Data Science With Python

Mosky

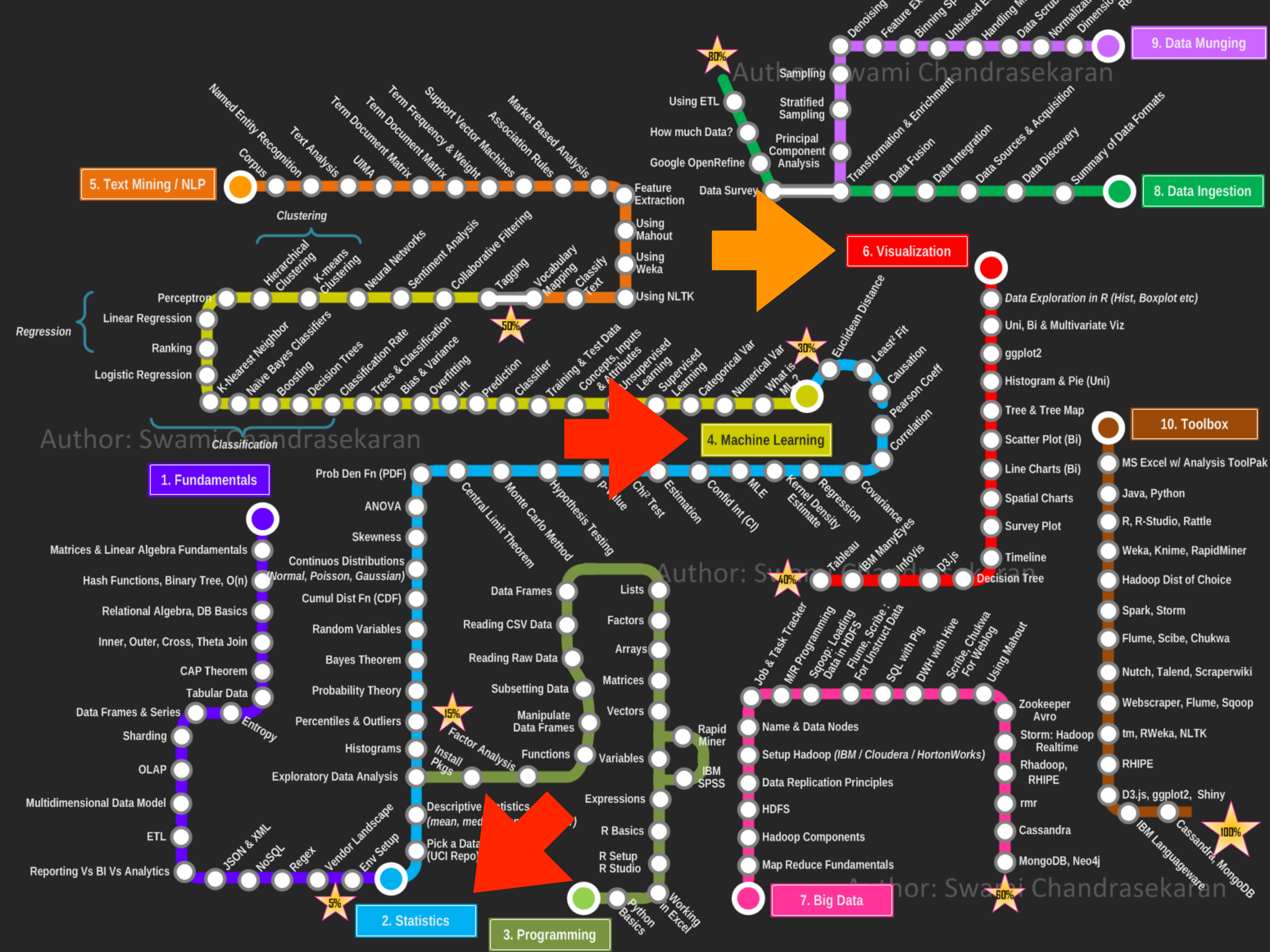
Data Science

- = Extract knowledge or insights from data.
- Data science includes:
  - Visualization
  - Statistics
  - Machine learning
  - Deep learning
  - Big data
  - And related methods
- ≈ Data mining

Data Science

- = Extract knowledge or insights from data.
- Data science includes:
  - Visualization
  - Statistics
  - Machine learning
  - Deep learning
  - Big data
  - And related methods
- ≈ Data mining

Will be introduced.



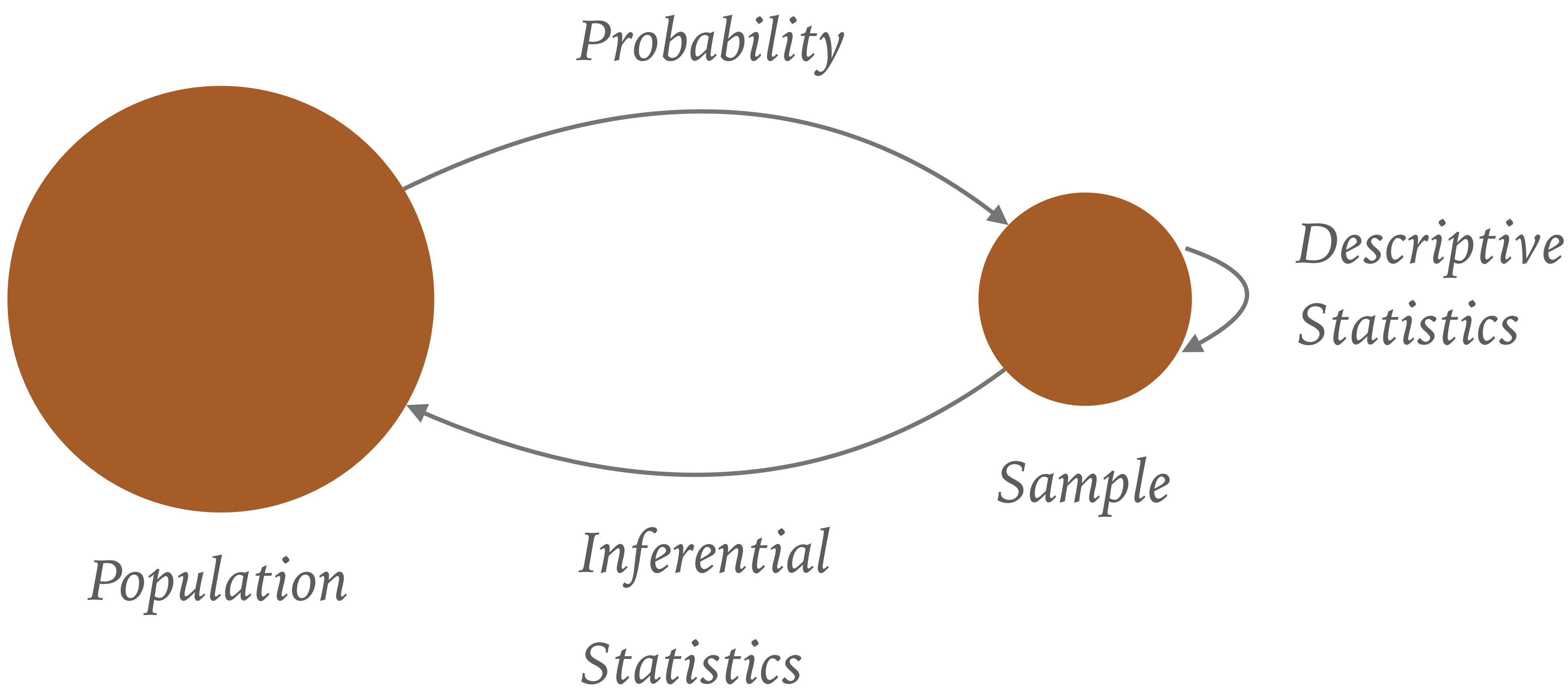
1. Fundamentals
Matrices & Linear Algebra Fundamentals
Hash Functions, Binary Tree, O(n)
Relational Algebra, DB Basics
Inner, Outer, Cross, Theta Join
CAP Theorem
Tabular Data
Entropy
Data Frames & Series
Sharding
OLAP
Multidimensional Data Model
ETL
Reporting Vs BI Vs Analytics
JSON & XML
NoSQL
Regex
Vendor Landscape
Env Setup
5%
2. Statistics
Pick a Data Set (UCI Repo)
Descriptive Statistics (mean, median...)
Exploratory Data Analysis
Histograms
Percentiles & Outliers
Probability Theory
Bayes Theorem
Random Variables
Cumul Dist Fn (CDF)
Continuos Distributions (Normal, Poisson, Gaussian)
Skewness
ANOVA
Prob Den Fn (PDF)
Central Limit Theorem
Monte Carlo Method
Hypothesis Testing
p-Value
Chi² Test
Estimation
Confid Int (CI)
MLE
Kernel Density Estimate
Regression
Covariance
Correlation
Pearson Coeff
Causation
Least² Fit
Euclidean Distance
15%
Factor Analysis
3. Programming
Python Basics
Working in Excel
R Setup R Studio
R Basics
Expressions
Variables
IBM SPSS
Rapid Miner
Vectors
Matrices
Arrays
Factors
Lists
Data Frames
Reading CSV Data
Reading Raw Data
Subsetting Data
Manipulate Data Frames
Functions
Install Pkgs
4. Machine Learning
What is ML ?
Numerical Var
Categorical Var
Supervised Learning
Unsupervised Learning
Concepts, Inputs & Attributes
Training & Test Data
Classifier
Prediction
Lift
Overfitting
Bias & Variance
Trees & Classification
Classification Rate
Decision Trees
Boosting
Naïve Bayes Classifiers
K-Nearest Neighbor
Classification
Regression
Logistic Regression
Ranking
Linear Regression
Perceptron
Clustering
Hierarchical Clustering
K-means Clustering
Neural Networks
Sentiment Analysis
Collaborative Filtering
Tagging
30%
50%
5. Text Mining / NLP
Corpus
Named Entity Recognition
Text Analysis
UIMA
Term Document Matrix
Term Frequency & Weight
Support Vector Machines
Association Rules
Market Based Analysis
Feature Extraction
Using Mahout
Using Weka
Using NLTK
Classify Text
Vocabulary Mapping
6. Visualization
Data Exploration in R (Hist, Boxplot etc)
Uni, Bi & Multivariate Viz
ggplot2
Histogram & Pie (Uni)
Tree & Tree Map
Scatter Plot (Bi)
Line Charts (Bi)
Spatial Charts
Survey Plot
Timeline
Decision Tree
D3.js
InfoVis
IBM ManyEyes
Tableau
40%
7. Big Data
Map Reduce Fundamentals
Hadoop Components
HDFS
Data Replication Principles
Setup Hadoop (IBM / Cloudera / HortonWorks)
Name & Data Nodes
Job & Task Tracker
M/R Programming
Sqoop: Loading Data in HDFS
Flume, Scribe: For Unstruct Data
SQL with Pig
DWH with Hive
Scribe, Chukwa For Weblog
Using Mahout
Zookeeper Avro
Storm: Hadoop Realtime
Rhadoop, RHIPE
rmr
Cassandra
MongoDB, Neo4j
60%
8. Data Ingestion
Summary of Data Formats
Data Discovery
Data Sources & Acquisition
Data Integration
Data Fusion
Transformation & Enrichment
Data Survey
Google OpenRefine
How much Data?
Using ETL
80%
9. Data Munging
Principal Component Analysis
Stratified Sampling
Sampling
Denoising
10. Toolbox
MS Excel w/ Analysis ToolPak
Java, Python
R, R-Studio, Rattle
Weka, Knime, RapidMiner
Hadoop Dist of Choice
Spark, Storm
Flume, Scibe, Chukwa
Nutch, Talend, Scraperwiki
Webscraper, Flume, Sqoop
tm, RWeka, NLTK
RHIPE
D3.js, ggplot2, Shiny
IBM Languageware
Cassandra, MongoDB
100%
Author: Swami Chandrasekaran

- It's kind of outdated, but still contains lot of keywords.
- MrMimic/data-scientist-roadmap – GitHub
- Becoming a Data Scientist – Curriculum via Metromap

Statistics vs. Machine Learning

- Machine learning = statistics - checking of assumptions 😆
- But does resolve more problems. 👍🏻
- Statistics constructs more solid inferences.
- Machine learning constructs more interesting predictions.

Probability, Descriptive Statistics, and Inferential Statistics



Machine Learning vs. Deep Learning

- Deep learning is the most renowned part of machine learning.
  - A.k.a. the “AI”.
- Deep learning uses artificial neural networks (NNs).
- Which are especially good at:
  - Computer vision (CV) 👀
  - Natural language processing (NLP) 📖
    - Machine translation
  - Speech recognition
- Too costly to simple problems.

Big Data

- The “size” is constantly moving.
  - As of 2012, ranges from 10n TB to n PB, which is 100x.
- Has high-3Vs:
  - Volume, amount of data.
  - Velocity, speed of data in and out.
  - Variety, range of data types and sources.
- A practical definition:
  - A single computer can't process in a reasonable time.
  - Distributed computing is a big deal.

Today,

- “Models” are the math models.
- “Statistical models”, emphasize inferences.
- “Machine learning models”, emphasize predictions.
- “Deep learning” and “big data” are gigantic subfields.
  - We won't introduce.
  - But the learning resources are listed at the end.

Mosky

- Python Charmer at Pinkoi.
- Has spoken at
  - PyCons in TW, MY, KR, JP, SG, HK, COSCUPs, and TEDx, etc.
- Countless hours on teaching Python.
- Own the Python packages:
  - ZIPCodeTW, MoSQL, Clime, etc.
- http://mosky.tw/

The Outline

1. “Data”
2. Visualization
3. Preprocessing
4. Dimensionality Reduction
5. Statistical Models
6. Machine Learning Models
7. The Analysis Steps
8. Keep Learning

The Packages

- `$ pip3 install jupyter numpy scipy sympy matplotlib ipython pandas seaborn statsmodels scikit-learn`
- Or
- `> conda install jupyter numpy scipy sympy matplotlib ipython pandas seaborn statsmodels scikit-learn`
- Visit the Pipfile for the exact version numbers.

Common Jupyter Notebook Shortcuts

`Esc`	Edit mode → command mode.
`Ctrl-Enter`	Run the cell.
`B`	Insert cell below.
`D, D`	Delete the current cell.
`M`	To Markdown cell.
`Cmd-/`	Comment the code.
`H`	Show keyboard shortcuts.
`P`	Open the command palette.

Checkpoint: The Packages

- Open *00_preface_the_packages.ipynb* up.
- Run it.
- The notebooks are available on https://github.com/moskytw/data-science-with-python.

"Data"

"Data"

- = Variables
- = Dimensions
- = Label + Features

“Data”

male	height_cm	weight_kg	age
1	152	48	63
1	157	53	41
0	140	37	63
0	137	32	65

Data in Different Types

Discrete	Nominal		{male, female}
	Ordinal Ranked	↑ & can be ordered.	{great > good > fair}
Continuous	Interval	↑ & distance is meaningful.	temperatures
	Ratio	↑ & 0 is meaningful.	weights

Data in X-Y Form

y	x
dependent variable	independent variable
response variable	explanatory variable
regressand	regressor
endogenous variable | endog	exogenous variable | exog
outcome	design
label	feature

Data in X-Y Form

- Confounding variables:
  - May affect y, but not x; can lead erroneous conclusions.
  - Deal with:
    - Controlling, e.g., fix the environment.
    - Randomizing, e.g, choose by computer.
    - Matching, e.g., order by gender and then assign group.
    - Statistical control, e.g., BMI to remove height effect.
    - Double-blind, even triple-blind trials.
  - "Garbage in, garbage out."

Get the Data

- Logs
- Existent datasets
  - The Datasets Package – StatsModels
  - Kaggle
- Experiments

Visualization

Visualization

- Make Data Colorful – Plotting
  - *01_1_visualization_plotting.ipynb*
- In a Statistical Way – Descriptive Statistics
  - *01_2_visualization_descriptive_statistics.ipynb*

Checkpoint: Plot the Variables

- Star98
  - `star98_df = sm.datasets.star98.load_pandas().data`
- Fair
  - `fair_df = sm.datasets.fair.load_pandas().data`
- Howell1
  -
    ```
    howell1_df = pd.read_csv(
        'dataset_howell1.csv', sep=';')
    ```
- Or your own datasets.
- Plot the variables that interest you.

Preprocessing

Feed the Data That Models Like

- Preprocess data for:
  - Hard requirements, e.g.,
    - corpus → vectors
    - "What kind of news will be voted down on PTT?"
  - Soft requirements (hypotheses), e.g.,
    - t-test: better when samples are normally distributed.
    - SVM: better when features range from -1 to 1.
  - More representative features, e.g., total price / units.
- Note that different models have different tastes.

Preprocessing

- The Dishes – Containers
  - *02_1_preprocessing_containers.ipynb*
- A Cooking Method – Standardization
  - *02_2_preprocessing_standardization.ipynb*
- Watch Out for Poisonous Data Points – Removing Outliers
  - *02_3_preprocessing_removing_outliers.ipynb*

“

In data science, 80% of time spent prepare data, 20% of time spent *complain* about need for prepare data.

Checkpoint: Preprocess the Variables

- Pick a dataset.
- Try to standardize and compare.
- Try to trim the outliners.

Dimensionality Reduction

The Model Sicks Up!

- Let's reduce the variables.
- Feed a subset → feature selection.
  - Feature selection using SelectFromModel – Scikit-Learn
- Feed a transformation → feature extraction.
  - PCA, FA, etc.
  - Another definition: non-numbers → numbers.

Dimensionality Reduction

- Principal Component Analysis
  - *03_1_dimensionality_reduction_principal_component_analysis.ipynb*
- Factor Analysis
  - *03_2_dimensionality_reduction_factor_analysis.ipynb*

Checkpoint: Reduce the Variables

- Pick a dataset.
- Try to $PCA(all\ variables) \rightarrow the\ better\ components$, or FA.
- And then plot n-dimensional data onto 2-dimensional plane.

Statistical Models

Statistical Models

- Identify Boring or Interesting – Hypothesis Testings
  - *04_1_statistical_models_hypothesis_testings.ipynb*
  - "Hypothesis Testing With Python"
- Identify X-Y Relationships – Regression
  - *04_2_statistical_models_regression_anova.ipynb*

Checkpoint: Apply a Statistical Model

- Pick a dataset.
- Apply a statistical model to extract knowledge.

Machine Learning Models

Machine Learning Models

- Apple or Orange? – Classification
  - *05_1_machine_learning_models_classification.ipynb*
- Without Labels – Clustering
  - *05_2_machine_learning_models_clustering.ipynb*
- Predict the Values – Regression
- Who Are the Best? – Model Selection
  - sklearn.model_selection.GridSearchCV

Confusion matrix, where $A = 00_2 = C[0, 0]$

	predicted negative AC	predicted positive BD
actual negative AB	true negative A	false positive B
actual positive CD	false negative C	true positive D

Rates in confusion matrix

	=	=	= observed
false positive rate	B/AB		α
false negative rate	C/CD		β
false discovery rate	B/BD		inverse α
false omission rate	C/AC		inverse β
actual negative rate	AB/ABCD		
sensitivity	D/CD	recall	power
specificity	A/AB		confidence level
precision	D/BD		inverse power
recall	D/CD	sensitivity	power

Ensemble Models

- Bagging
  - *N* independent models and then average their output.
  - e.g., the random forest models.
- Boosting
  - *N* sequential models, *n*-th learns from *(n-1)*-th's error.
  - e.g., gradient tree boosting.

Checkpoint: Apply a Machine Learning Model

- Pick a dataset.
- Apply a machine learning model to extract knowledge.

The Analysis Steps

The Three Steps

1. Define Assumptions
2. Validate Assumptions
3. Validated Assumptions?

1. Define Assumptions

- Specify a *feasible* objective.
    - “Use AI to get the moon!”
- Write *formal* assumptions.
    - “The users will buy 1% items from our recommendation.” rather than “The users will love our recommendation!”
- Consider the *next* actions.
    - “Release to 100% of users.” rather than “So great!”

2. Validate Assumption

- Collect *potential* data.
- List *possible* methods.
  - A plotting, median, or even mean may be good enough.
  - Selecting Statistical Tests – Bates College
  - Choosing the right estimator – Scikit-Learn
- Evaluate the metrics of methods with data.
- Note the *dangerous* gaps.
  - "All the items from recommendation are free!"
  - "Correlation does not imply causation."

3. Validated Assumption?

- Yes → Congrats! 🎉
- No → Check:
    - The *hypotheses* of methods.
    - The *confounding variables* in data.
    - The *formality* of assumptions.
    - The *gaps* between assumptions and the objective.
    - The *feasibility* of the objective.
- Always report fully and take the actions.

Focus on Business Impact

1. Choose *low-effort* methods first.
    - Like logistic regression, random forest, etc.
    - Minimize the development time by sampling, better pipeline, etc.
2. Evaluate the business *impact* of the assumptions.
    - Affect internal or external people!
3. Validate assumptions as more as possible.

Checkpoint: Pick a Method

- Think of an interesting problem.
  - E.g., the revenue is higher, but is it because of the noise?
- Pick one method from the cheatsheets.
  - Selecting Statistical Tests – Bates College
  - Choosing a statistical test – HBS
  - Choosing the right estimator – Scikit-Learn

Keep Learning

Keep Learning

- Statistics
  - Seeing Theory
  - Statistics – SciPy Tutorial
  - StatsModels
  - Biological Statistics
  - Research Methods
- Machine Learning
  - Scikit-learn Tutorials
  - Standford CS229
  - Hsuan-Tien Lin
- Deep Learning
  - TensorFlow | PyTorch
  - Standford CS231n
  - Standford CS224n
- Big Data
  - Dask, Spark
  - Superset, Airflow
  - Hive, HBase
  - AWS, GCP

The Facts

- You can't learn *all* the things in the data science!
- ∴
- “Let's learn to do!” 🙅🏻‍♀️
- “Let's do to learn!” 🙆🏻‍♀️

Recap

- Let's *do* to learn.
- What is your *objective*?
- For the objective, what are your assumptions?
- For the assumptions, what methods may validate it?
- For the methods, how will you evaluate it with data?
- Q & A