Statistical Modelling for Data Science

Prof. Nino Narido

Outline for the day

- ❑ My background
- ❑ Why statistics is 'hot' today
- ❑ Why use Python for statistical modelling? What about other choices?
- ❑ Descriptive statistics (with Python)
- ❑ Inferential statistics (with Python)
- ❑ Application in machine learning/data science

Hands-on bootcamp on the same topic starting next week. Will be running for 4 weeks.

What will we learn in the bootcamp?

- ***Descriptive statistics, and probability basics***
  - Central tendency and dispersions, bivariate analysis
  - Probability concepts, counting stuff
  - Statistical distributions - discrete and continuous
  - Exploratory data analysis (EDA)
- ***Inferential and Bayesian statistics***
  - Estimation
  - P-values, confidence intervals
  - Hypothesis testing, ANOVA
  - Bayes' rule and its applications
- ***Machine learning and data science applications***
  - Linear regression
  - Logistic regression
  - Naive Bayes classification
  - Clustering and MLE

- NumPy
- Pandas
- Matplotlib
- Scipy
- Statsmodels
- Seaborn
- MLR
- Scikit-learn

My background

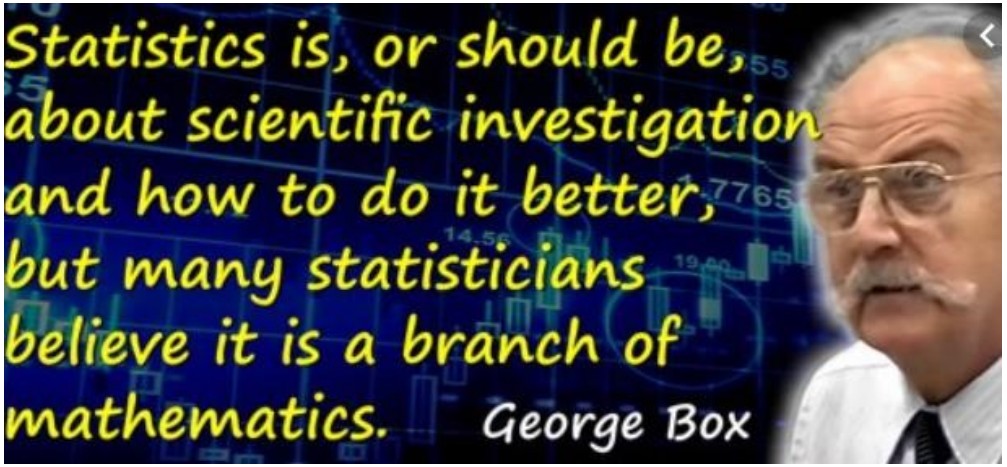
- IT Consultant/Professor, Solutioning Business Analytics and Data Engineering projects, also certified associate in SAP Analytics :-); leading various AI/ML projects in my organization for 12 years (IBM Consulting).
- Graduate Studies from University of the East, doing an Doctorate in IT. specializing Artificial Analytics from MCST now (never stop learning).
- Member of "Data Science Central", largest online platform for data science/ML articles, publish regularly on DS/ML topics
- Teaching "Data wrangling using Python" practicing Advanced computational MLops and LLMops, working on "Hands-on Data Science for BS in IS/IT/CS" lecture in MCST/UMAK
- Open-source advocate, released user of Python packages, may use one of them in the linear regression portion
- LinkedIn, Github links here - feel free to connect and fork my code repos

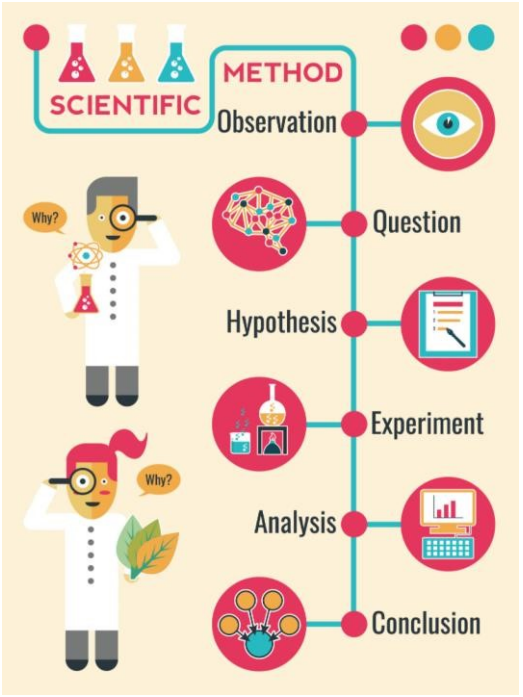
Takeaway

- ***You will learn all there is to learn about statistics for data science*** - this is the starting hypothesis. What is the probability of that?
- You are right. Probability of above is vanishingly small ~ zero! (But how to estimate it anyway?)
- Goal is to
  - Expose you to the core methodology of statistical analysis and modelling,
  - Python packages and functions to accomplish some of those methods,
  - Impress upon you the habit of statistical way of thinking and coding up quick test/scripts to test ideas,
  - Illustrate core concepts through programming and list advanced concepts so that you can go back and explore.

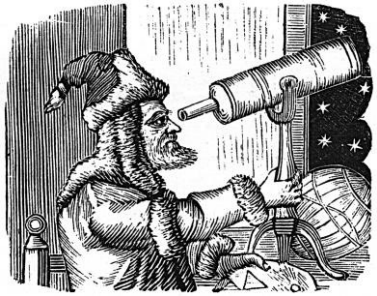
Why statistics is the foundation of modern data science?

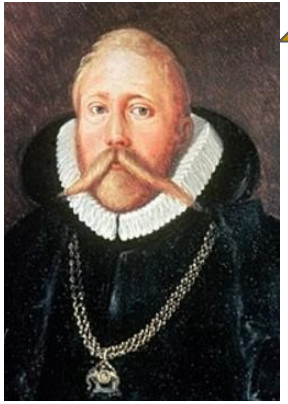
Statistics and the '***Scientific Method***'

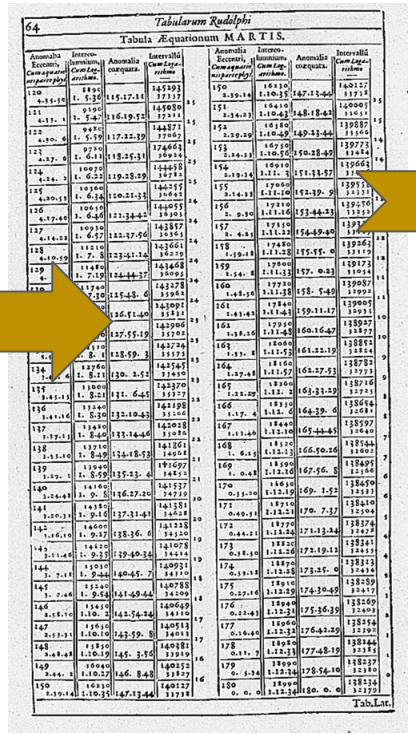


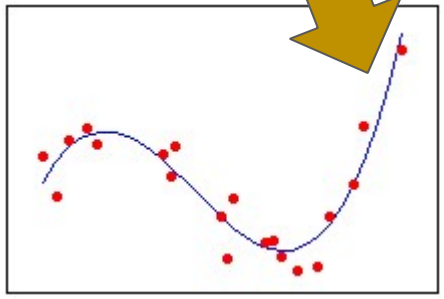


Example of Data Science in 17th century...



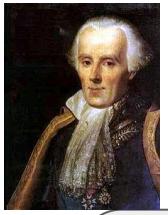




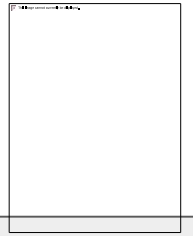


The modern science was kickstarted - using data table, and curve fitting...

A brief history of statistics...








Middle of 18th century

19th century

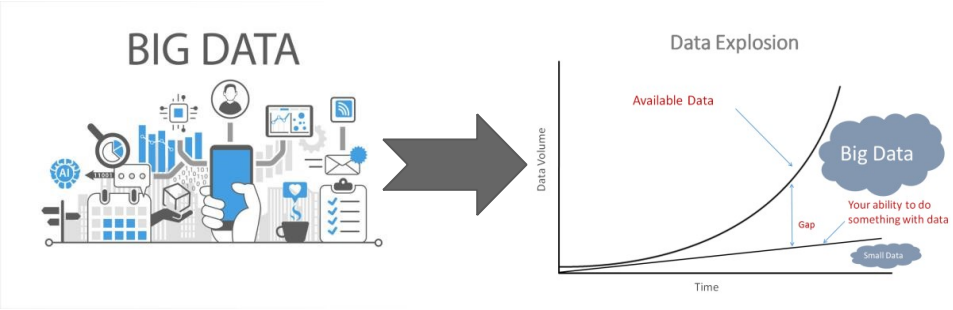
20th century

Tax, economy, military data collection

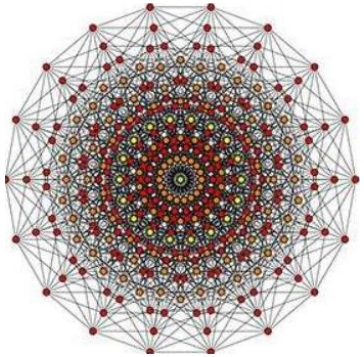
Probability, uncertainty, mathematical foundation

Inference, prediction, modern concepts

Why is statistics ‘hot’ today?



High- dimension



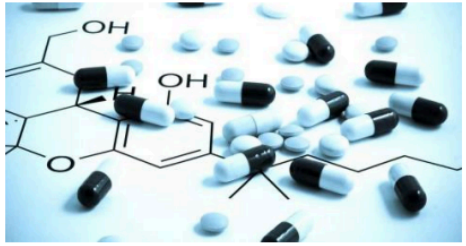
But statistics was developed to handle ‘**Small Data**

- Incredibly rich and mathematically sound **modeling and prediction** techniques
- Standard way to deal with **uncertainty**
- Fast, ‘good enough’ **alternative** to model data w/o using heavy computation
- Bayesian approach **fits naturally** with the mantra of ‘*update your belief*’

Examples of statistics in action...

Linear and non-linear regression

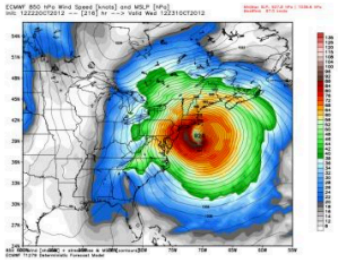
Probabilistic modeling for motion control



Hypothesis testing, ANOVA

Descriptive stats, visualization

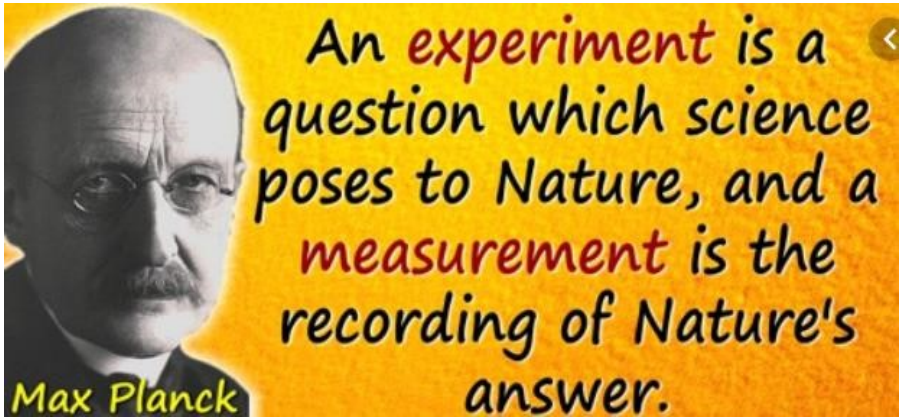
Failure analysis, QA models

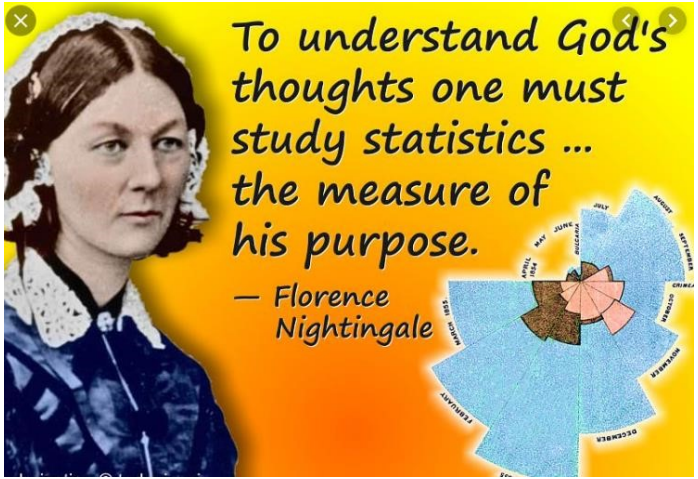


Time series modeling, geo-spatial regression

Descriptive statistics (this is the stuff you also see as the major part of “*Analytics*”)

Measure and describe...

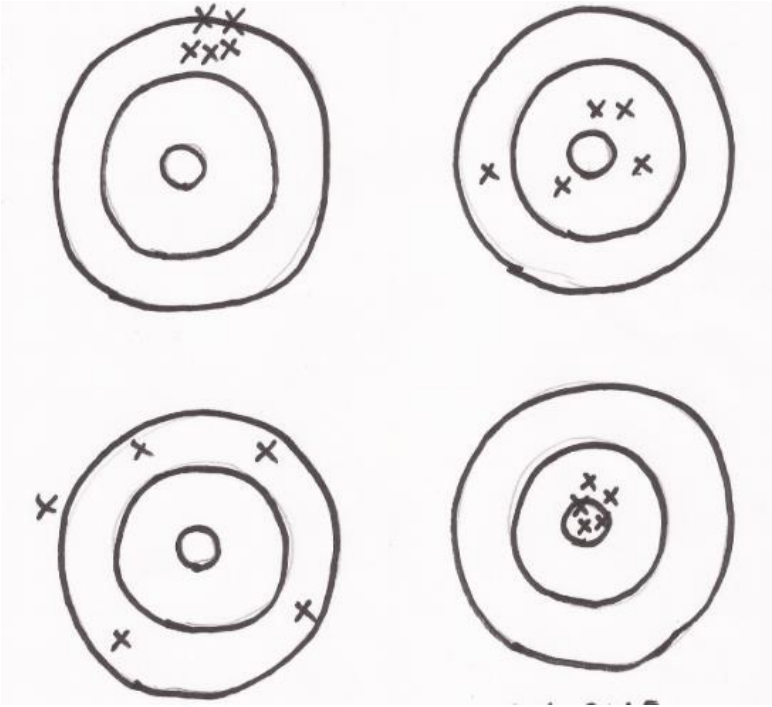




Deep scientific thoughts have been inspired by measuring raw data for patterns

Statistics has been serving society always centuries before data science took and analyzing off :-)

Central tendency and dispersion



There are ups and downs in life. Same for data. There are peaks and troughs but also a baseline.

What is that one number toward which all the measurements tend to gravitate?

But there are enormous variety among datasets. Some look like a close-knit family, others behave like highly dispersed organization.

Need to measure dispersion or spread

Mean(s), median, mode

(Arithmetic) mean: Colloquially called the all too familiar '*Average*'.

Median: The central quantity in a dataset, when it is sorted by (increasing or decreasing) values. Robust to outliers.

Mode: The most frequently occurring data. Mainly useful for categorical datasets.

Geometric mean: Used in special circumstances (datasets where multiplicative growth is observed)

Harmonic mean: Used in special circumstances.

Variance and standard deviation

Variance: A measure of the *average* (squared) distance of data from their mean value. Squaring is done to turn distance measures positive.

Standard deviation: Just the (positive) square-root of variance to make the quantity comparable to the actual data points in terms of physical units.

These concepts are general. ***Do not equate them with the 'Bell curve'****. Your dataset may not follow a Bell curve, and you may not be able to apply six sigma techniques but* ***you can always compute mean and variance****.*

‘Pythonic’ way of doing descriptive stats…

```
from random import randint
lst = []
for _ in range(1000000):
    lst.append(randint(1,100))
```

1 million data points

```
len(lst)
```

```
1000000
```

```
t1 = time()
for _ in range(100):
    sum = 0
    for num in lst:
        sum+=num
    mean = sum/len(lst)
t2 = time()

print("Mean: {}\nAverage time taken for computing the mean using for loop: {} seconds ".format(mean,(t2-t
1)/100))
```

```
Mean: 50.539717
Average time taken for computing the mean using for loop: 0.06872276782989502 seconds
```

Not one-off result, the simulation is done 100 times and averaged over.

Use vectorized code and functions whenever you can.

Use **`NumPy`** and **`Pandas`** libraries.

```
t1 = time()
np_lst = np.array(lst)
for _ in range(100):
    mean = np_lst.mean()
t2 = time()

print("Mean: {}\nAverage time taken for computing the mean using NumPy: {} seconds ".format(mean,(t2-t1)/1
00))
```

```
Mean: 50.539717
Average time taken for computing the mean using NumPy: 0.001326603889465332 seconds
```

What is bivariate analysis?

- As the name suggests, it involves 'bi' or two variables.
- It has a special place in analytics.
- We humans are most comfortable with a X-Y plot. We cannot perceive more than 3-D, and 3-D plots are hard to interpret in most cases. So, a simple **2-D visualization** works best for data modeling and analysis.
- However, the bivariate analysis goes far beyond just the visualization. It also involves - **correlation, simple linear regression,** etc.

Keywords: "*Population*", "*Sample*", "*Estimation*"

- Population represents the entire mass of objects about which some measurement can be made for statistical purpose. Almost **impossible to obtain** data about.
- Sample is a tiny, but **fairly representative**, fraction of a population, used for actual statistical modeling.
- The central goal of statistics is to **'estimate' properties** of the population by examining the sample - **estimation**.
- Note the parallel to ML - **generalization**.

Scatterplot: Bread and butter of bivariate analysis

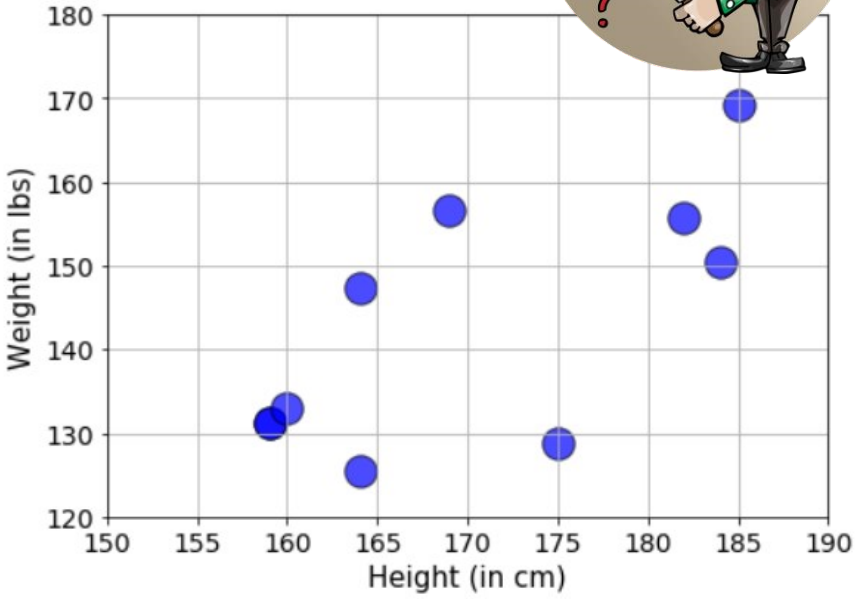
Suppose, we want to want estimate the correlation between height and weights of high-school students.

We cannot obtain data for all U.S. high-school students easily/cheaply.

So, we sample from our local school.

Then, we visualize the data in a 2-D plot (shown on the right).

- **Is there a 'relationship'?**
- **Is the relationship linear?**
- **Is there a cause-effect?**



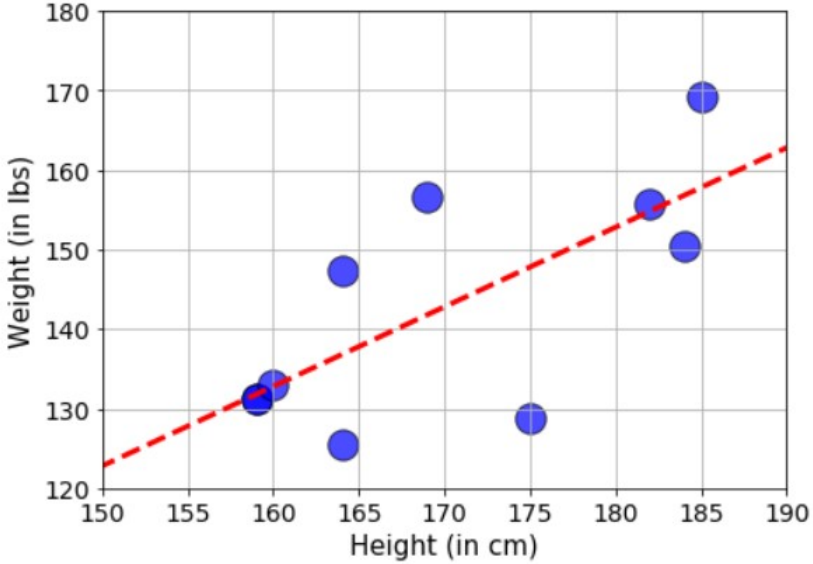
"*Fitting the data*": Just a quick view

If we fit a linear model with this sample data, we see the red dashed line shown in the right hand side figure.

Is this line the **best we could do?**

What does it mean? Can I **predict** weight given any height data from this line?

Will a **jiggly nonlinear** fit be better than a simple straight line?



These questions will be re-visited multiple times later and discussed at depth.

Correlation and its measure

At the minimum, even before we do any kind of fitting, we want to do the **extent of correlation** between the two sets of data points - weight and height.

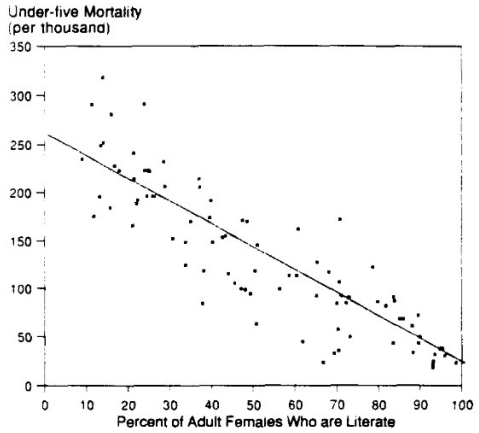
Being correlated means - moving in the same general direction as the other variable moves. Note it involves direction, so it is like a vector. Its sign matters.

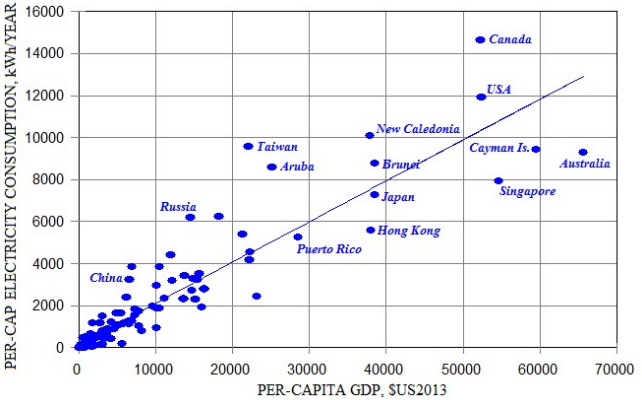
If we see a general increase in weight as the height increases, then we say they are '**positively correlated**', and the correlation coefficient is a positive number. And vice versa for the negative case.

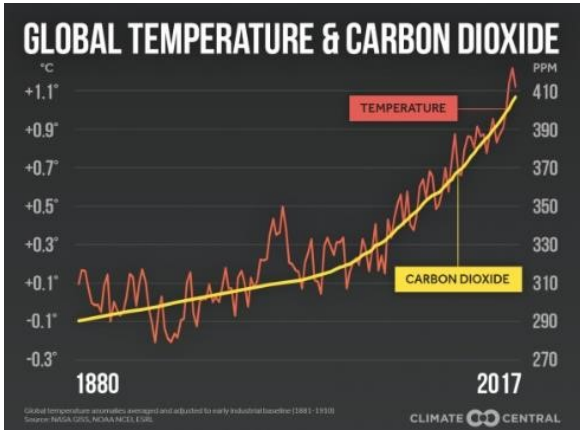
It is measured by so-called **correlation coefficient** (ranges from -1 to 1).

Note that, correlation is a strictly **linear concept**. If $y = x^2$, then the correlation coefficient is zero even though y values are completely determined by x.

Some simple examples from the real world







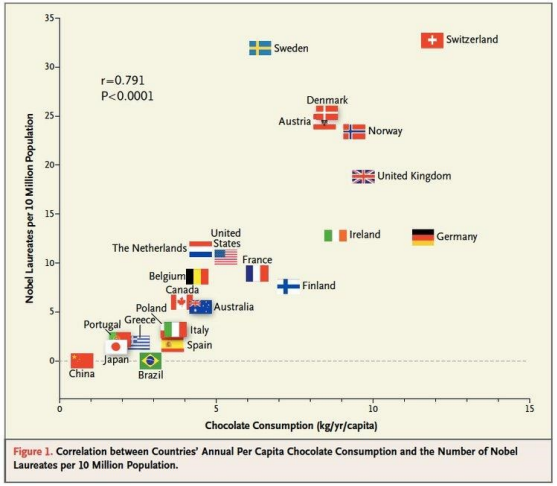


Figure 1. Correlation between Countries' Annual Per Capita Chocolate Consumption and the Number of Nobel Laureates per 10 Million Population.

The NEW ENGLAND JOURNAL of MEDICINE

OCCASIONAL NOTES

Chocolate Consumption, Cognitive Function, and Nobel Laureates

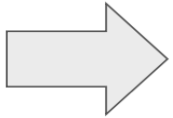
Franz H. Messerli, M.D.

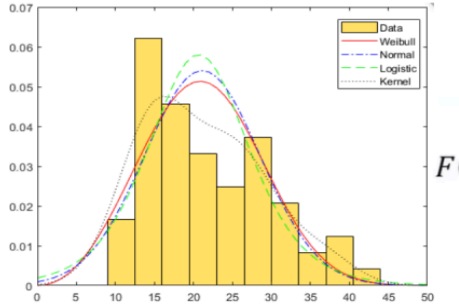
Probability: The cornerstone of modern data science and machine learning

“Greed is Good”

“Greed is Good”

A gambler's dispute in 1654 led to the creation of a mathematical theory of probability by two famous French mathematicians, **Blaise Pascal** and **Pierre de Fermat**.





$$F(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2}$$

Two camps - **frequentists** and Bayesians

" Two dice are rolled together. What is the probability of getting a total which is a multiple of 3?"

This problem is solved by **Counting**. You count and enumerate all the possible outcomes such as {1,1}, {1,2}, {2,1}... {6,6}, then count the number of outcomes where the total is a multiple of 3, such as {1,2}, {2,1}, {2,4}, {4,2}, etc. Then you just take the ratio.

You count and enumerate all the outcomes such as {1,1}, {2,1}, {2,4}, {4,2}, etc.

Count the number of outcomes where the total is a multiple of 3, such as {1,2}, {2,1}... {6,6},

Take the ratio

possible {1,2},

Would Sir Isaac Newton approve this method?

Where in data science/machine learning?

- Data is inherently noisy and uncertain. But customers want a "confident" answer from you, the data scientist.
- Therefore, you must pass the data through the sieve of probability to be able to give some guarantee to your customer.
- Probability theory is applied in virtually all machine learning and statistical modeling techniques -
  - Linear Regression, Logistic regression, Poisson regression
  - Naive Bayes classification, O/P layer of most deep learning models
  - Clustering such as k-means
  - Ensemble techniques such as Random forests, Bagging, Boosting
  - Language models (NLP), text mining
  - Advanced AI algorithms - Markov processes, Bayesian networks, Reinforcement learning

Definitions

- **Random variable**: A variable whose possible values are numerical/categorical outcomes of a random/natural/statistical phenomenon or experiment. There are two types of random variables - discrete and continuous.
- **Trial/experiment**: Scientifically designed test/experiments to measure statistical properties of a population or a natural phenomenon. Often randomized to reduce selection bias and derive causation. Example - clinical drug trial, political polling.
- **Control group**: The control group is defined as the group in an experiment or study that does not receive treatment by the researchers and is then used as a benchmark to measure how the other tested subjects do.
- **Expectation/Expected value**: The expected value of a random variable, intuitively, is the long-run average value of repetitions of the same experiment.

Probability distribution

"A probability distribution is a statistical function that describes all the possible values and likelihoods that a random variable can take within a given range."

- Quantifies uncertainty
- Random variable
- Used for statistical inference
- Maps I/P to O/P
- Has a domain of validity
- Outputs discrete and continuous values

Discrete probability distributions

A discrete probability distribution (applicable to the scenarios where the set of possible outcomes is discrete, such as a coin toss, or a roll of dice, or a quality failure event in manufacturing) can be encoded by a discrete list of the probabilities of the outcomes, known as a **probability mass function**.

Bernoulli distribution

Binomial distribution

Poisson distribution

Geometric distribution

Continuous probability distributions

A continuous probability distribution (applicable to the scenarios where the set of possible outcomes can take on values in a continuous range (e.g. real numbers), such as the temperature on a given day) is typically described by **probability density functions** (with the probability of any individual outcome actually being 0).

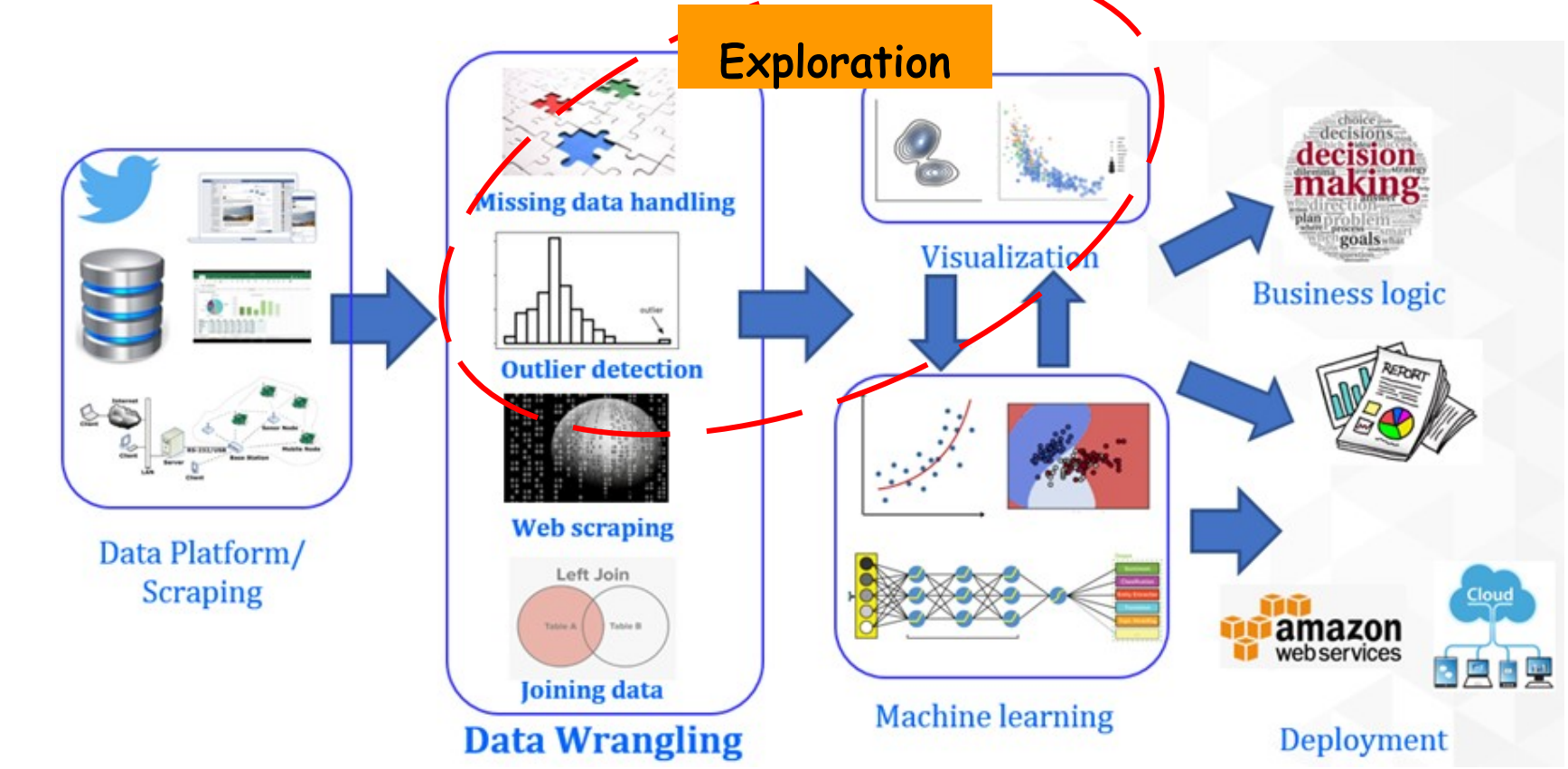
Formally, if X is a continuous random variable, then it has a probability density function $f(x)$, and therefore its probability of falling into a given interval, say [a, b], is given by the integral

$$\mathrm{P}[a \leq X \leq b] = \int_a^b f(x)\,dx$$

- **Normal distribution**
- **Weibull distribution**
- **Chi-square distribution**
- **Student's t-distribution**
- **F- distribution**
- **Uniform distribution**

Exploratory data analysis (EDA) for data science

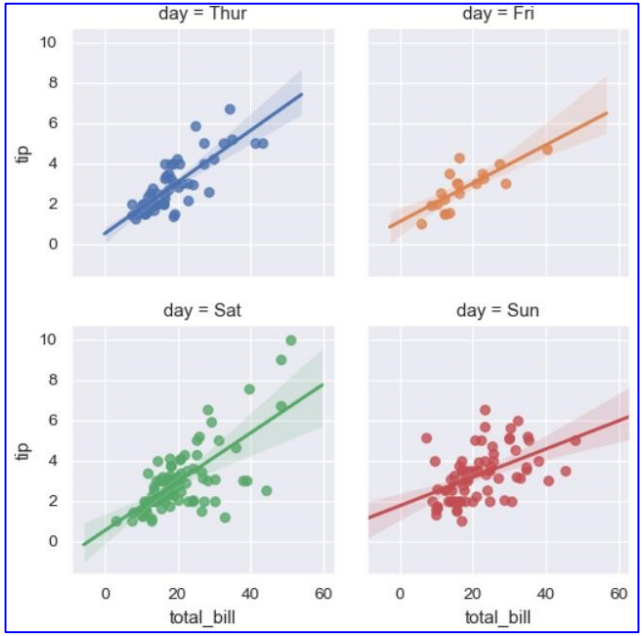
Data wrangling at the forefront



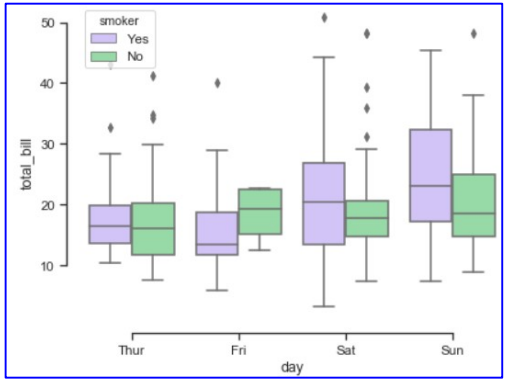
EDA: Visualize, probe, detect, summarize

- While creating visualizations - think what will bring out the unique characteristics of the data and the business problem.
- Do not go fancier than what is needed. Avoid 3-D plots if possible.
- Are interactive plots helpful? If yes, add them with minimalistic user controls. But do not create interactive plots with more than one/two controlling variables.
- Use plots and charts for probing anomalies and easy-to-find patterns in the data. If something looks odd, probe into the source and cause. Add note, record assumptions, pass on any recommendation up the model chain.
- Always try to perform basic descriptive statistics (if the scale allows) to gather numerical scores.
- Summarize key insights of EDA, so that machine learning or business intelligence team can have a baseline to start with.

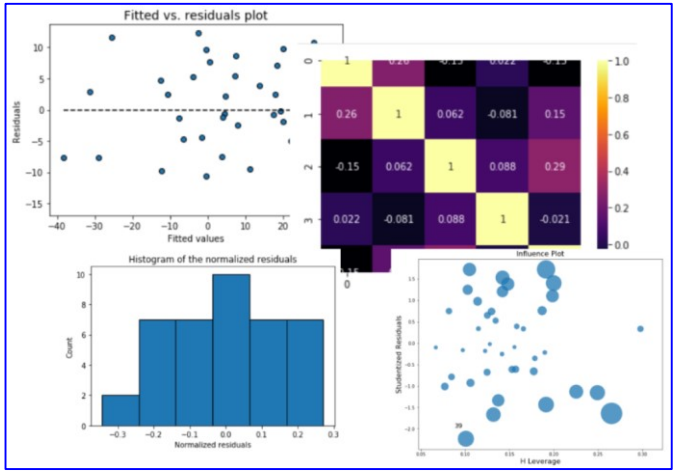
Purposeful plots and charts



Clean scatter plots in a grid, showing the linear fit and confidence interval



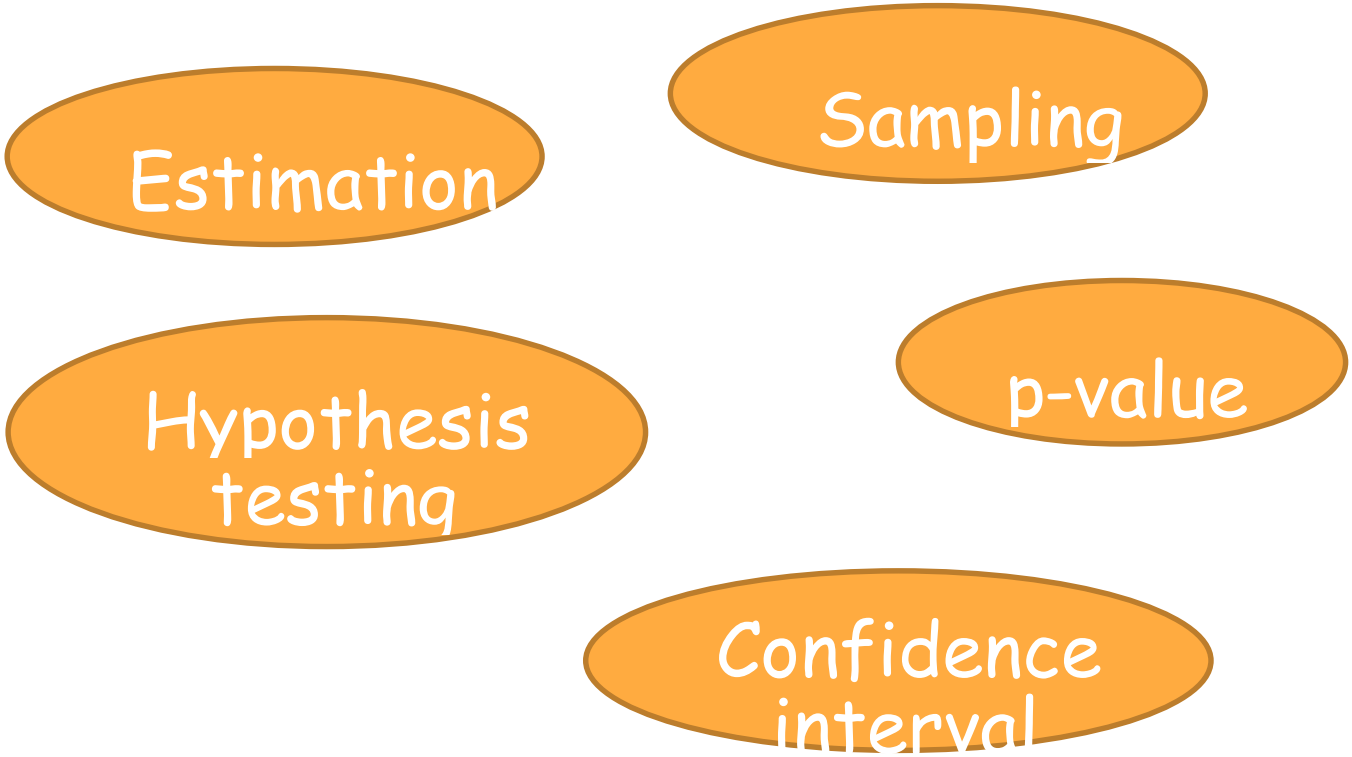
Box plots with outliers, aided by color to include multiple categorical variables



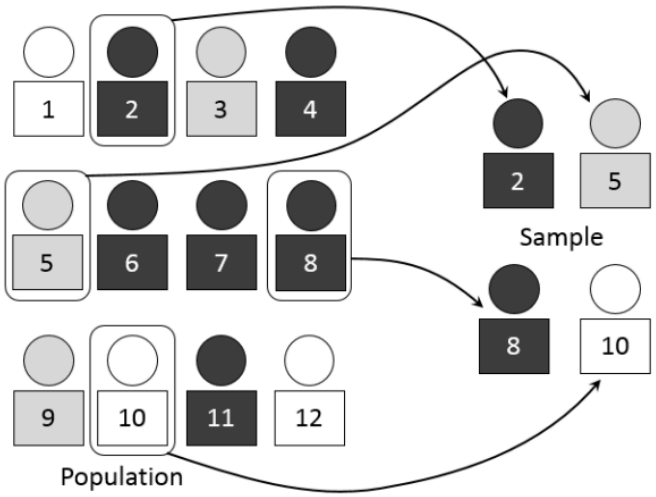
Residual plots, histogram, leverage plots with dynamic marker size to show outliers, correlation matrix etc., for a regression problem.

Inferential statistics

Few concepts of inferential stats



Sampling

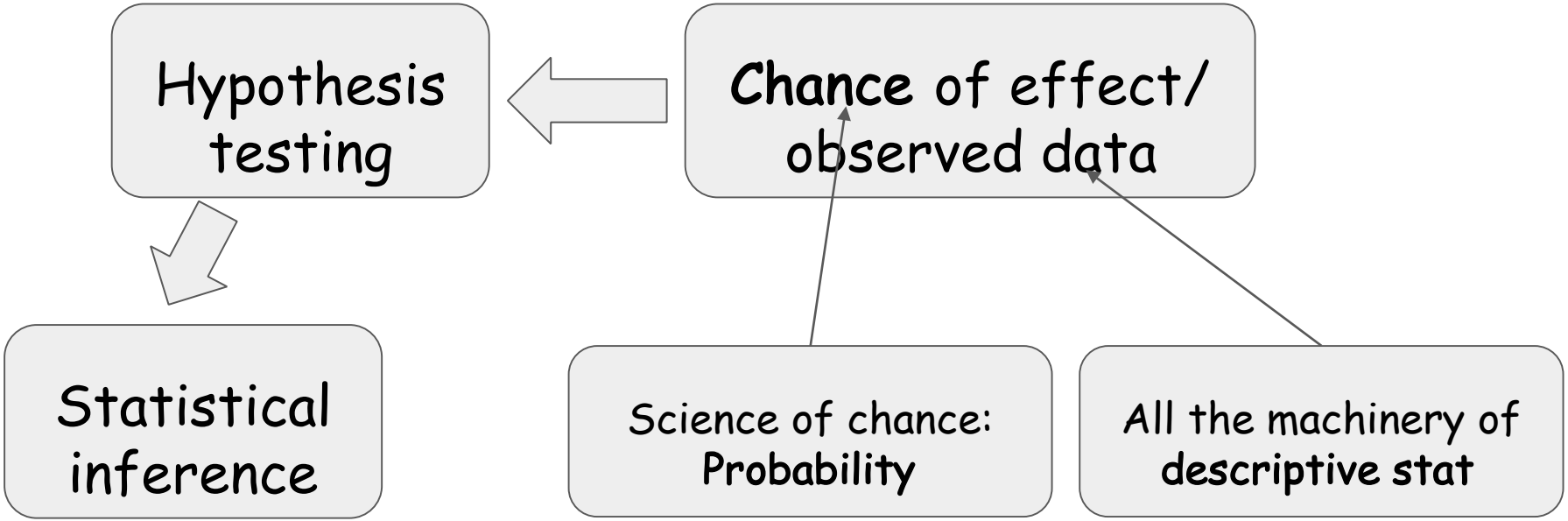


- Simple random sampling
- Systematic (order-based) sampling
- Stratified sampling
- Probability-proportional-to-size sampling
- Cluster sampling
- Sampling with/without replacement

https://www.math.upenn.edu/~deturck/m170/wk4/lecture/case1.html

Common sense sampling :-)

The link between descriptive and inferential statistics



“My name is Bond… James Bond”

Suppose we gave Mr. Bond a series of 16 taste tests. In each test, we flipped a fair coin to determine whether to stir or shake the martini.

Then we presented the martini to Mr. Bond and asked him to decide whether it was shaken or stirred.

Let's say Mr. Bond was correct on 13 of the 16 taste tests.

Does this prove that Mr. Bond has at least some ability to tell whether the martini was shaken or stirred?

Was 007 just lucky or he has it in him?

Mr. Bond could have been just lucky and guessed right 13 out of 16 times. **But how plausible is the explanation that he was just lucky?**

To assess its plausibility, we determine the **probability that someone who was just guessing would be correct 13/16 times or more**. This probability can be computed from the **binomial distribution**, and the binomial distribution calculator shows it to be 0.0106.

This is a pretty low probability, and therefore someone would have to be very lucky to be correct 13 or more times out of 16 if they were just guessing. So either Mr. Bond was very lucky, or he can tell whether the drink was shaken or stirred.

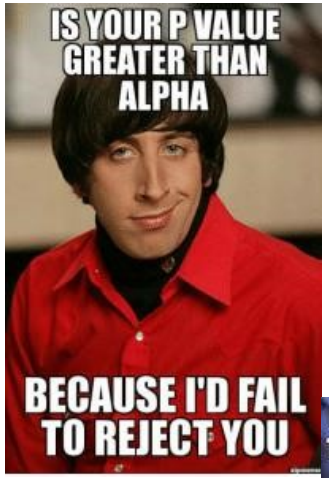
The hypothesis that he was guessing is not proven false, but considerable doubt is cast on it. Therefore, there is strong evidence that Mr. Bond can tell whether a drink was shaken or stirred.

P-value and statistical significance

In hypothesis testing, the p-value or probability value is, for a given statistical model, the probability that, when the null hypothesis is true, the statistical summary would be equal to, or more extreme than, the actual observed results.

In the previous example, the probability value is the **probability of an outcome given the hypothesis**. It is **not the probability of the hypothesis given the outcome.**

Statistical significance is a predetermined level, below which we reject the NULL hypothesis. **A popular choice is p=0.05.**



Common examples of statistical inference in play

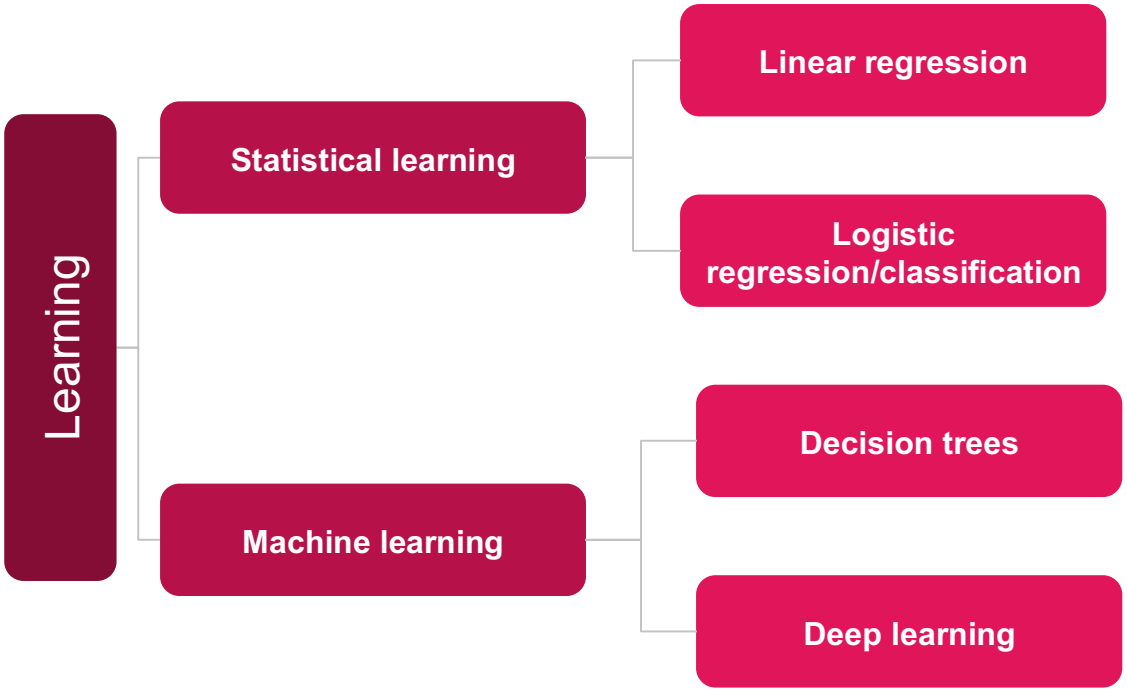
- **Drug efficacy testing** ("This new medicine trial shows 20% more effectiveness over the current best one")
- **Advertising claims** ("My product/service/ad reduces the cost of business by 15% on average/increases the click-through rate by > 30%")
- **Scientific discovery claims** ("My research group claims with 99% confidence that this gene is the root cause behind this disease")
- **Political polls** ("There is a 7 percentage points difference between the means of the groups who support this candidate vs. those who don't, p-value is < 0.05")
- **Your online behavior**. Google, Amazon, Facebook are constantly subjecting you, the consumer, to A/B testing, which is another form of hypothesis testing. ("Showing this recommendation increased the revenue vs. the alternative one.")

Statistical learning and machine learning: Are they same?

Statistical learning vs. Machine learning

The process or machinery of the statistical-inference-based learning is somewhat opposite to the general principle of machine learning, as we understand and practice it.

In ML, we follow (or want to follow) **inductive generalization** i.e. given the data we want to learn some general truth. In statistical learning/inference, we assume a known truth/distribution and computes the probability of data not supporting that truth/not coming from that distribution.

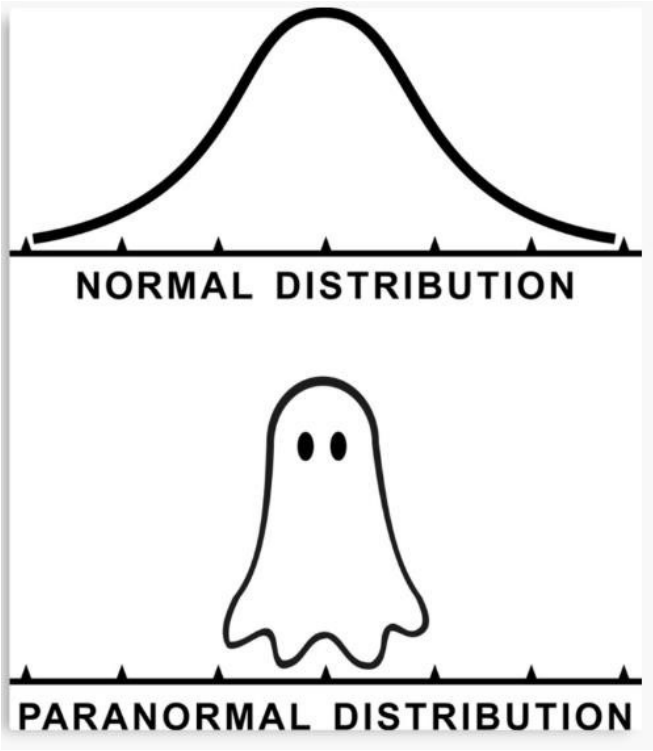


A practical illustration of how "*we assume a truth*"

A large bodyof process control, quality assurance, and reliability testing in manufacturing depends on the concept of '**sixsigma**'. Thousands of managers and engineers live by this principle everyday.

Related theory depends on the preconceived notion that the **data points are coming from a perfect Gaussian distribution**.

What if the underlying distribution is not a Gaussian Normal?



You need both skills to do well in data science

- You need sound background in conventional statistical analysis, modeling, and inference techniques to answer **important questions for your business and product**.
- Statistical learning/inference skills will give you ability to **reject myths, unscientific claims**, and get the most truth out of the given data. It gives you **ability to choose between alternatives**, which is almost always what businesses are doing to maximize value to shareholders.
- You also need to acquire machine learning skills to be able to **predict and learn continuously** based on the data that is always coming at you.
- We will also see, how another branch of statistical learning - **Bayesian inference** - is more aligned to the machine learning machinery learn and update model based on new data.

Bayesian statistics

What can Bayes' rule do?

- Bayes' theorem **converts the results from your test into the real probability** of the event. For example, you can:
- **Correct for measurement errors**. If you know the real probabilities and the chance of a false positive and false negative, you can correct for measurement errors.
- **Relate the actual probability to the measured test probability**. Given mammogram test results and known error rates, you can predict the actual chance of having cancer given a positive test.

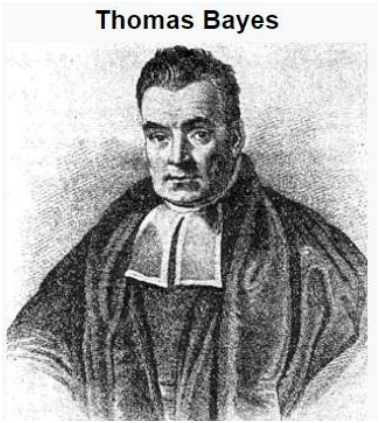
Bayes' rule/theorem

So, we saw $P(A|B) = \frac{P(A \bigcap B)}{P(B)}$

They are the same!

Interchanging A and B $P(B|A) = \frac{P(B \bigcap A)}{P(A)}$

$$P(A|B) = \frac{P(B|A).P(A)}{P(B)}$$



Bayes' theorem (alternatively Bayes' law or Bayes' rule) describes the probability of an event, based on prior knowledge of conditions that might be related to the event. For example, if a disease is related to age, then, using Bayes' theorem, a person's age can be used to more accurately assess the probability that they have the disease, compared to the assessment of the probability of disease made without knowledge of the person's age.

A new way of doing data science and learning...

Posterior **Likelihood** **Prior**

$$P(Hypothesis|Data) = \frac{P(Data|Hypothesis).P(Hypothesis)}{P(Data)}$$

Marginalization

We start with a hypothesis.

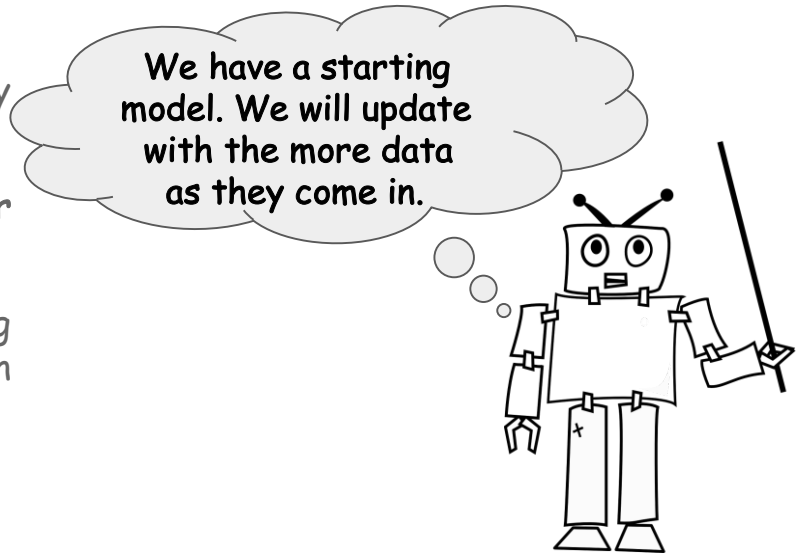
Then, we gather data, and update our initial belief.

If the data supports the hypothesis then the probability goes up, if it does not match, then probability goes down.

But in majority of statistical learning, the notion of **Prior** is not used or not looked favorably.

Also, the computational intricacies of Bayesian learning has prevented it from being mainstream for more than two hundred years.

But things are changing now...



The original autonomous vehicle

https://stanford.edu/~cpiech/cs221/apps/driverlessCar.html

The Story

The Google driverless car is a project that involves developing technology for a car to drive without input from a human. The project is currently being led Stanford researcher Sebastian Thrun, a professor in the Stanford Artificial Intelligence Laboratory and co-inventor of Google Street View. Thrun's team at Stanford created the robotic vehicle Stanley which won the 2005 DARPA Grand Challenge and its US$2 million prize from the United States Department of Defense.

Stanley the driverless car.

AI Techniques

Localization

In order to drive the car must first know its location within its environment however it is not possible to use an instrument to measure exactly where a driverless car is. Even the best GPS sensors have a margin of error of around 5 meters (and think about what that means for a car given that the standard lane width is around 3.5 meters). Finding where a robot is when it can't measure location directly is called localization. There are several AI techniques that allow a driverless car to incorporate temporal, noisy input to generate a probabilistically sound estimate.

For localization, the underlying model is generally some form of Bayesian model similar to a hidden markov model where the state space of the the unknown "location" variables are continuous. At each time point *t* there are two variables: an unknown variable which is the location of the car, *x(t)*, and observations about the car's location based on the sensor inputs at that given time, *y(t)*. The model assumes that x(t) is generated from x(t - 1) with some unknown distribution and that y(t) is generated from x(t) with some unknown distribution.

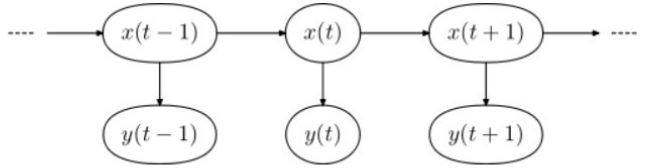
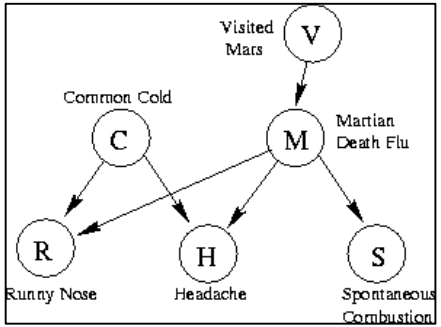
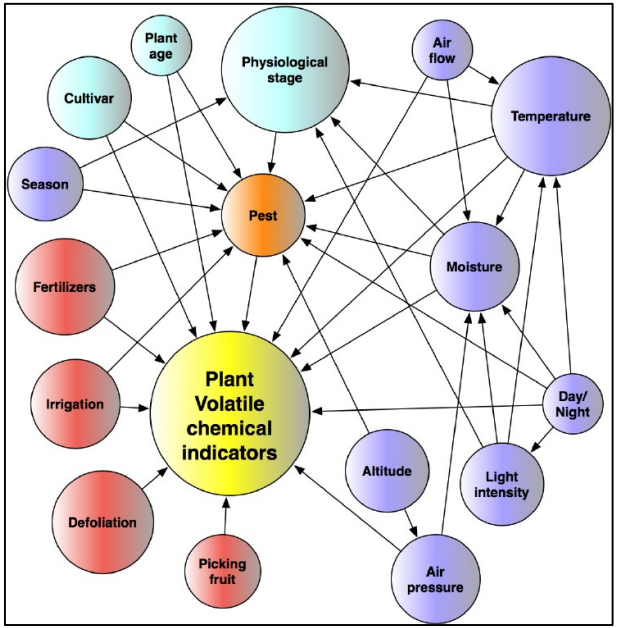
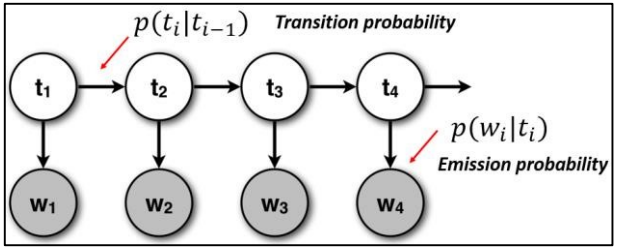


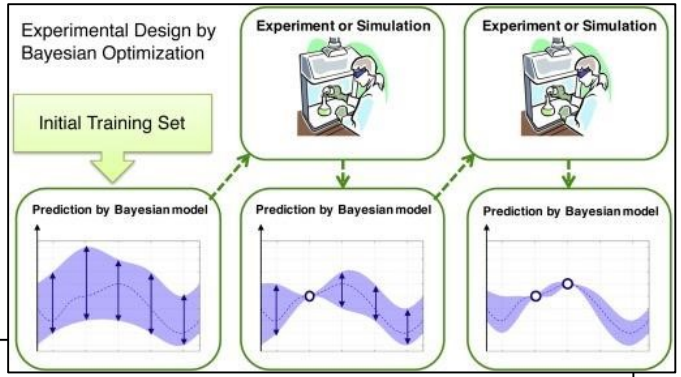
Figure 1: The baysian model for localization.

Bayesian network and optimization









Bayesian Optimization in AlphaGo

Yutian Chen, Aja Huang, Ziyu Wang, Ioannis Antonoglou, Julian Schrittwieser, David Silver, Nando de Freitas

(Submitted on 17 Dec 2018)

During the development of AlphaGo, its many hyper-parameters were tuned with Bayesian optimization multiple times. This automatic tuning process resulted in substantial improvements in playing strength. For example, prior to the match with Lee Sedol, we tuned the latest AlphaGo agent and this improved its win-rate from 50% to 66.5% in self-play

Summary and conclusions

Cliché but true ☺

Google's Chief Economist Hal Varian on Statistics and Data

> *I keep saying the sexy job in the next ten years will be statisticians. People think I'm joking, but who would've guessed that computer engineers would've been the sexy job of the 1990s?*

Build the habit of *statistical thinking*

- Do I have a 'data problem' i.e. can a data-driven approach solve my business/science problem? Am I comfortable with uncertainty? Are my customers?
- What kind of descriptive stats I need? Data wrangling and cleaning? What is an effective visualization for me?
- What kind of inference I need to draw from the data?
- Starting hypotheses? How wide my confidence interval should be? What kind of error margin I can tolerate?
- How often do I need to update my model? Prediction ability and speed?