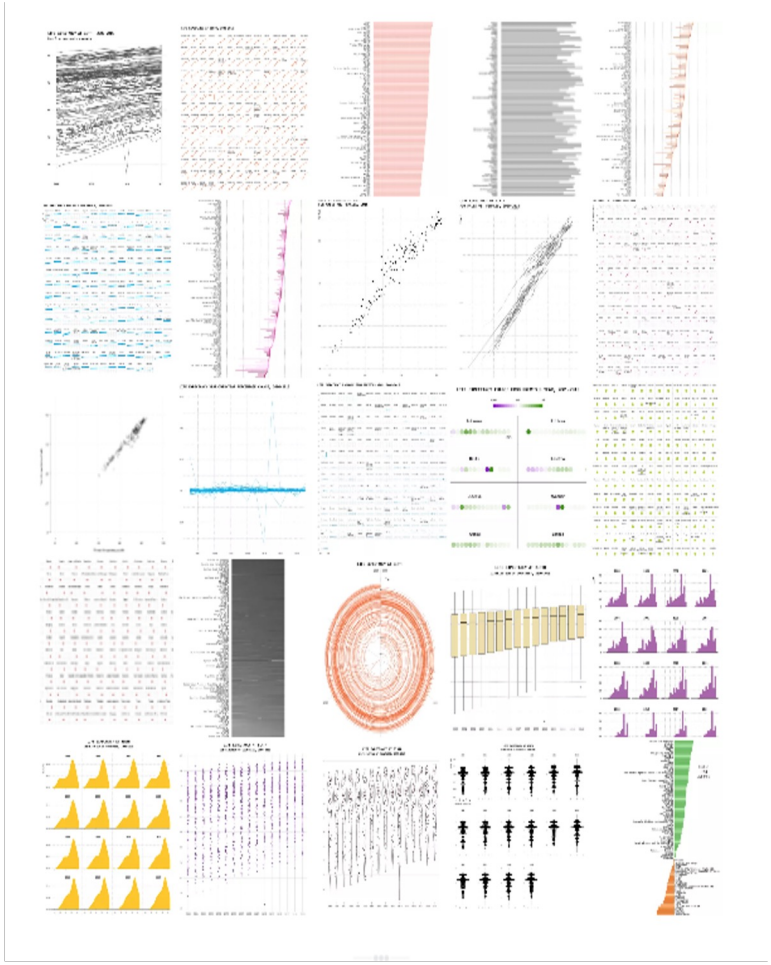
Data Visualization

- "*... finding the artificial memory that best supports our natural means of perception.*" [Bertin 1967]

- "*Transformation of the symbolic into the geometric*" [McCormick et al. 1987]

- "*The use of computer-generated, interactive, visual representations of data to amplify cognition.*" [Card, Mackinlay, & Shneiderman 1999]

One Dataset, Visualized 25 Ways



"You must help the data focus and get to the point. Otherwise, it just ends up rambling!"

http://flowingdata.com/2017/01/24/one-dataset-visualized-25-ways/

From Data Visualization to Visual Analytics

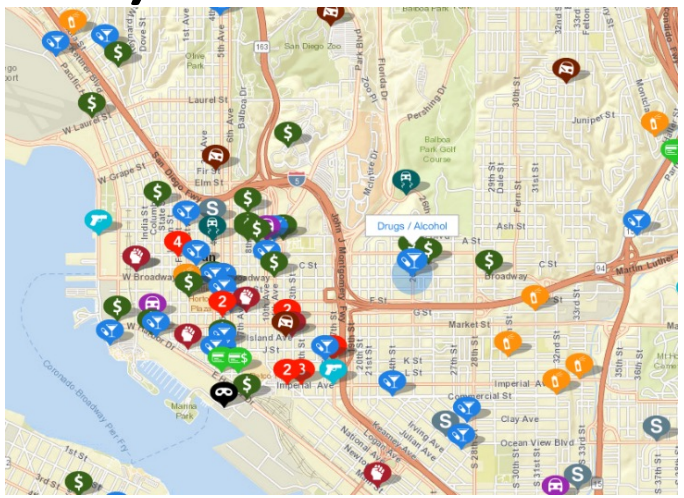
- Interactive viz = Old-fashioned viz + Interaction Scheme
  - Enable **visual analytics** via interactive and reproducible results
  - Easy and fast to develop and customize

- Old-fashioned viz
  - Great for data exploration, developed throughout the last few centuries
  - Rapid data exploration
  - Focus on most important details

- Interactive viz
  - More and more common nowadays. New frameworks are the key enabler.
  - Support multiple analyses
  - Focus on more dimensions

From Data Visualization to Visual Analytics (cont'd)

- Map-based analytics, such as CrimeMapping

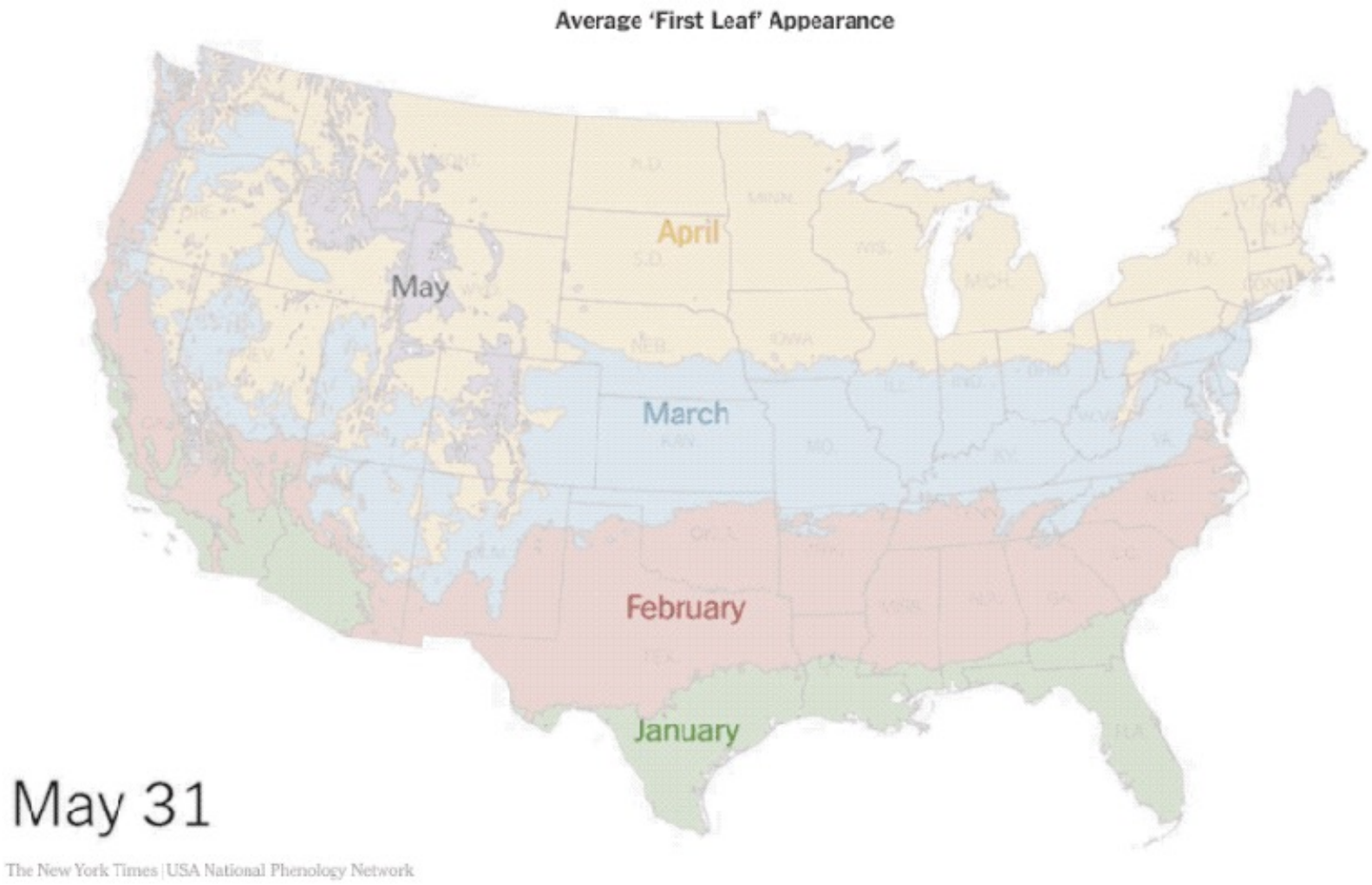


https://www.crimemapping.com/map

- Interactive Education
  - The famous Gapminder Video, Hans Rosling: 200 Countries, 200 Years, 4 Minutes. https://www.youtube.com/watch?feature=player_embedded&v=jbkSRLYSojo

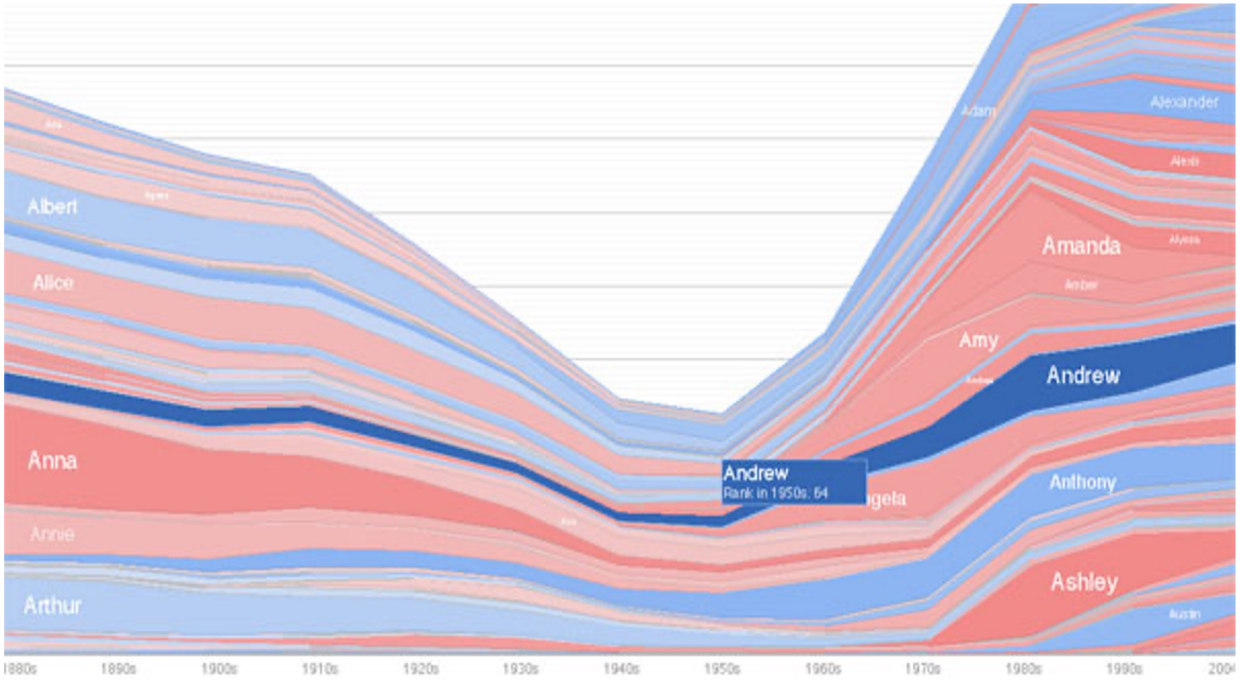
- Future of Journalism: e.g. NY Times
  - NY Times Interactive Visualizations (recession/recovery 2014). http://www.nytimes.com/interactive/2014/06/05/upshot/how-the-recession-reshaped-the-economy-in-255-charts.html
  - And 2014 "the year in interactive storytelling". http://www.nytimes.com/interactive/2014/12/29/us/year-in-interactive-storytelling.html?_r=0

Another Visual Analytics Example



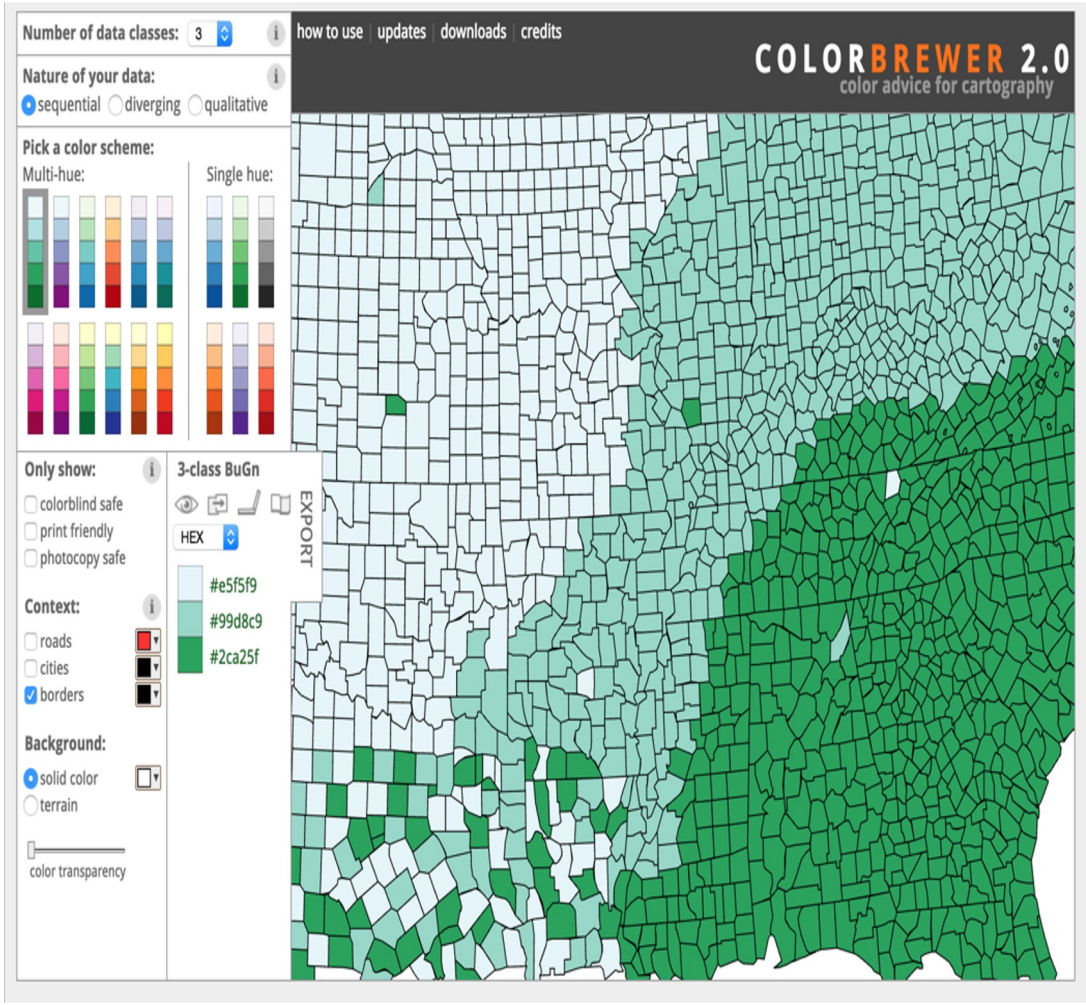
Why Interactive?

- With interactive charts you can keep things very simple by **hiding** and **dynamically revealing** important structure.
- On an interactive chart, you reveal the information most useful for **navigating** the chart.

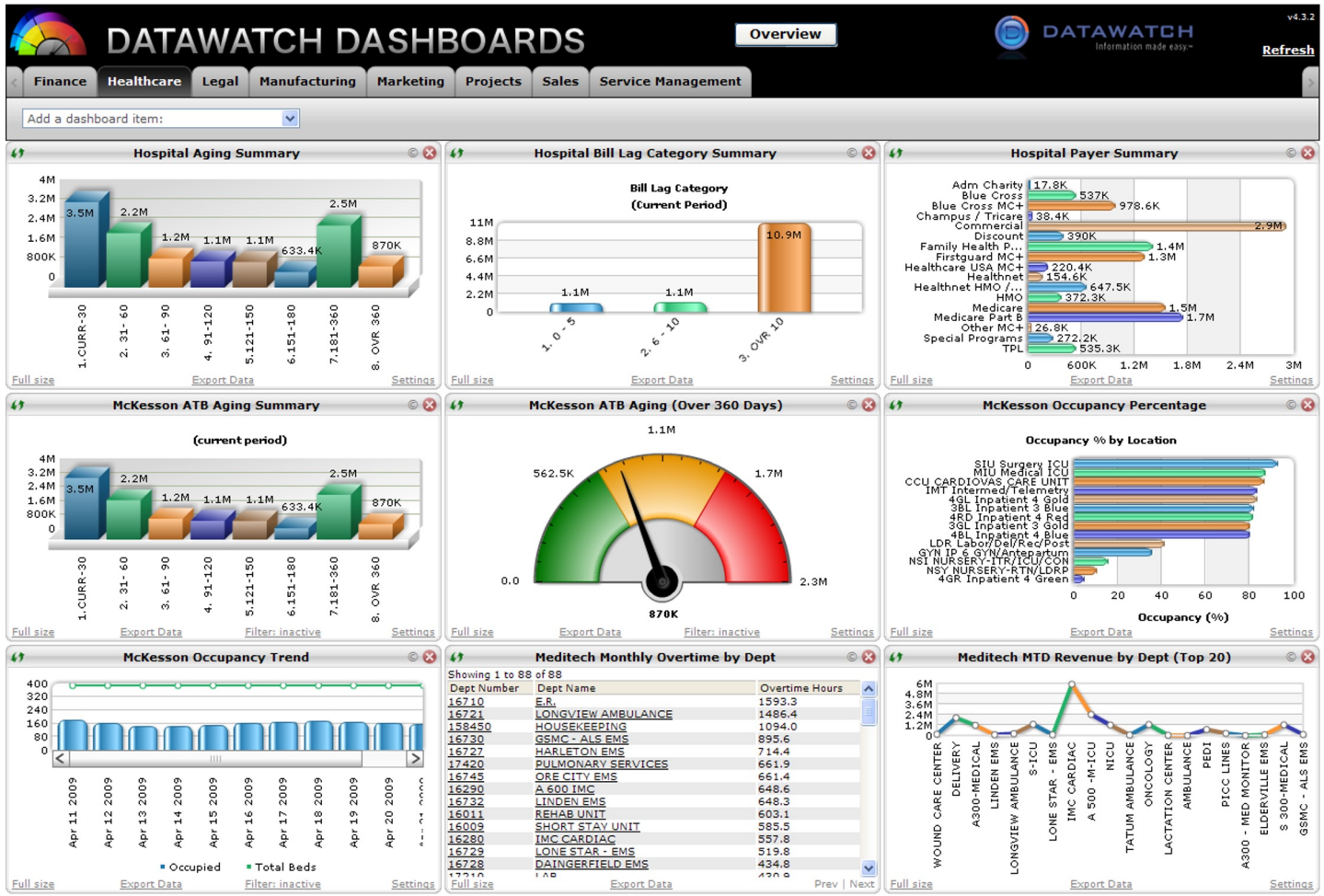


Why interactive?

- A translation between visual information and raw data



Visual Analytics: Dashboards



Source https://www.vocalabs.com/blog/my-dashboard-pet-peeve

Exploratory data analysis with Matplotlib

- Office website: http://matplotlib.org/
- Matplotlib allows you to explore data and create reproducible visual results (2D and 3D figures) programmatically
- Features:
  - Generally easy to get started for simple plots
  - Support for custom labels and texts
  - Great control of every element in a figure
  - High-quality output in many formats
  - Very customizable in general
- Installation:

  conda install matplotlib

  Or pip install matplotlib
- How to use in Jupyter notebook:

  import matplotlib.pyplot as plt

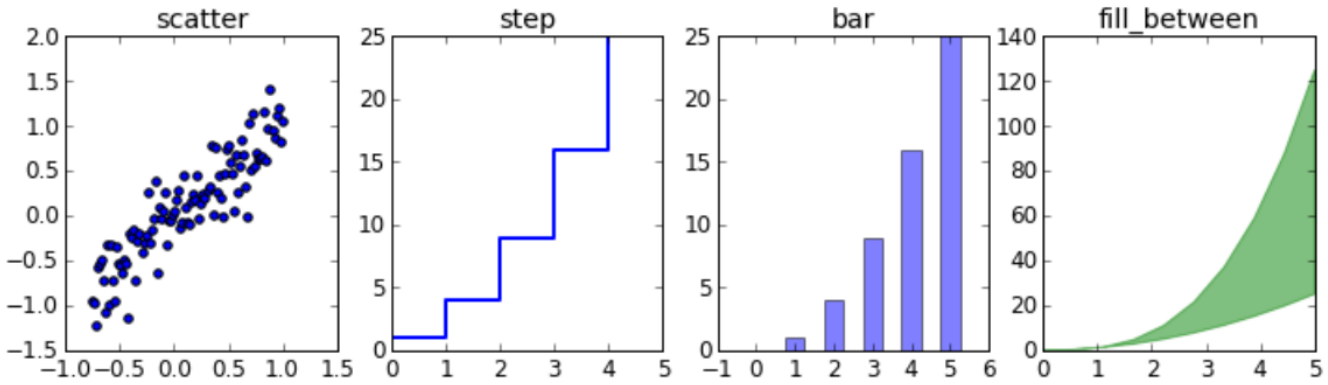
  %matplotlib inline

Matplotlib: Exploratory Data Analysis

- Some 2D plots:

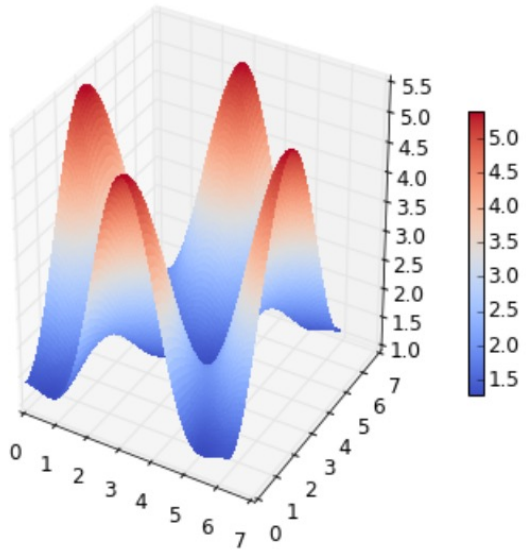
```
n = np.array([0,1,2,3,4,5])
```

```
fig, axes = plt.subplots(1, 4, figsize=(12,3))

axes[0].scatter(xx, xx + 0.25*np.random.randn(len(xx)))
axes[0].set_title("scatter")

axes[1].step(n, n**2, lw=2)
axes[1].set_title("step")

axes[2].bar(n, n**2, align="center", width=0.5, alpha=0.5)
axes[2].set_title("bar")

axes[3].fill_between(x, x**2, x**3, color="green", alpha=0.5);
axes[3].set_title("fill_between");
```

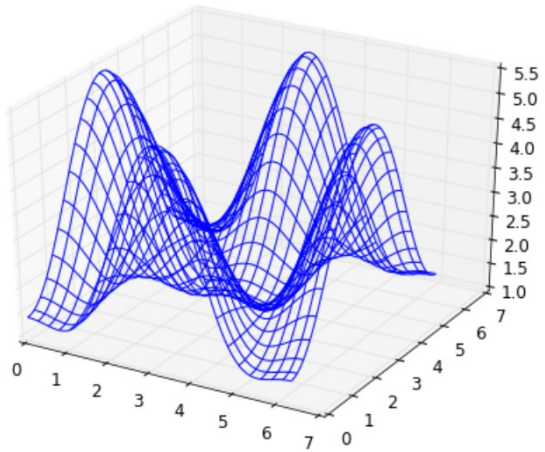


Matplotlib: Exploratory Data Analysis

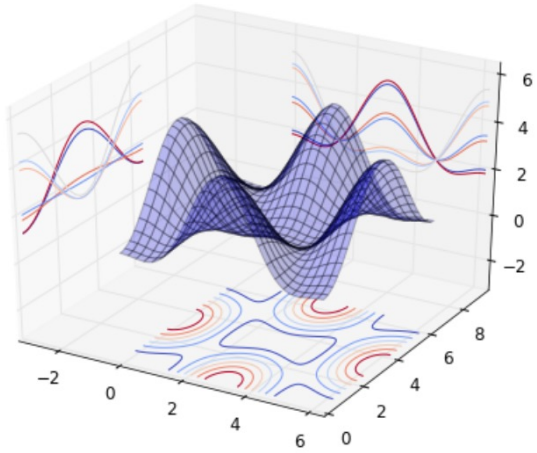
- 3D plots: sometimes we need to analyze data in 3D, but don't overuse it



Surface plot



Wire-frame plot



Contour plot with projections

Exploratory data analysis with Seaborn

- Office website: https://seaborn.pydata.org/
- Built on top of matplotlib
- Features:
  - Beautiful default styles → less customization effort than Matplotlib
  - Exploratory data analysis
  - Statistical data analysis (next lecture)
  - Designed to work well with Pandas data frame
  - A rich gallery: https://seaborn.pydata.org/
- Installation:
  - conda install seaborn

  or
  - pip install seaborn
- How to use in Jupyter notebook:
  - import seaborn as sns
  - %matplotlib inline

Seaborn: Data Distribution

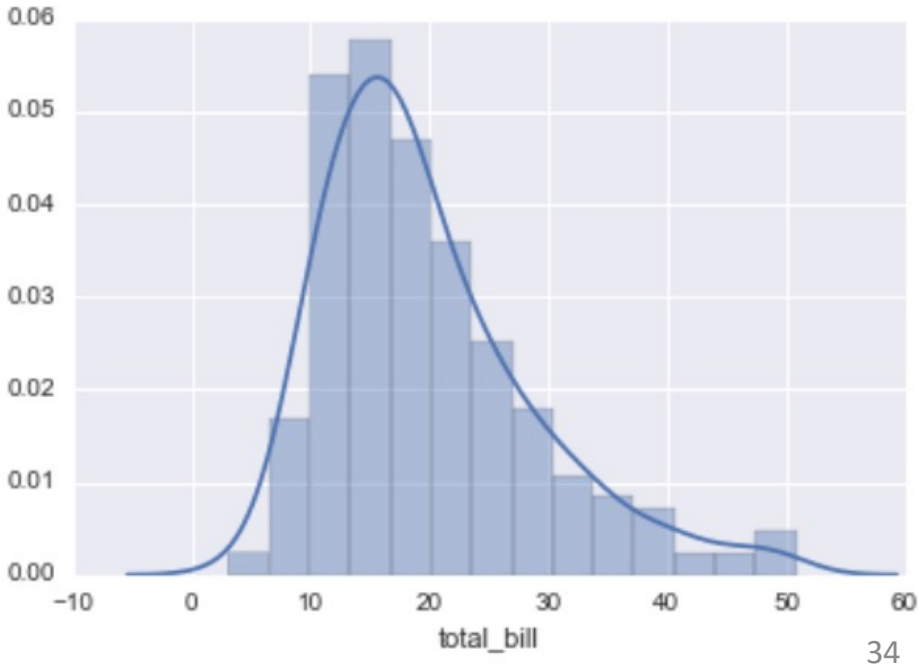
- Histogram for a single variable

```
tips = sns.load_dataset('tips')
```

```
tips.head()
```

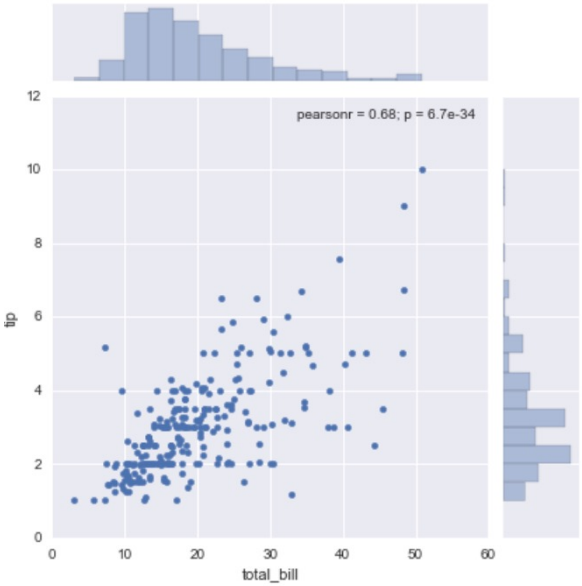
	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4

```
sns.distplot(tips['total_bill'])
```

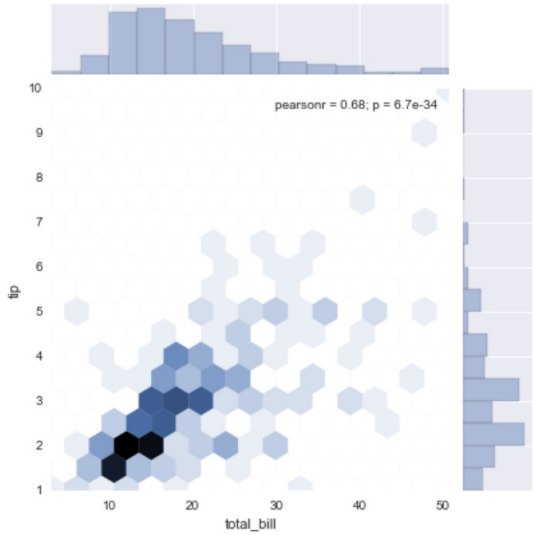


Seaborn: Data Relationship

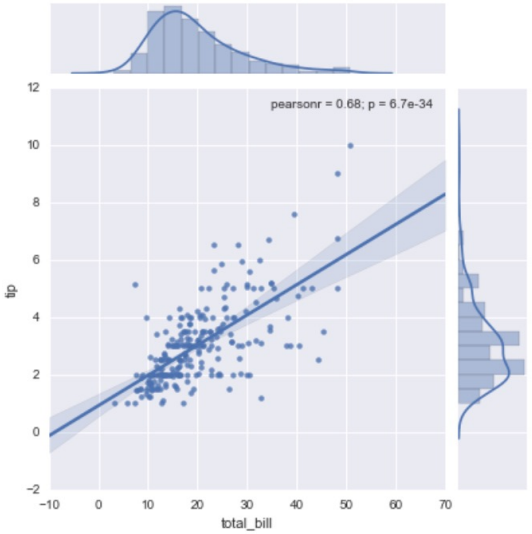
- Jointplot do distribution and correlation at once
  - sns.jointplot(x='total_bill',y='tip',data=tips,kind='scatter')
  - sns.jointplot(x='total_bill',y='tip',data=tips,kind='hex')
  - sns.jointplot(x='total_bill',y='tip',data=tips,kind='reg')



With scatter



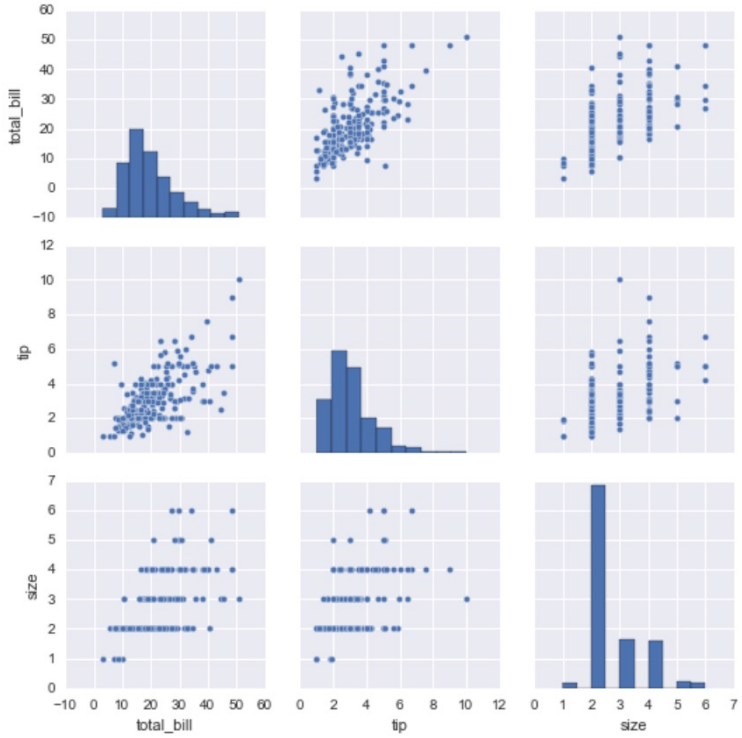
With hexagon shape



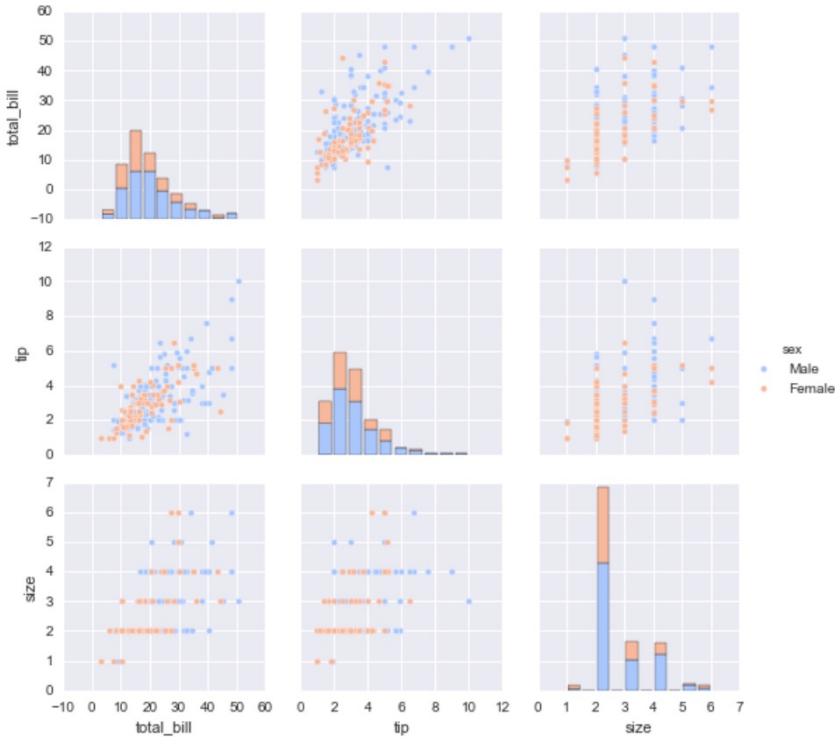
With linear regression

Seaborn: Data Relationship

- Pairplot automatically analyze pairwise relationships



sns.pairplot(tips)



sns.pairplot(tips,hue='sex',palette='coolwarm')

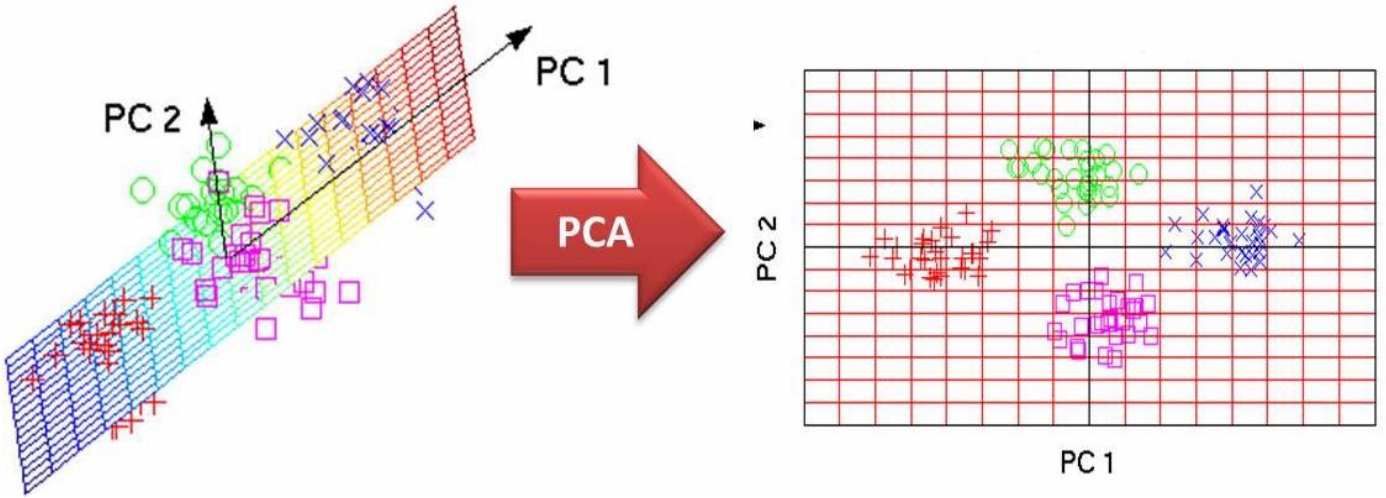
III. Dimensionality Reduction

Visualize High-Dimensional Data

- Projection of high-dimensional data onto smaller dimensions
- Why?: The curse of dimensionality
  - Hard to visualize
  - Hard to analyze since high-dimensional data points are far from each other
  - Computational expensive
  - → Dimensionality reduction: distill higher-dimensional data down to a smaller number of dimensions, while preserving as much of the variance in the data as possible.
    - Good for visualization/compression
    - Good for feature extraction
- Techniques:
  - Feature selection: $\{d_1, d_2, d_3, d_4, d_5\} \rightarrow \{\mathrm{d}_1, \mathrm{d}_3, \mathrm{d}_4\}$
  - Feature reduction: $\{d_1, d_2, d_3, d_4, d_5\} \rightarrow \{\mathrm{d'}_1, \mathrm{d'}_2, \mathrm{d'}_3\}$

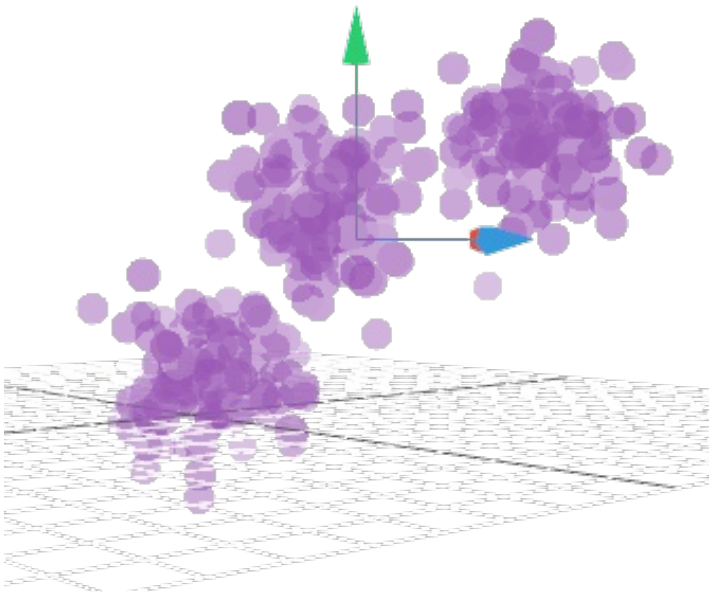
Principal Component Analysis (PCA)

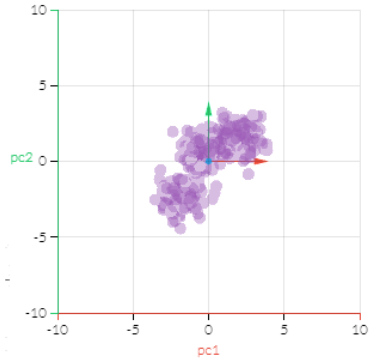
- What:
  - Allows visualization of high-dimensional continuous data in 2-3D
  - The principal components are the strongest (highest variation) dimensions in the dataset, and are orthogonal
  - Really useful for things like image compression and facial recognition



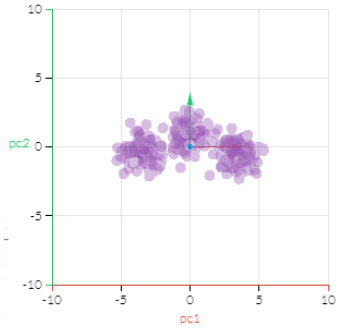
PCA: Another Example

- http://setosa.io/ev/principal-component-analysis/





Bad feature reduction

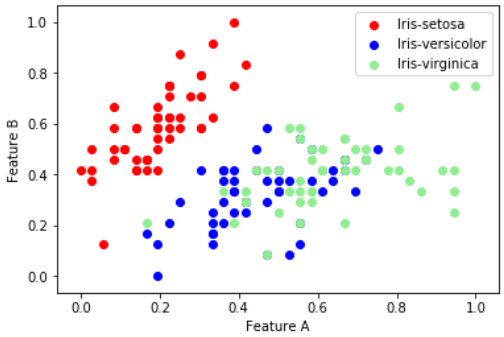


Good feature reduction

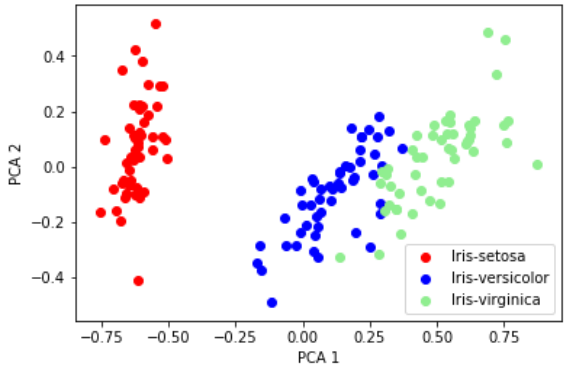
PCA using Scikit-learn

Visualize 4-D Iris Flower Data in 2-D

- Naïve approach: using 2 random dimensions
- PCA: use 2 dimensions, while still preserving variance.



Cannot see a clear separation between different flower classes



A better separation between different flower classes

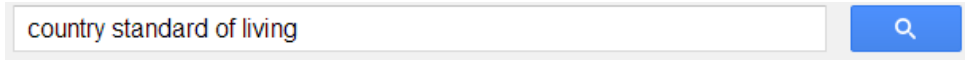
```
import matplotlib.pyplot as plt
import pandas as pd

from sklearn.decomposition import PCA as sklearnPCA
```

```
pca = sklearnPCA(n_components=2) #2-dimensional PCA
transformed = pd.DataFrame(pca.fit_transform(X_norm))
```

IV. Data Sampling (OPTIONAL)

Sampling

- Show only a subset of data
- Sampling requirements:
  - Small enough for user cognitive load
  - Reflect the properties of original data
- **Application**: Web Table searching



List of **countries** by Human Development Index - Wikipedia, the free ...
en.wikipedia.org/.../List_of_**countries**_by_Human_Devel... ▾ Dịch trang này
The Human Development Index (HDI) is a comparative measure of life expectancy, literacy, education, **standards of living**, and quality of life for **countries** ...
Methodology - Complete list of countries - List of countries by continent

Standard of living - Wikipedia, the free encyclopedia
en.wikipedia.org/wiki/**Standard_of_living** ▾ Dịch trang này
Standard of living refers to the level of wealth, comfort, material goods and ... As an example, **countries** with a very small, very rich upper class and a very large, ...

Cost of Living Index by **Country** 2014
www.numbeo.com › Numbeo › Cost of Living ▾ Dịch trang này
By **Country** : **Cost of Living** Index, Consumer Price Index, Restaurant Prices Index, Transportation Price Index, Grocery Price Index, Local Purchasing Power ...

Cost of Living Comparison Between Two **Countries**
www.numbeo.com › Numbeo › Cost of Living ▾ Dịch trang này
Select section ---, **Cost of Living** Comparison, Crime Comparison, Health Care ... **Cost of Living** Comparison Between Two **Countries**. Tweet. Select first **country**.

Searching web pages

title → List of **countries** by inequality-adjusted HDI - Wikipedia, the ...
hyperlink → http://en.wikipedia.org/wiki/List_of_**countries**_by_inequality-adjusted_HDI

Country | Norway | Australia | Sweden

Show less (135 rows / 5 columns total) - Import data

A sampling of full web table

Country	IHDI	HDI	Loss	Rank change
Italy	0.779	0.874	10.9	−2
United States	0.771	0.910	15.3	−19
Jamaica	0.610	0.727	16.2	4
Rep. of Macedonia	0.609	0.728	16.4	2
India	0.392	0.547	28.3	1
Fed. Sts. of Micronesia	0.390	0.636	38.6	−12
Ghana	0.367	0.541	32.2	−1
Rep. of the Congo	0.367	0.533	31.1	-1
Niger	0.195	0.295	34.2	0
Dem. Rep. of the	0.172	0.286	39.9	0

List of **countries** by inequality-adjusted HDI - Wikipedia, the ...
http://en.wikipedia.org/wiki/List_of_**countries**_by_inequality-adjusted_HDI

Country | Norway | Australia | Sweden

Show more (133 rows / 5 columns total) - Import data

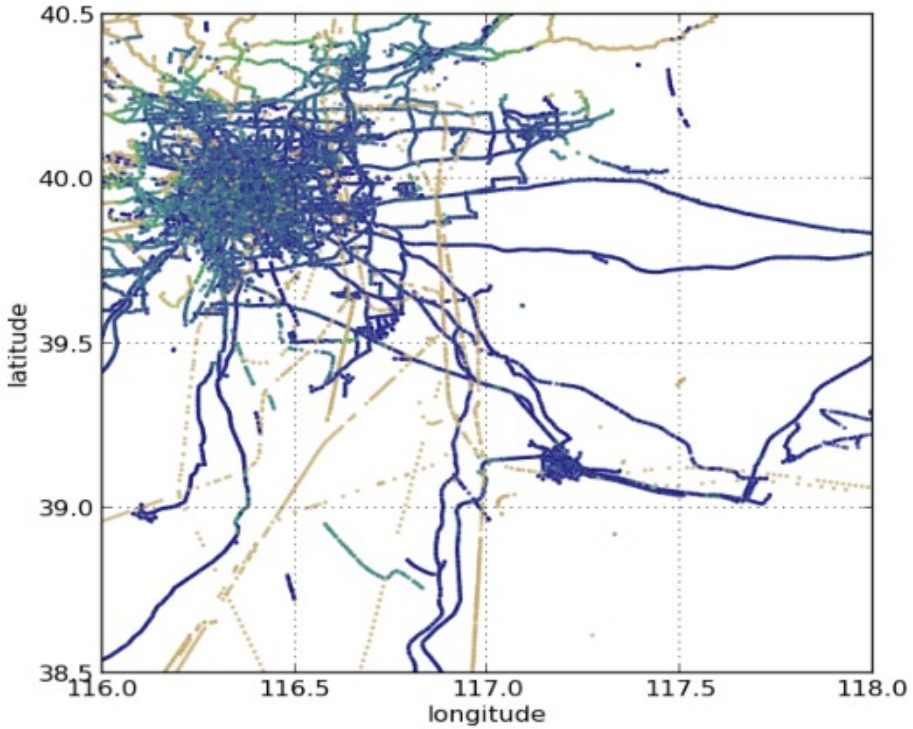
List of **countries** by GDP (PPP) per capita - Wikipedia, the free ...
http://en.wikipedia.org/wiki/List_of_**countries**_by_GDP_(PPP)_per_capita

Country | Qatar | Luxembourg | Singapore

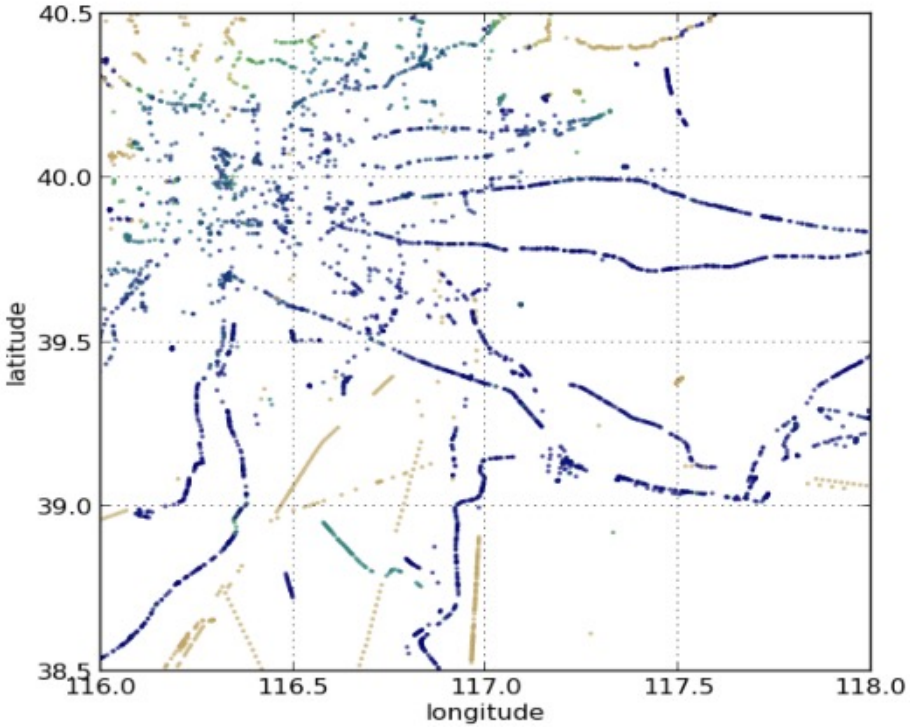
Show more (191 rows / 4 columns total) - Import data

Searching web tables

Sampling: another application



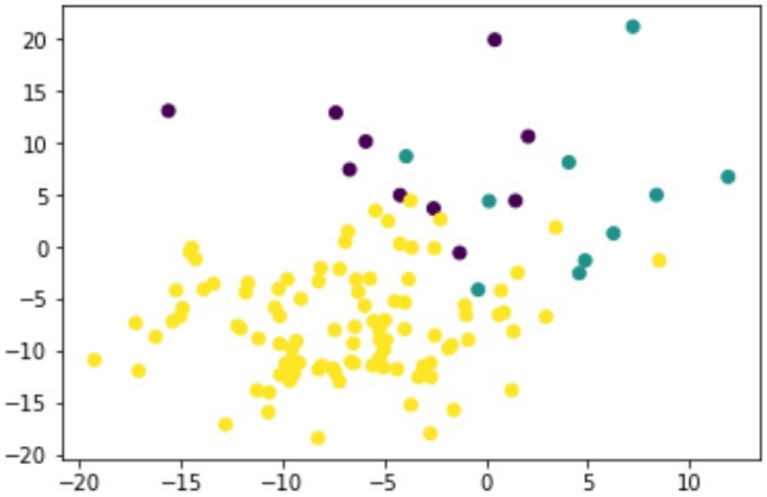
original data



a good sample

Sampling Techniques

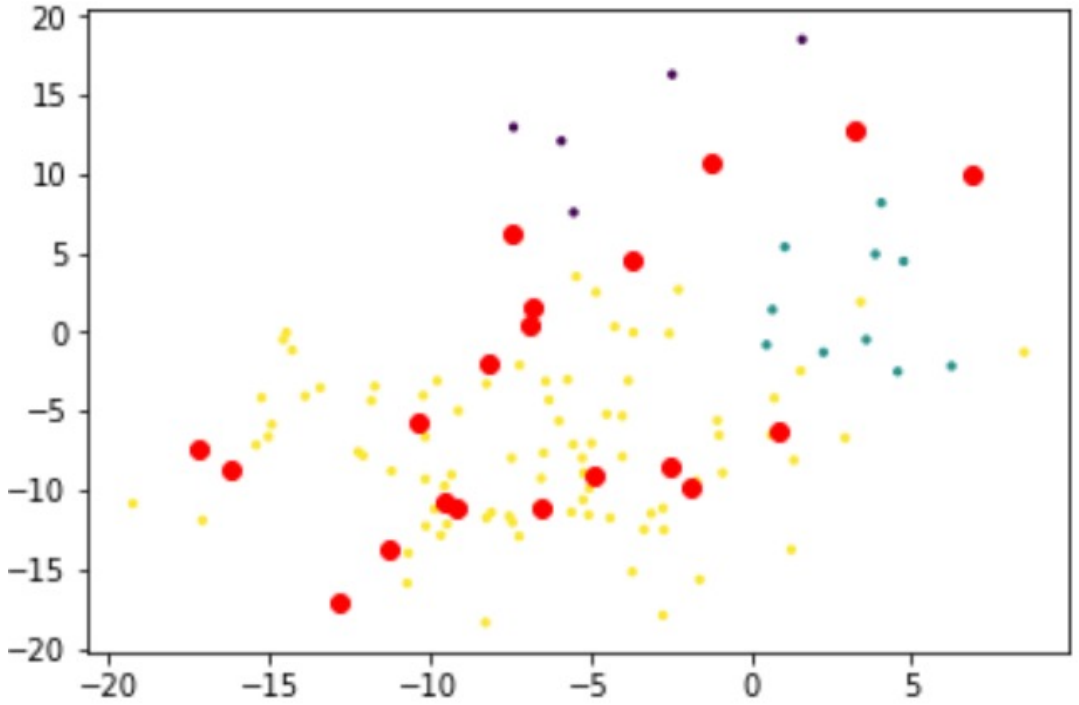
- **Pre-processing**: transform data items to d-dimensional points
  - Define features or reuse attributes
  - Consider each feature/attribute a one dimension
  - Each item is represented by a vector of its feature/attribute values
- **Goal of sampling:**
  - Select k out of n points (k <<n)



Example: a labeled dataset from scikit-learn

Simple Random Sampling (SRS)

- The probability of selecting every data item is uniform
- Algorithm?:
  - **Input**: a set of original data points D, |D|=n
  - **Output**: a set of samples $S \subseteq D$, with $|S| = k$

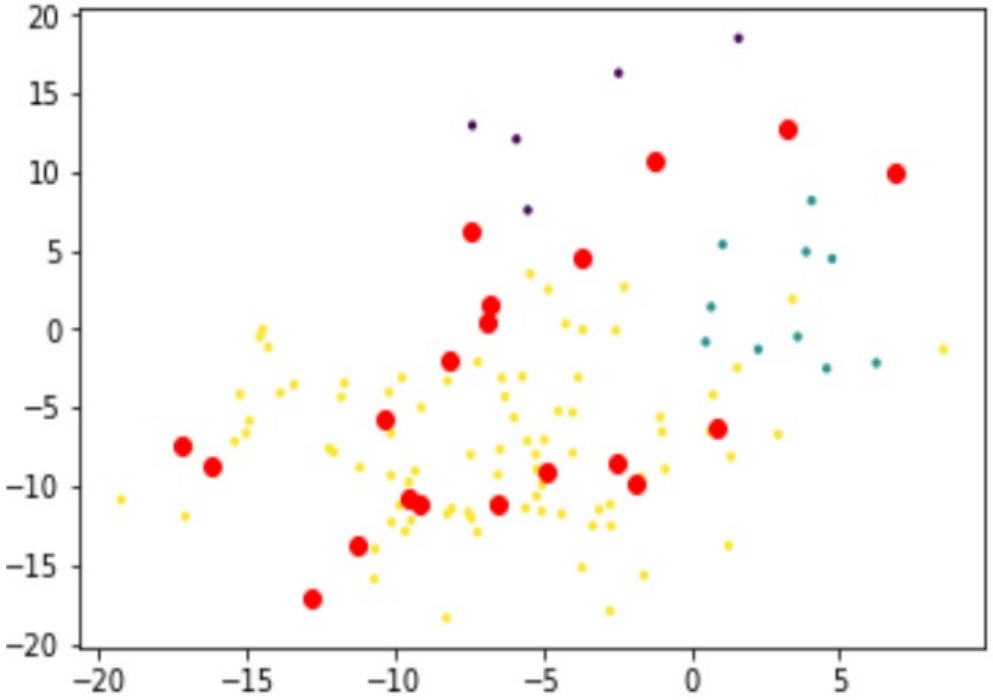


Simple Random Sampling (SRS)

- The probability of selecting every data item is uniform
- An algorithm for generate SRS:
  - **Input**: a set of original data points D, |D|=n
  - **Output**: a set of samples $S \subseteq D$, with $|S| = k$
  1. Generate a random permutation
     - for i from 0 to n−2 do
       - j ← random integer such that i ≤ j < n
       - exchange D[i] and D[j]
  2. Return the first k-elements: $S = D[0:k]$
- **A Python implementation**: S = numpy.random.permutation(D)[:k]

Simple Random Sampling (SRS)

- **Properties:**
  - Most of the sampling points will fall into big clusters
  - In extreme cases, small clusters will have no sampling points

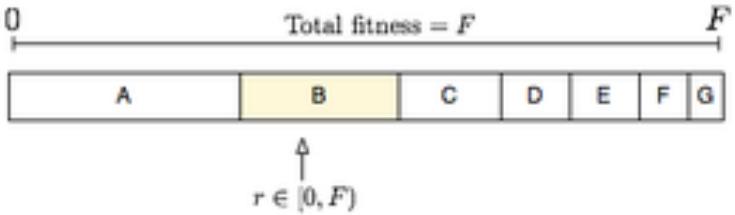


Weighted Random Sampling

- Data items are weighted and the probability of selecting each item is determined by its relative weight.
- Algorithm?
  - **Input**: a set D of n weighted items, each $d_i \in D$ is associated with $w_i \geq 0$
  - **Output**: a set of samples $S \subseteq D$, with $|S| = k$

Weighted Random Sampling

- Data items are weighted and the probability of selecting each item is determined by its relative weight.
- Algorithm:
  - **Input**: a set D of n weighted items
  - **Output**: a set of samples $S \subseteq D$, with $|S| = k$
  1. For $\mathrm{r} = 1$ to k do
     - Update $p_i = \frac{w_i}{\sum_{d_j \in D \setminus S} w_j}$ be the probability of item $d_i \in D \setminus S$ to be selected in round r
     - Randomly select an item $d_i \in D \setminus S$ by Roulette-wheel selection [*]
     - Insert it into S
  2. Return S



[*] https://en.wikipedia.org/wiki/Fitness_proportionate_selection

Weighted Random Sampling

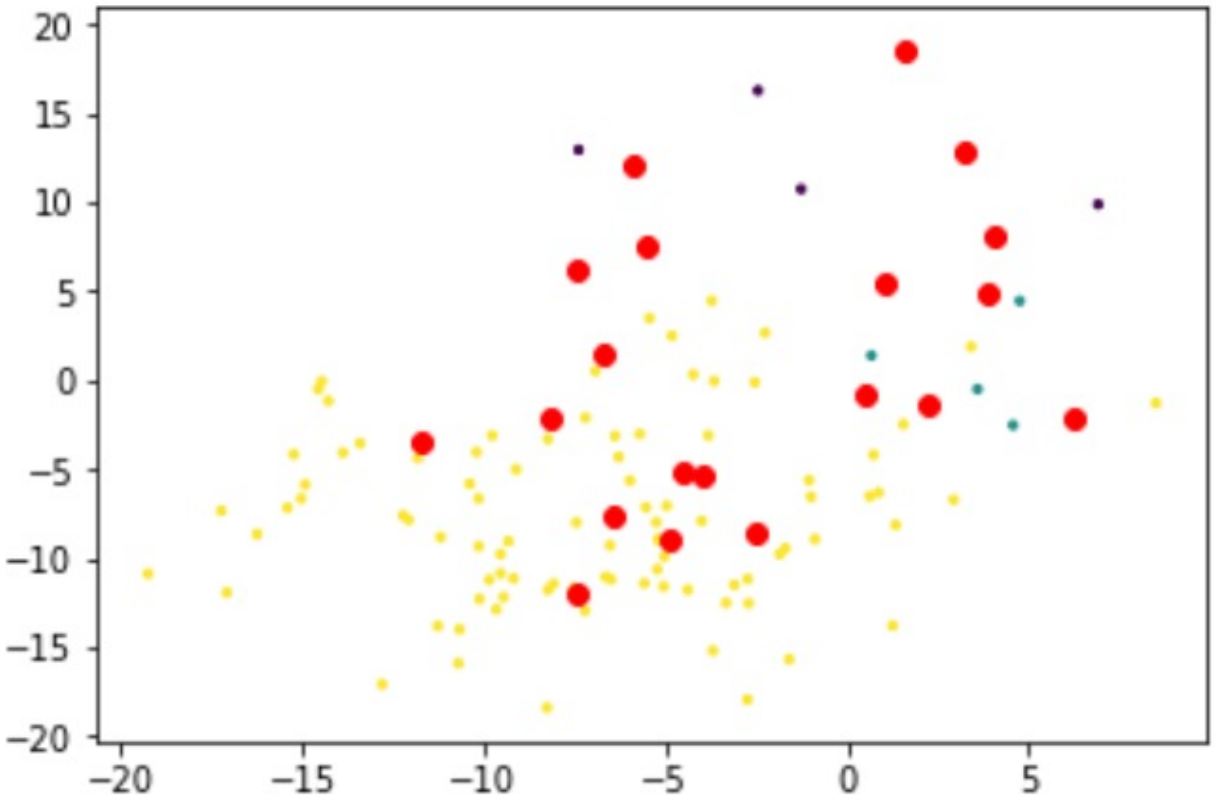
- **Pros**: improve Simple Random Sampling by putting the weights for small clusters
- **Cons**: you have to assign weights for each data point yourself

Stratified Sampling

- **Definition**:
  - A stratified random sample is essentially a series of SRSs performed on subgroups of a given population.
  - The SRS taken within each group in a stratified random sample need not be of the same size
- **General procedure:**
  - Divide data points to group (if not available before-hand)
  - Sample for each group by random sampling
- **Pseudo-code**:
  - sample = SRS(partition1, k_1) ∪ SRS(partition2, k_2) ∪ …

Stratified Sampling

- **Properties:**
  - Uniform distribution across different groups (if we set stratified sample of the same size)



Big Data Visualization: Variety

- Data comes from different sources and needs to be filtered properly
- Application: Facebook News Feed
  1. **Novelty**: show new information
  2. **Relevance**: show feeds from close friends
  3. **Diversity**:
     - Showing most recent feeds is redundant for strangers
     - Showing all birthdate congratulations for a single friend is redundant

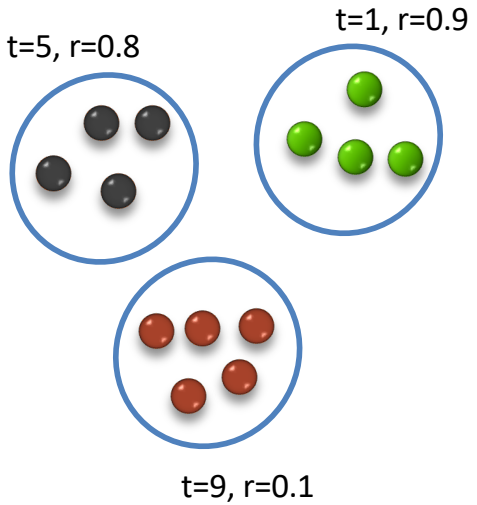
https://blog.bufferapp.com/facebook-news-feed-algorithm

Variety: Techniques

- Formulated as an **optimization problem**:
  - Each post is represented as a data point
  - Each data point has a relevance degree to user interest
  - Each data point is associated by a time index.
  - A similarity function for a pair of posts
- **Objective function**: relevance + novelty – similarity

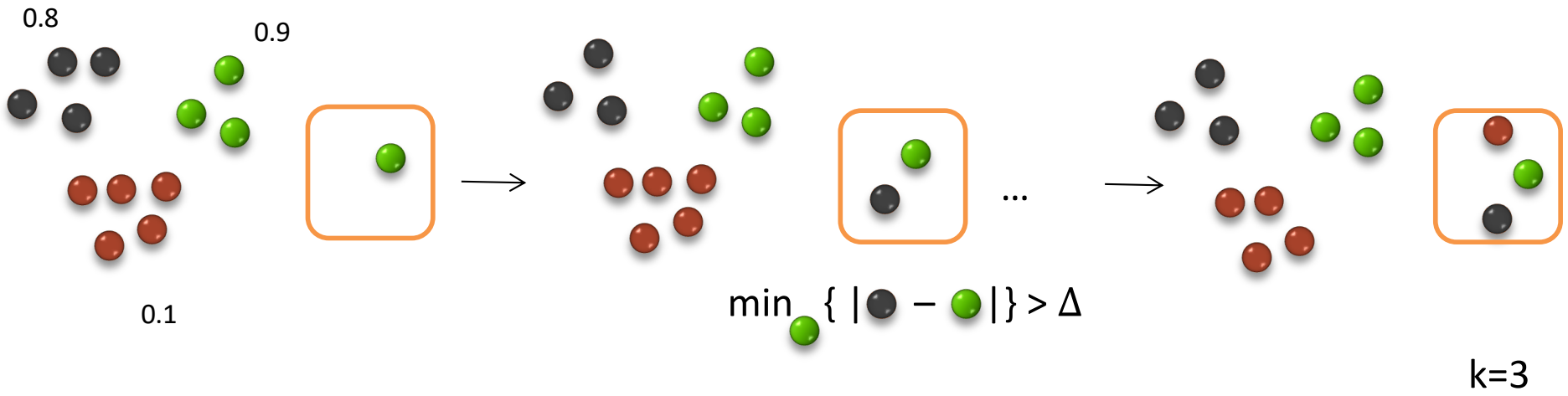
Multi-objective problem

- **Goal:**
  - Select k data items to create an output set
  - Maximize criteria simultaneously:
    - **Novelty**: the more recent the better
    - **Relevance**: relevance score of each data item for a given query
    - **Diversity**: dissimilarity between data items



A diversification algorithm

- Motley [15]: constructs the output by incrementally adding items in the decreasing order of relevance and maximizing the minimum dissimilarity.
  1. Traverse items in the decreasing order of relevance
  2. Add an item to output if the minimum dissimilarity with other selected items is larger than a threshold Δ



Facebook News Feed Algorithm

- Design principles:
  - **Friend and family come first:** The main objective of the News Feed is to connect people with their friends and family. So posts from friends and family are prioritized. After those posts, Facebook found that people want their feed to inform and entertain them.
  - **A platform for all ideas:** Facebook welcomes all ideas while making sure that everyone feels and is safe. They aim to deliver stories that each individual wants to see the most, based on their actions and feedback.
  - **Authentic communications:** Facebook prioritizes genuine stories over misleading, sensational, and spammy ones.
  - **You control your experience:** Individuals know themselves best. So Facebook creates features (such as unfollow and see first) to let people customize their Facebook experience.

https://blog.bufferapp.com/facebook-news-feed-algorithm

Big Data Visualization: Velocity

- **Data** is big but contains many low-valued items: redundant, overlapping, sparse
- Data comes in stream and with high speed
  - E.g. social media, Internet of Things (IoT)
- Data monitoring systems have limited storage

→ Need to effectively decide which old data to replace

References

[1] https://www.slideshare.net/AshwiniKuntamukkala/data-wrangling-62017599
[2] https://www.slideshare.net/hhamalai/python-for-data-science
[3] https://www.slideshare.net/hassass15/data-cleaning-and-screening-58372141
[4] https://machinelearningmastery.com/handle-missing-data-python/
[5] https://www.slideshare.net/AshwiniKuntamukkala/data-wrangling-62017599
[6] https://www.slideshare.net/hhamalai/python-for-data-science
[7] https://www.slideshare.net/hafidztio/resampling-methods
[8] https://www.slideshare.net/AjinkyaMore3/python-resampling
[9] http://kindsonthegenius.blogspot.com.au/2017/12/dimensionality-reduction-and-principal.html
[10] https://www.udemy.com/data-science-and-machine-learning-with-python-hands-on
[11] https://www.youtube.com/watch?v=NWldkpR2sGA
[12] http://scikit-learn.org/stable/auto_examples/bicluster/plot_spectral_coclustering.html
[13] https://www.kaggle.com/ekami66/detailed-exploratory-data-analysis-with-python
[14] https://www.slideshare.net/ajandne/pearson-correlation
[15] https://www.slideshare.net/AnishMaman/correlation-36754774
[16] https://www.slideshare.net/EricMarsden1/modelling-correlations-using-python
[17] https://lagunita.stanford.edu/courses/OLI/StatReasoning/Open/about
[18] https://www.slideshare.net/killver/modeling-and-mining-sequential-data
[19] https://www.slideshare.net/hnly228078/spectral-clustering-tutorial
[20] https://www.slideshare.net/AllenWu/information-theoretic-co-clustering
[21] https://www.udemy.com/python-for-data-science-and-machine-learning-bootcamp/

References (cont'd)

[22] Benzi Kirell Mael. Data Visualization, Autumn 2017. http://edu.epfl.ch/coursebook/en/data-visualization-COM-480

[23] West Robert. Applied Data Analysis, Autumn 2017. http://edu.epfl.ch/coursebook/en/applied-data-analysis-CS-401

[24] T. Munzner, Visualization Analysis and Design, 2014.

[25] Jacques Bertin, Semilogy of Graphics, 1967.

[26] Heer and Bostock. Crowdsourcing Graphical Perception: Using Mechanical Turk to Assess Visualization Design. 2010

[27] https://www.youtube.com/watch?v=xAoljeRJ3lU&feature=youtu.be

[28] Good Enough to Great: A Quick Guide for Better Data Visualizations

[29] The Power of "Where"

[30] Visual Analysis Best Practices

[31] Visual Analysis for Everyone

[32] 6 Best Practices for Creating Effective Dashboards

[33] The Power of R and Visual Analytics

[34] Lei Yu, Jieping Ye, Huan Liu. Dimensionality Reduction for Data Mining: Techniques, Applications and Trends. 2007.

[35] Yu, Cong, Laks Lakshmanan, and Sihem Amer-Yahia. "It takes variety to make a world: diversification in recommender systems." *Proceedings of the 12th international conference on extending database technology: Advances in database technology*. ACM, 2009.

[36] Jain, Anoop, Parag Sarda, and Jayant R. Haritsa. "Providing diversity in k-nearest neighbor query results." *PAKDD*. Vol. 4. 2004.

[37] Glyph-based visualization http://vis.cs.ucdavis.edu/vis2014papers/VIS_Conference/tutorials/Glyph-based_Visualization/Glyph-Tutorial-vis2014.pdf