Probability for HACKERS

Nipun Sadvilkar
Pycon India 2017

< About me >

- Machine Learning Engineer @Juxt-Smart Mandate
- AI and ML Enthusiast
- Likes to crack puns- Ni..pun (^-^)
- @nipunsadvilkar on GitHub
- More on website:

https://nipunsadvilkar.github.io/

#Questions:

1. How many of you are from heavy mathematical background?

   E.g. Engineering, Physics

2. How many of you have used ML libraries like sklearn in your work?

3. How many of you want to get into following fields?
   - Artificial Intelligence
   - Machine Learning
   - Deep Learning
   - Data Science

MOTIVATION #1

Prerequisite for any famous AI and ML course/Book

CS229
Machine Learning
Autumn 2016

Course Information

Instructors:

Andrew Ng, John Duchi

Course Description

This course provides a broad introduction to machine learning and statistical
(generative/discriminative learning, parametric/non-parametric learning, neur
dimensionality reduction, kernel methods); learning theory (bias/variance trad
control. The course will also discuss recent applications of machine learning,
bioinformatics, speech recognition, and text and web data processing.

Prerequisites

Students are expected to have the following background:

- Knowledge of basic computer science principles and skills, at a level suf
- Familiarity with the probability theory. (CS 109 or STATS 116)
- Familiarity with linear algebra (any one of Math 104, Math 113, or CS 20

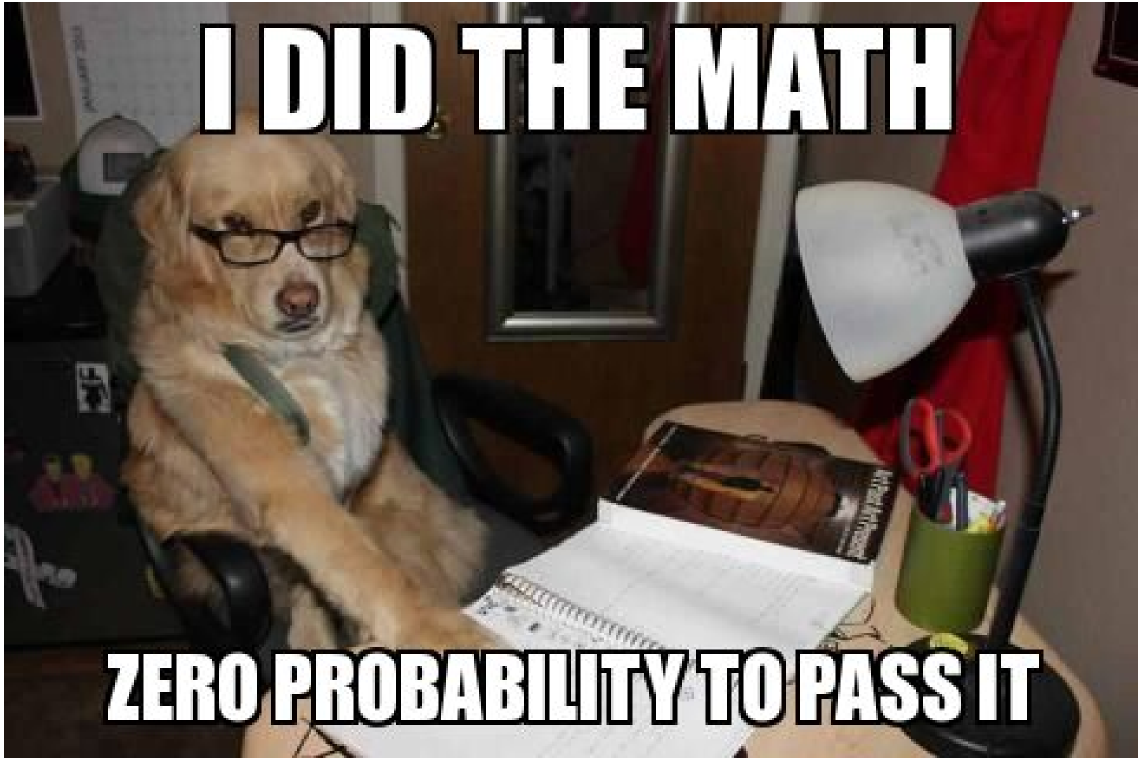
Deep Learning

An MIT Press book

Ian Goodfellow and Yoshua Bengio and Aaron Courville

Deep Learning

Those who struggle with math be like:



 Hackers' approach to learn Math

- Math is difficult but through coding, we can make it more interactive and intuitive.

- I like this quote:

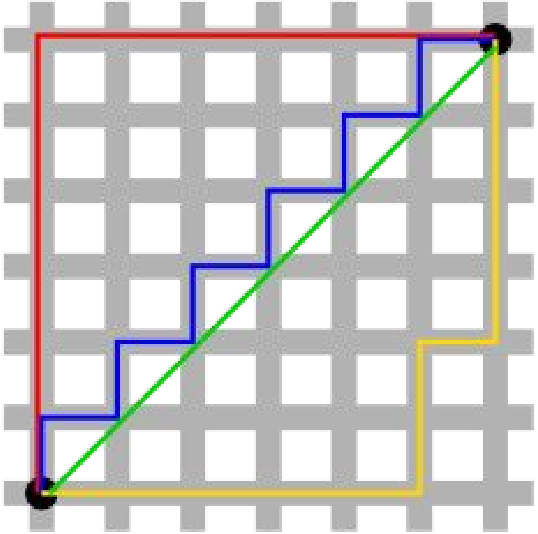
  "Statistics is Hard.
  Using programming skills it can be easy"
  – *Jake VanderPlas (Statistics for Hackers - Pycon 2016)*
  [Same for Probability]

- Though, I want you to focus more on concepts and not on code (Code is available on GitHub. Have a look at it later)

MOTIVATION #2

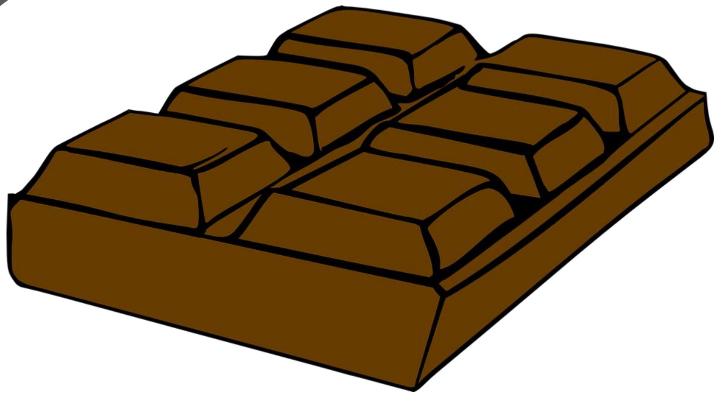
Modern AI

"Study and design of any agent that behaves in an intelligent way"



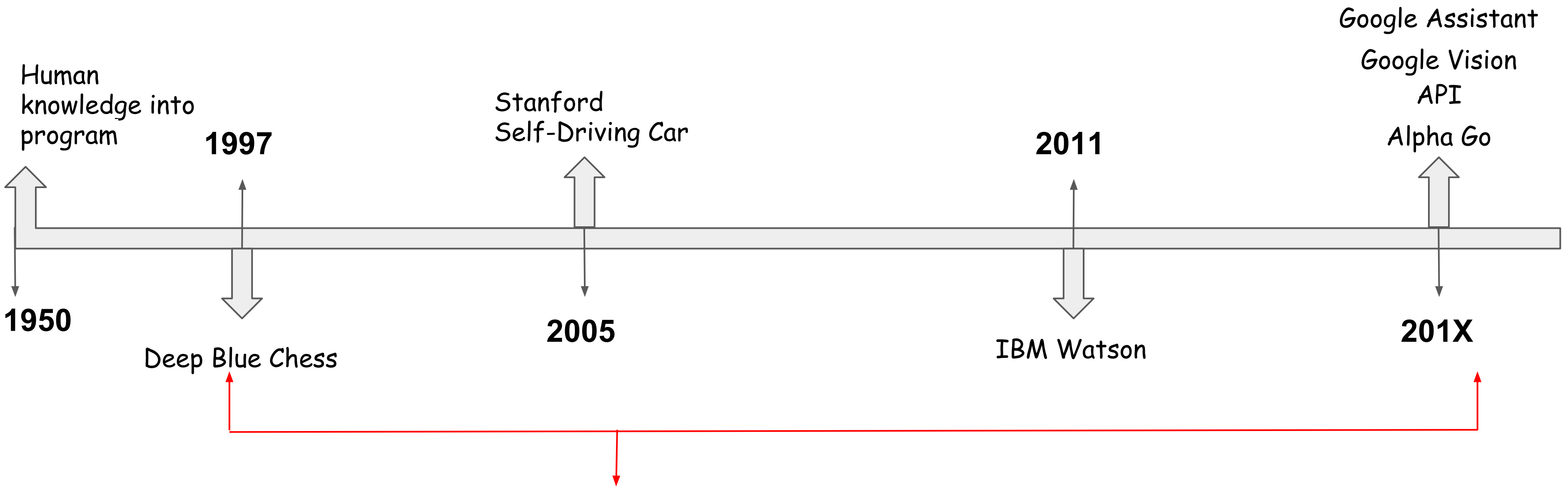
Demo Time! 🕒

WHO LIKES CHOCOLATES HERE?



MOTIVATION #2

HISTORY OF AI



Uses two revolutionary ideas:

1. Probability with Examples
2. Artificial Neurons

Conclusion

"Not once, but twice AI was revolutionized by people who understood Probability Theory"

- Stanford University | CS 109: Probability for Computer Scientists

TARGET

To be able to understand following math

$$p(C_k \mid \mathbf{x}) = \frac{p(C_k)\ p(\mathbf{x} \mid C_k)}{p(\mathbf{x})}$$

$$\begin{aligned} p(C_k \mid x_1, \dots, x_n) &\propto p(C_k, x_1, \dots, x_n) \\ &\propto p(C_k)\ p(x_1 \mid C_k)\ p(x_2 \mid C_k)\ p(x_3 \mid C_k)\ \cdots \\ &\propto p(C_k) \prod_{i=1}^{n} p(x_i \mid C_k)\,. \end{aligned}$$

"Naive" conditional independence assumptions

$$\hat{y} = \underset{k \in \{1,\dots,K\}}{\operatorname{argmax}}\ p(C_k) \prod_{i=1}^{n} p(x_i \mid C_k).$$

Diving into Probability

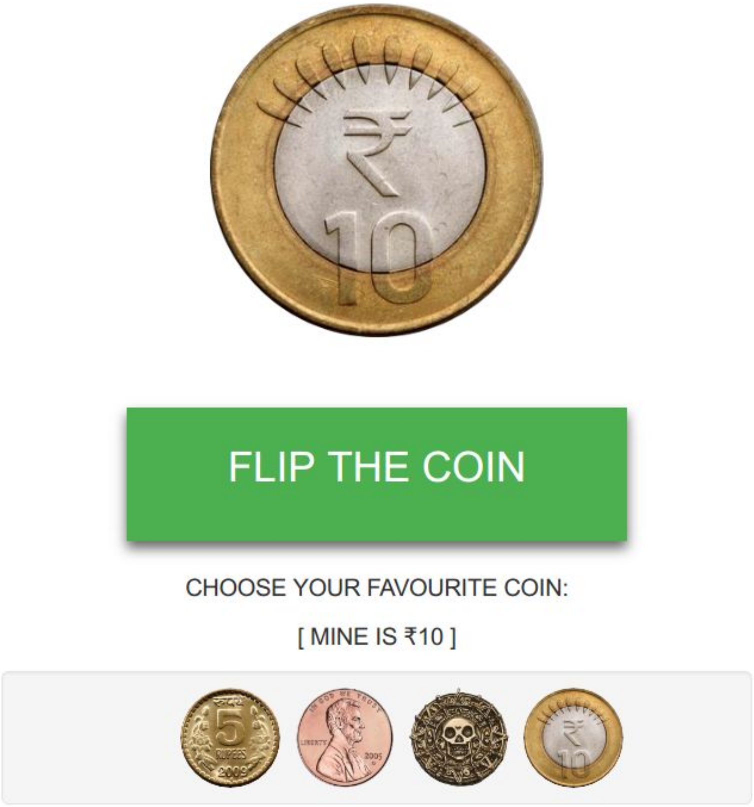
Obligatory coin toss experiment (interactive way)

Using:

1. Virtual Coin with jQuery
2. Comparing theoretical Vs experimental probability with D3
3. Simulating experiment with Python

Introduction to Probability.ipynb

#Activity 1 - Virtual Coin with jQuery



#Activity 2 - Comparing theoretical to experimental probability with

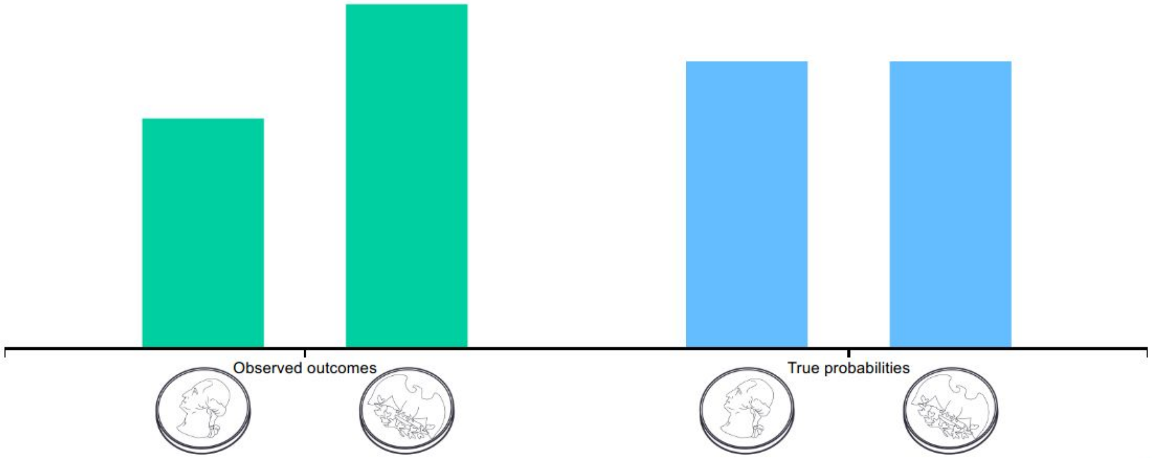
Chance Events

Randomness is all around us. Probability theory is a mathematical framework that allows us to analyze chance events in a logically sound manner. The probability of an event is a number indicating how likely that event will occur. This number is always between 0 and 1, where 0 indicates impossibility and 1 indicates certainty. A classic example of a random experiment is a fair coin toss, in which the two possible outcomes are heads or tails. In this case, the probability of flipping a head or a tail is 1/2. In an actual series of coin tosses, however, we may get more or less than exactly 50% heads.

Flip the Coin

Flip 100 times

For an unfair or weighted coin, the two outcomes are not equally likely. You can change the weight of the coin by dragging the true probability bars up or down.

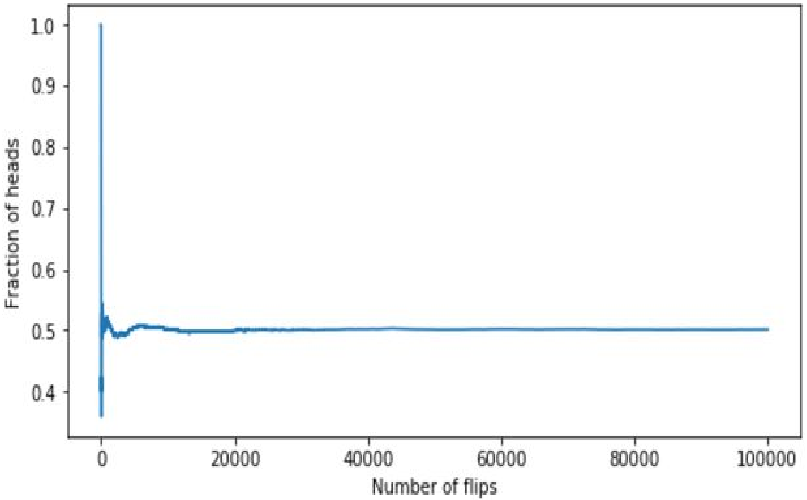


#Activity 3 - Simulating coin-toss experiment with Python

For number of coin-toss (n) = 100000

In [17]:
```python
n = 100000
heads_so_far = 0
fraction_of_heads = []
for i in range(n):
    if comp_prob_inference.flip_fair_coin() == 'heads':
        heads_so_far += 1
    fraction_of_heads.append(heads_so_far / (i+1))
```

In [18]:
```python
plt.figure(figsize=(8, 4))
plt.plot(range(1, n+1), fraction_of_heads)
plt.xlabel('Number of flips')
plt.ylabel('Fraction of heads')
```



Ingredients to Modelling Uncertainty

1. Sample space:

   The set of all possible outcomes for the experiment

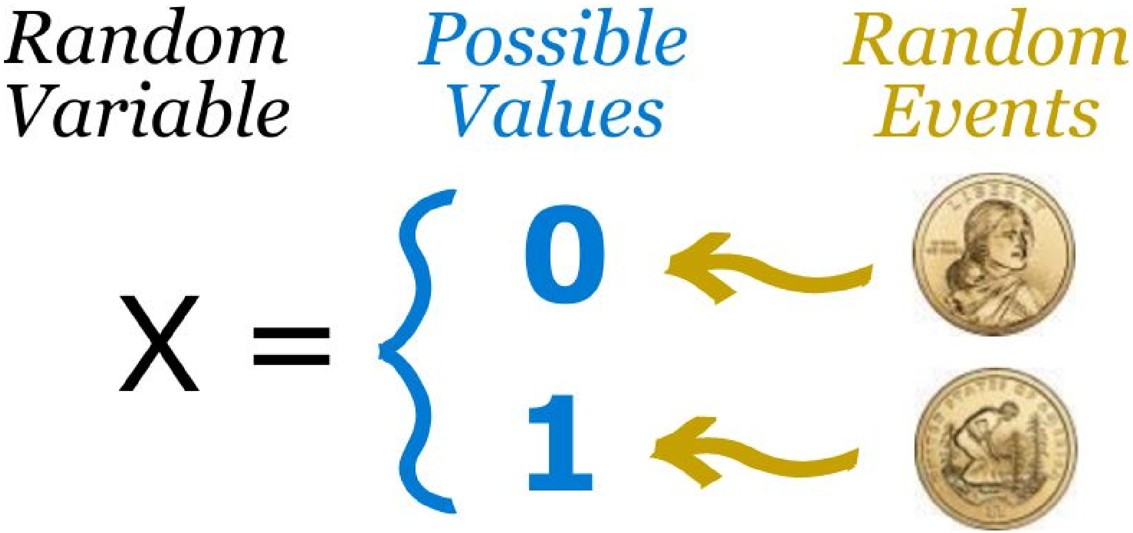
   $$\Omega = \{\text{heads}, \text{tails}\}$$

2. The probability of each outcome:

   For each possible outcome, assign a probability that is at least 0 and at most 1. For the fair coin flip:

   $$\mathbb{P}(\text{heads}) = \frac{1}{2} \text{ and } \mathbb{P}(\text{tails}) = \frac{1}{2}$$

Introduction to Random Variables



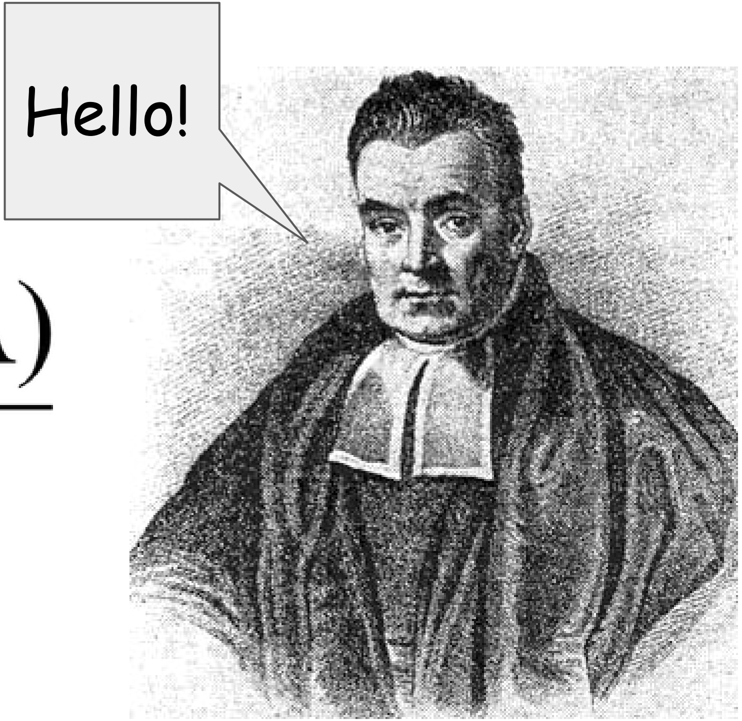
[Introducing Random Variables.ipynb]

Relation between Random Variables

1. Joint Probability
2. Marginal Probability
3. Conditional Probability
4. Dependence & Independence

Demystifying Bayes theorem

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$



Application of Probability Theory learnt so far in Machine learning

Naive Bayes Algorithm

As a Spam filter

Take Away:

- Mathematics is Hard, using programming skills it can be easy

- Intuition of Probability theory behind simple yet fast Naive Bayes algorithm

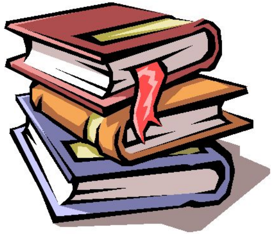
~ Thank you! ~

https://nipunsadvilkar.github.io

https://github.com/nipunsadvilkar

https://facebook.com/nipunsadvilkar

nipunsadvilkar@gmail.com



References:

1. MITx: 6.008.1x Computational Probability and Inference, Edx
2. CS 109: Probability for Computer Scientists, Stanford University
3. Prob140: Probability for Data Science, UC Berkeley
4. Mathsisfun.com
5. https://en.wikipedia.org/wiki/Naive_Bayes_classifier
6. Seeing Theory, Brown University
7. https://github.com/wintersummermint/coin-flip-javascript