



# Data Science With Python

---

*Mosky*

# Data Science

---

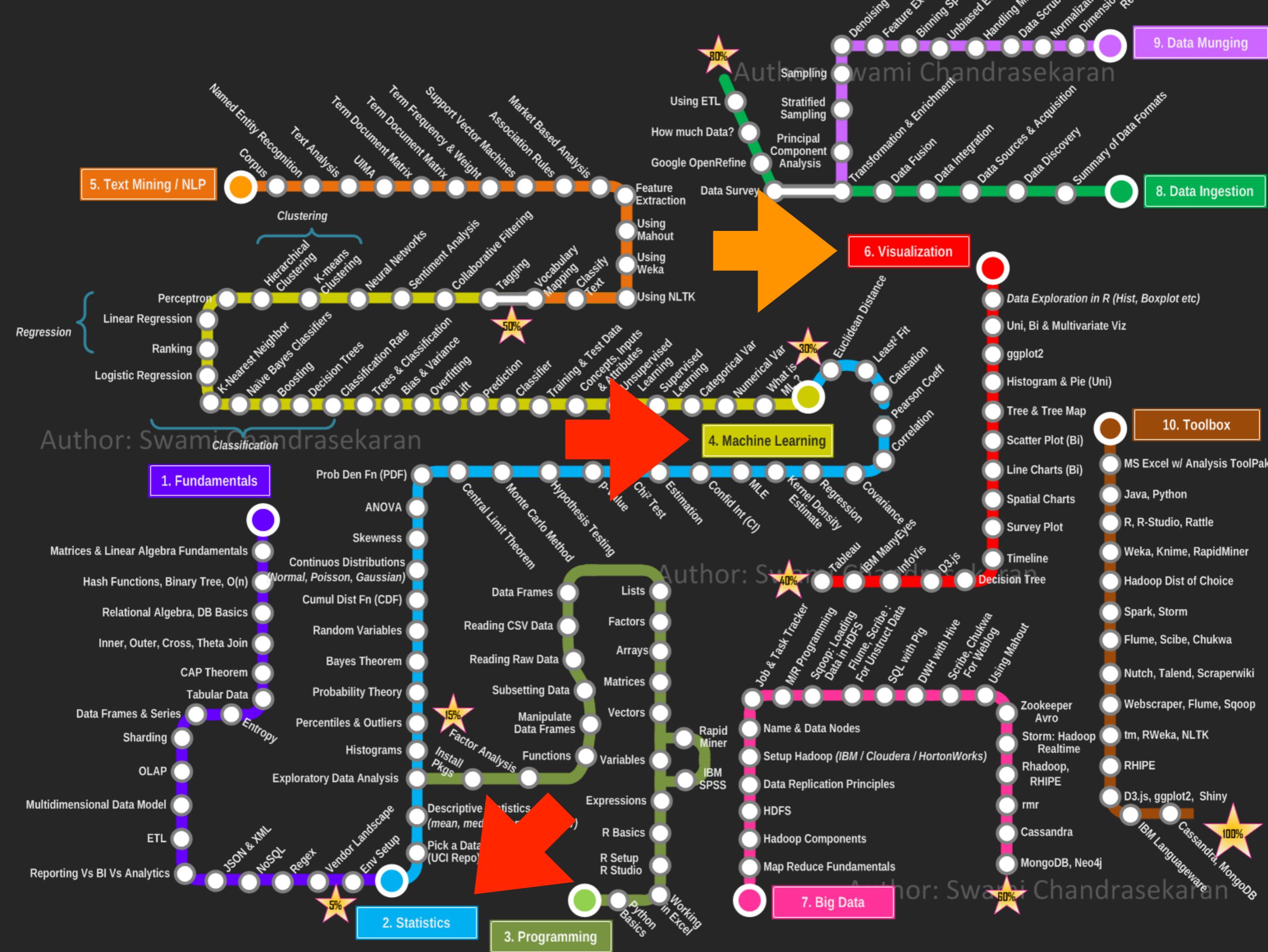
- = Extract knowledge or insights from data.
- Data science includes:
  - Visualization.
  - Statistics.
  - Machine learning.
  - Deep learning.
  - Big data.
  - And their related methods.
- ≈ Data mining.

# Data Science

---

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

We will introduce.



- MrMimic/data-scientist-roadmap – GitHub
- Becoming a Data Scientist – Curriculum via Metromap

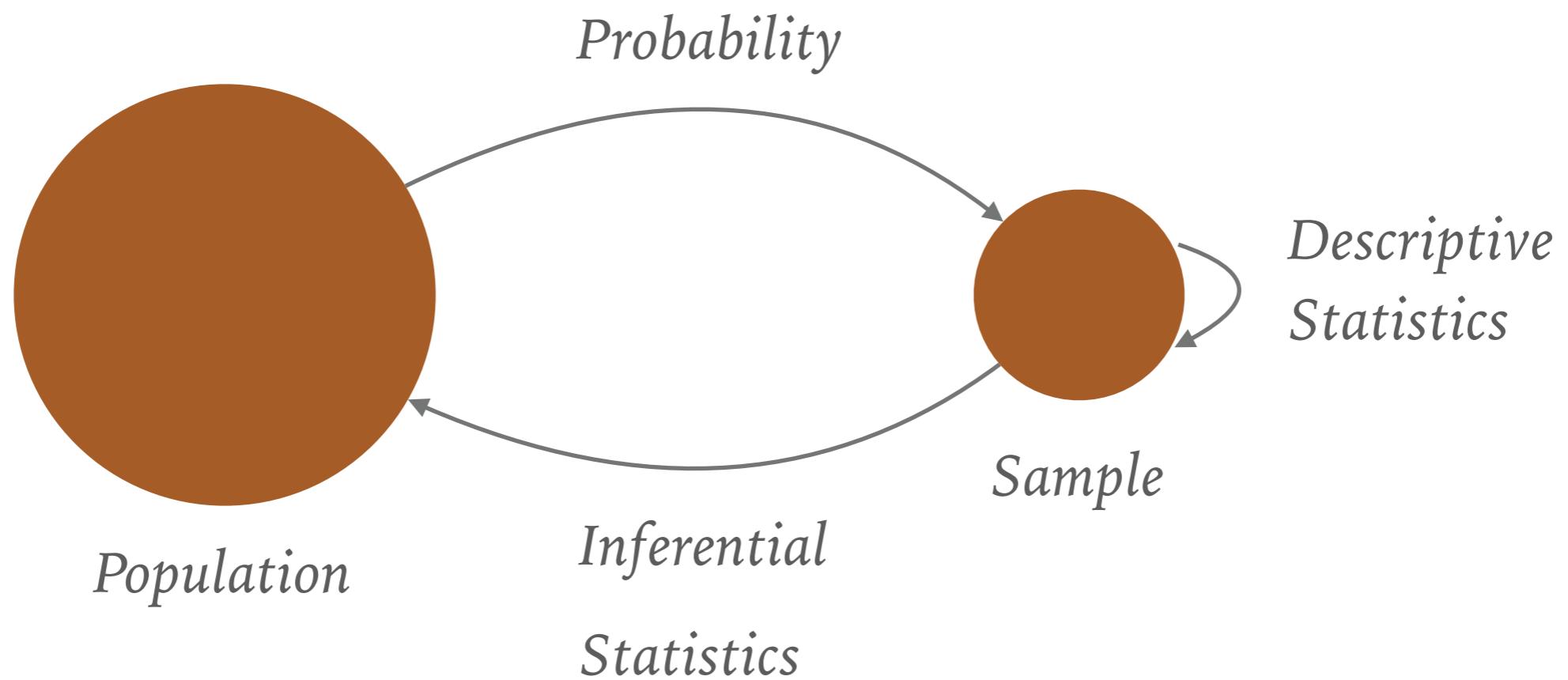
# Statistics vs. Machine Learning

---

- Machine learning = statistics - checking of assumptions 😊
- But does resolve more problems. 👍
- Statistics constructs more solid inferences.
- Machine learning is more result-oriented.

# Probability, Descriptive Statistics, and Inferential Statistics

---



# Machine Learning vs. Deep Learning

---

- Deep learning is a renowned part of machine learning. 🔥
- :: AlphaGo.
- Deep learning uses artificial neural networks (NNs).
- Which are especially good at:
  - Computer vision.
  - Speech recognition.
  - Natural language processing (NLP).
  - Machine translation.

# 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” are the result-oriented models.
- 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

---

- The Learning Flow
- The Analysis Steps
- “Data”
- Visualization
- Preprocessing
- Dimensionality Reduction
- Statistical Models
- Machine Learning Models

# 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

# Common Jupyter Notebook Shortcuts

---

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

# Checkpoint: The Packages

---

- Open *00\_preface\_the\_packages.ipynb* up.
- Run it.

# The Learning Flow

# The Facts

---

- ::
- You can't learn *all* things in the data science!
- ∴
- “Let's do something after learning all things” 
- “Let's do something *while learning.*” 

# The Four Steps

---

1. Ask a question.
  - “How to tell the differences confidently?”
2. Identify the topic.
  - “Hypothesis testing.”
3. Explore the references of the topic
  - “T-test, ANOVA, ...”
4. Digest the references by the breadth-first way.
  - Write the code.
  - Make it work, make it right, finally make it fast.

# The Analysis Steps

# The Major Two Steps

---

1. Define the problem.
2. Resolve the problem.

# 1. Define the Problem

---

- Discuss the *feasible* goal.
  - It must come from the understanding of business.
  - Including all the gain and loss like users, resources, etc.
- Write the *exact* problem.
  - Rewrite it in a formal form, e.g.,
    - Statistical hypotheses.
    - Equations.

## 2. Resolve the Problem

---

- Collect the data.
- List the candidate methods.
  - Selecting Statistical Tests – Bates College
  - Choosing a statistical test – HBS
  - Choosing the right estimator – Scikit-Learn
  - A method also can be a simple median or a combination.
- Evaluate the candidate methods.
  - Understand the data by visualizing and *asking*.
  - Check the *assumptions* and the *metrics*.

- The best method achieves the goal?
  - Yes → Congrats! 
  - No → Check:
    - Method; note the assumptions.
    - Data; note the confounding variables, the correlations.
    - Problem; note the formalization.
    - Goal; it may be unachievable.

# Industry Changes Rapidly

---

- Need a fast iteration.
  - Resolve the small problems first.
  - Resolve the high C/P problems first.
  - One week to get a quick result and improve rather than one year to get the best result.
  - Fail fast!

**"Data"**

# “Data”

---

- = Variables.
- = Labels + Features.
- Features = Dimensions.

# Data in Different Types

---

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

# Data in the X-Y Form

---

y	x
label	features
dependent variable	independent variables
response variable	explanatory variables
regressand	regressors
endogenous   endog	exogenous   exog
outcome	design

- Confounding variables:
  - May affect  $y$ , but not  $x$ .
  - May lead erroneous conclusions.
    - Controlling, e.g., fix the environment.
    - Randomizing, e.g., choose by computer.
    - Matching, e.g., drug-vs-placebo, before-and-after.
    - Statistical control, e.g., BMI rather than height.
    - Double-blind, even triple-blind trials.

# Get the Data

---

- Logging.
- Experiments.
  - Strongly recommend Research Methods Knowledge Base.
- The existent dataset:
  - Kaggle
  - The Datasets Package – StatsModels

# Checkpoint: Pick up a Method

---

- Think of an interesting problem at work or in life.
  - E.g., revenue is higher, but is it random?
- Pick one or more methods from the cheatsheets.
  - Selecting Statistical Tests – Bates College
  - Choosing a statistical test – HBS
  - Choosing the right estimator – Scikit-Learn
- Remember:
  - The Learning Flow.
  - The Analysis Steps.

# 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

---

- We have three datasets.
- 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 among them.

# Preprocessing

# Feed the Data That Models Like

---

- Preprocess data for:
  - Hard requirements, e.g., corpus → vectors.
  - Assumptions, e.g.,
    - Samples are normally distributed.
    - Assuming features are centered around zero.
  - More representative features, e.g., total / 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 Outliners
  - *02\_3\_preprocessing\_removing\_outliners.ipynb*

# Checkpoint: Preprocess the Variables

---

- You already have the variables that interest you, right?
- Try to standardize them.
- Try to remove the outliers.

# Dimensionality Reduction

# The Model Sicks Up!

---

- Let's reduce the variables.
- Feed a subset → feature selection.
  - Feature Selection – Scikit-Learn
- Feed a transformation → feature extraction.
  - PCA, FA, etc.
  - Scikit-Learn defines it like non-numbers → numbers.

# Dimensionality Reduction

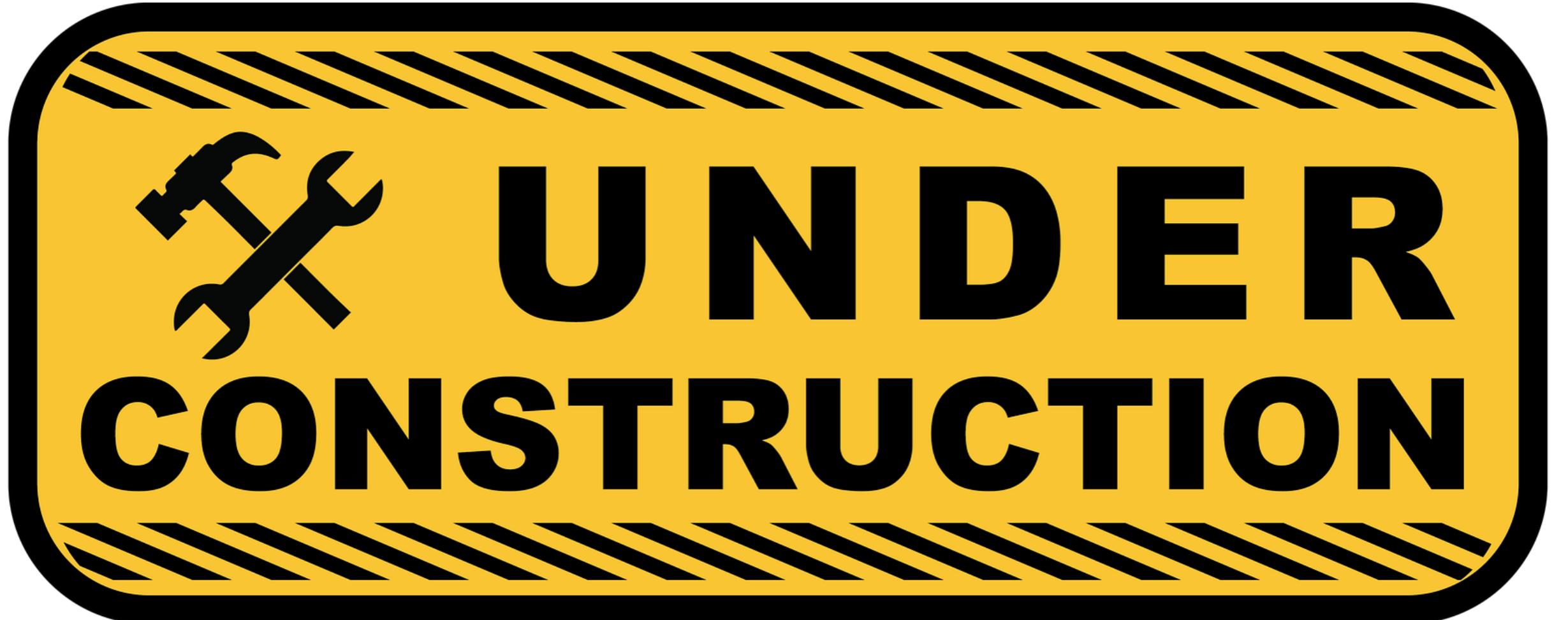
---

- Principal Component Analysis
- Factor Analysis
- *03\_dimensionality\_reduction.ipynb*

# Checkpoint: Reduce the Variables

---

- You must have a dataset that you are most interested in.
- Try to *PCA (all variables)* → *the better components*, or FA.
- And then plot n-dimensional data onto 2-dimensional plane.



**UNDER  
CONSTRUCTION**

# Statistical Models

# Statistical Models

---

- Identify Boring or Interesting – Hypothesis Testings
- Identify X-Y Relationships – Regression
- Identify Differences Among Groups – ANOVA
  - *04\_1\_statistical\_models\_t\_test.ipynb*
  - *04\_2\_statistical\_models\_reg\_anova.ipynb*

# Checkpoint: Apply a Statistical Method

---

- Understand the dataset more.
- Ask a question.
- Answer by the above methods.
- Try to apply the analysis steps.

# Machine Learning Models

# Machine Learning Models

---

- Apple or Orange? – Classification
  - Without Labels – Clustering
  - Predict the Values – Regression
  - Who Are the Best? – Model Selection
- *05\_1\_machine\_learning\_models.ipynb*

# Checkpoint: Apply a Machine Learning Method

---

- Understand the dataset more.
- Ask a question.
- Answer by the above methods.
- Try to apply the analysis steps.

**Keep Learning**

# Keep Learning

---

- Statistics
  - [Biological Statistics](#)
  - [scipy.stats + StatsModels](#)
  - [Research Methods](#)
- Machine Learning
  - [Scikit-learn Tutorials](#)
  - [Standford CS229](#)
  - [Hsuan-Tien Lin](#)
- Deep Learning
  - [TensorFlow](#)
  - [Standford CS231n](#)
  - [Standford CS224n](#)
- Big Data
  - [Spark](#)
  - [HBase](#)
  - [Hive](#)
  - [AWS](#)