ENGR-E 511; ENGR-E 399

“Machine Learning for Signal Processing”

Module 01: Lecture 01:

Probability

Minje Kim

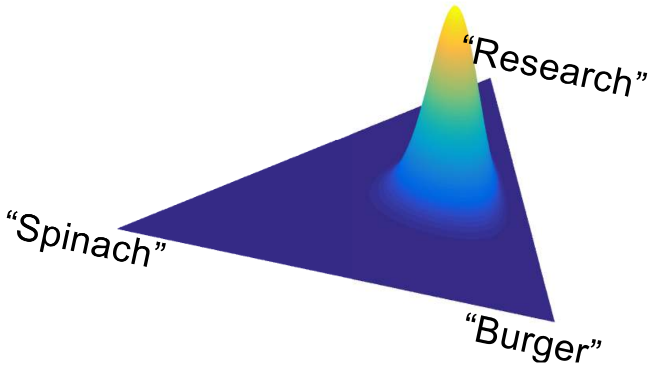
Department of Intelligent Systems Engineering

Email: minje@indiana.edu

Website: http://minjekim.com

Research Group: http://saige.sice.indiana.edu

Meeting Request: http://doodle.com/minje



INDIANA UNIVERSITY

SCHOOL OF INFORMATICS, COMPUTING, AND ENGINEERING

Basic Probability Theories

- Mean & Random Variables

- My wife has been worried about my health during my PhD study
- So, after I finished my dissertation she ordered me to lose weight
- I worked out, ate more greens, stayed away from junk food, etc.
- A month later, I weighed and reported a measurement to her
- (Of course) she didn't trust me. Why?
  - Because it was just one observation
- I had to measure my weight five more times in the following mornings, and averaged the results
  - "Are you happy now?"
- What are we doing here?
  - Empirical moments can be used as a summary of your data
  - Measurements are with error (random variable)

$$\frac{1}{5}\sum_{i=1}^{5} x_i$$

Basic Probability Theories

- Gaussian (Normal) Distribution

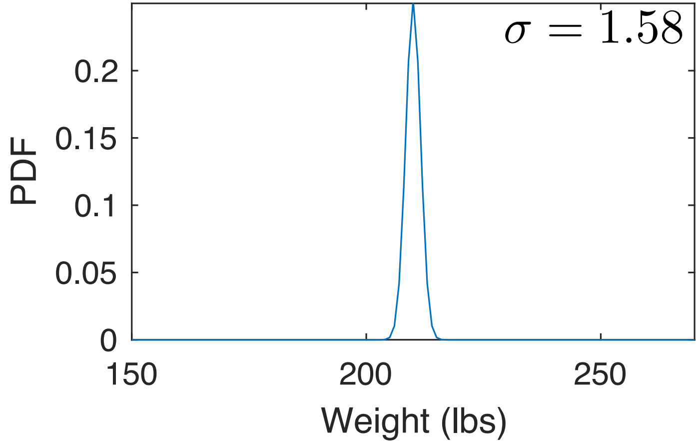
- It turned out that the average is 210 lbs
  - Is that all she wants to know?
  - Gaussian (Normal) distribution with same means can have different meanings

Normalization

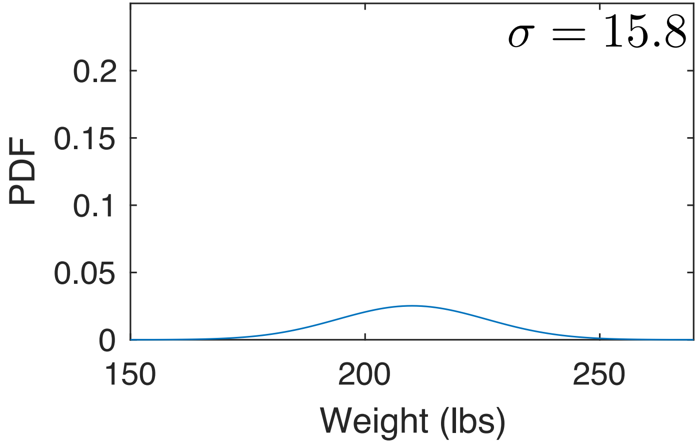
How far you are from the mean

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

How certain you are



[211, 208, 210, 209, 212]



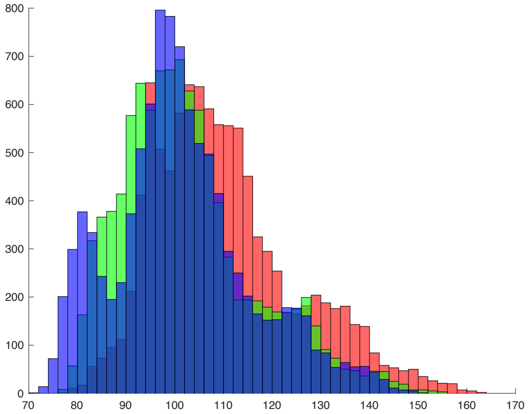
[190, 200, 210, 220, 230]

- Are we good?
  - My weights are following a Gaussian distribution?

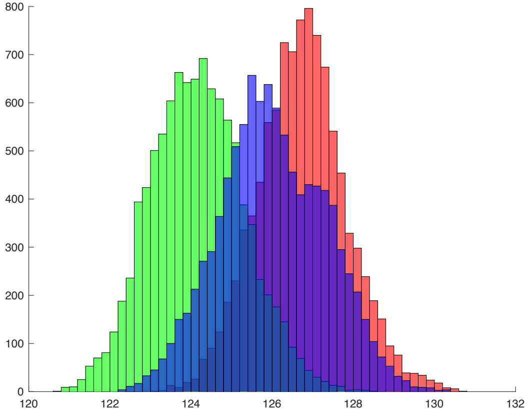
Basic Probability Theories

- Central Limit Theorem

- Sum of many i.i.d. random samples follows a Gaussian distribution
  - That's why it's the *Normal* distribution



Sum of 10 patches



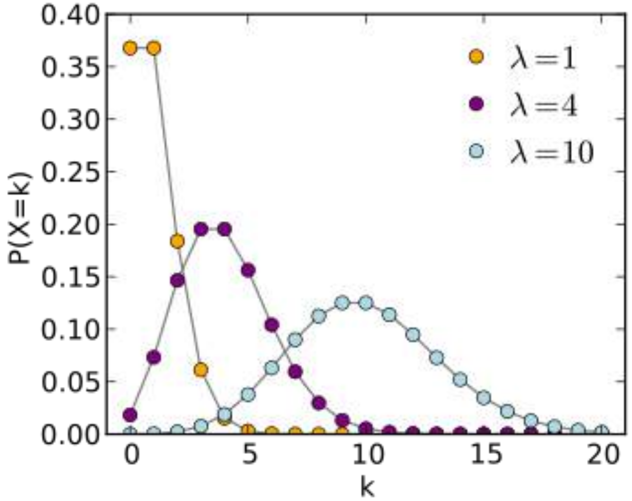
Sum of 1000 patches

Basic Probability Theories

- Poisson Distribution

- My wife is worried if I forget to measure my weight everyday
  - So, she decided to ask me at the end of the day, and count it

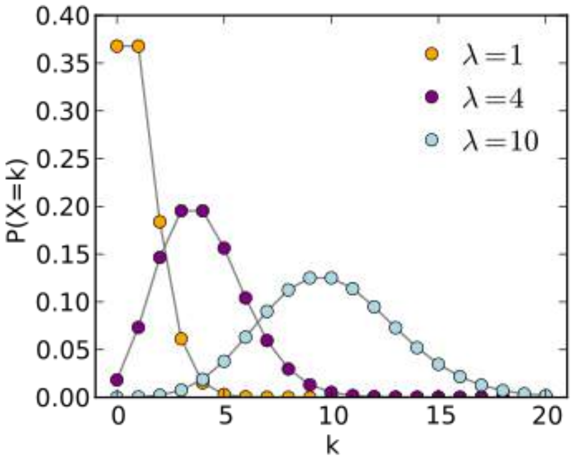
    `[1, 0, 1, 2, 1, 1, 0, 2]`
  - What's the mean/standard deviation of these counts?
    - Is this Gaussian?
- There are other distributions defined with a different set of parameters
  - i.e. Poisson distribution $$P(X=k)=\frac{\lambda^k e^{-\lambda}}{k!}$$
  - Is it different from Gaussian?
- The counts are actually following a Poisson distribution with $\lambda=1$
  - Although it's kinda unclear just by watching the samples:
    `[1, 0, 1, 2, 1, 1, 0, 2]`



https://en.wikipedia.org/wiki/Poisson_distribution

Basic Probability Theories

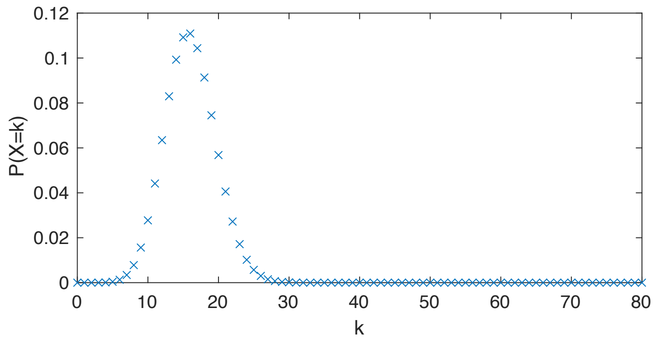
- Binomial Distribution



- She thinks that I'm not checking on my weight frequently enough
  - Because she's doing it more often:
    `[2, 3, 4, 4, 5, 5, 6, 3]`
  - Her $\lambda = 4$
- For given eight days, there were 40 measurements in the house
  - We know that only 20% of them are mine (why?)
- In the next month, we observe another set of 80 measurements in the same family of two people
  - What's the probability that 40 of them are mine?
  - What's the probability that 16 of them are mine?
- **Binomial distribution**

$$P(X=k) = \binom{N}{k} p^k (1-p)^{N-k} \qquad p = \frac{\lambda_1}{\lambda_1 + \lambda_2}$$

  - Why do we need the combination?



Basic Probability Theories

- Multinomial Distribution

- Let's say that there's another person in the family
- We need another distribution called **Multinomial distribution**

$$P(X_1 = x_1, X_2 = x_2, \cdots, X_K = x_K) = \frac{N!}{x_1! x_2! \cdots x_K!} p_1^{x_1} p_2^{x_2} \cdots p_K^{x_K} \qquad p_k = \frac{\lambda_k}{\lambda_1 + \lambda_2 + \cdots + \lambda_k}$$

- This is a very useful distribution

What's the probability of seeing the word "my" 2 times, "like" 3 times, and so on, among the 14 words?

k	p_k
I	.30
my	.20
wife	.13
today	.12
like	.10
research	.05
hate	.04
burger	.04
spinach	.02

From Probability to Machine Learning
- Maximum Likelihood

$$P(X = k) = \frac{\lambda^k e^{-\lambda}}{k!}$$

- My weighing behavior follows Poisson dist. Can we be more specific?
  - IOW, can we estimate λ from the data samples? `[1, 0, 1, 2, 1, 1, 0, 2]`
  - $P^4(X=1)\cdot P^2(X=0)\cdot P^2(X=2) = \left(\frac{\lambda e^{-\lambda}}{1}\right)^4 \cdot \left(\frac{\lambda^0 e^{-\lambda}}{1}\right)^2 \cdot \left(\frac{\lambda^2 e^{-\lambda}}{2}\right)^2 = \lambda^8 e^{-8\lambda}/4$
  - $\lambda = 1.0 : 8.3865 \times 10^{-5}$

    $\lambda = 1.5 : 3.9367 \times 10^{-5}$

    $\lambda = 3.0 : 6.1921 \times 10^{-8}$ An analytic way?

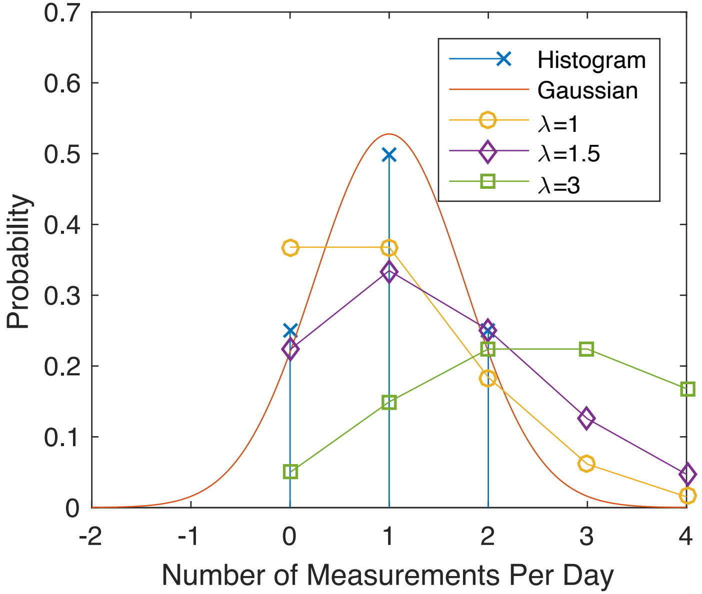
- **Maximum likelihood** estimation

$$\arg\max_\lambda \prod_i P(X = x_i) = \arg\max_\lambda \prod_i \frac{\lambda^{x_i} e^{-\lambda}}{x_i!}$$

$$\arg\max_\lambda \sum_i \ln P(X = x_i) = \arg\max_\lambda \sum_i (x_i \ln\lambda - \lambda - \ln x_i!)$$

$$= \arg\max_\lambda \ln\lambda \sum_i x_i - N\lambda$$

$$\frac{\partial \ln\lambda \sum_i x_i - N\lambda}{\partial\lambda} = \frac{\sum_i x_i}{\lambda} - N \qquad \lambda^* = \frac{1}{N}\sum_i^N x_i$$



Why the mismatch between the histogram and the ground-truth distribution?
➔Lack of data

From Probability to Machine Learning

- Independence, Conditional Probabilities

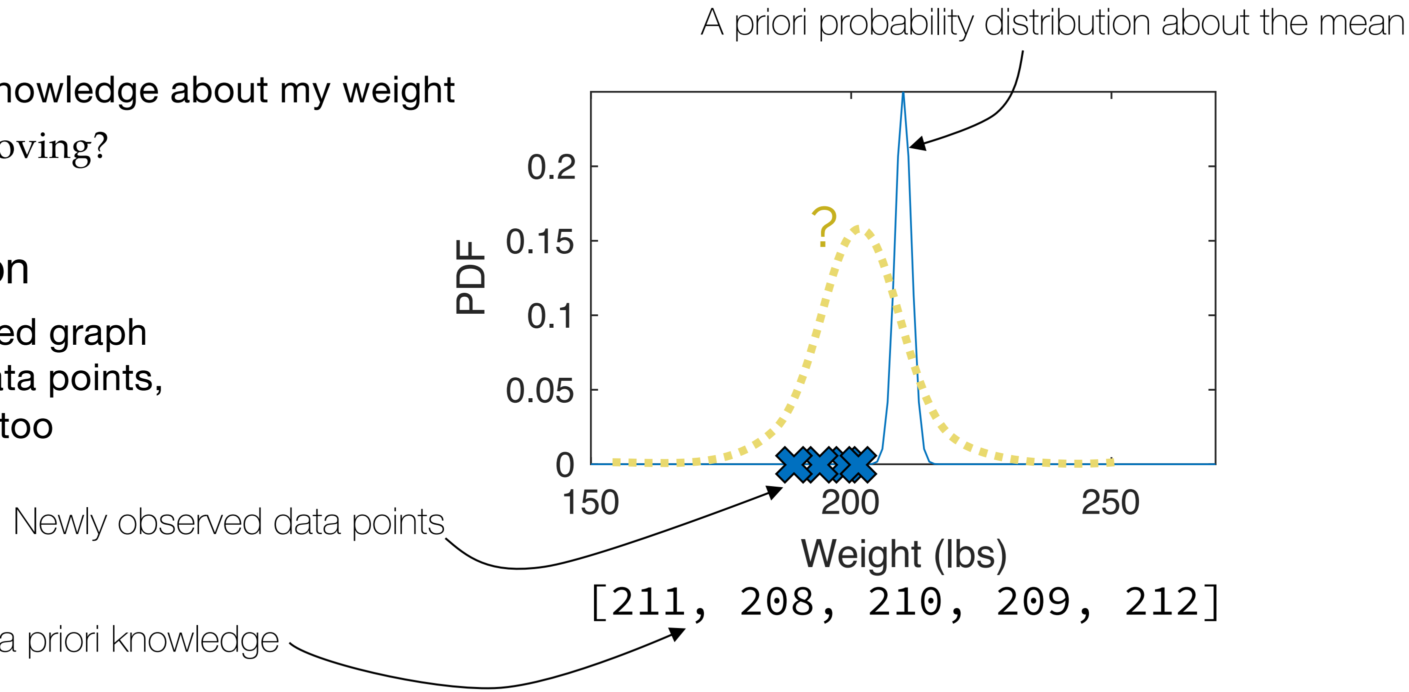
- It turned out that I lost 5 lbs for the first month (it's a true story)
  - And, I bragged
- My wife argued that (as always she's right)
  "You're taller than me, so it's easier for you to lose weight"
  1. P("weight">200lbs)=0.45
  2. P("weight">200lbs | "height">6.5ft)=?
  3. P("weight" >200lbs | "eye color"="black")=?
- The probabilities 1 and 3 are same, and we say that the weight and eye color are **independent** $0.15/0.33 \approx 0.45$
- The probability 2 and 3 are called **conditional probabilities** given the observations about the height and eye color, respectively
- P("weight" > 200lbs, "eye color"="black")
  =P("weight" > 200lbs | "eye color"="black") X P("eye color"="black")
  - What if they are independent?
  - P(200lbs < "weight" , "eye color"="black")
    =P(200lbs < "weight") X P("eye color"="black") Check if this is true in the tables!

	W<200	W>=200	P(H)
H>=6.5	0.13	0.27	0.40
H<6.5	0.42	0.18	0.60
P(W)	0.55	0.45	

	W<200	W>=200	P(EC)
EC=BK	0.18	0.15	0.33
EC=BR	0.37	0.30	0.67
P(W)	0.55	0.45	

From Probability to Machine Learning
- Maximum A Posteriori

- We moved into Bloomington after my graduation.
  After unpacking, I was asked to weigh myself immediately
  - It turned out that I finally lost 15 lbs
- Once again, she didn't trust me
  - Why is she so grumpy?
  - She has something called **a priori** knowledge about my weight
  - "What if the scale is broken while moving?
    It must be 210, not 200."
- **Maximum A Posteriori** estimation
  - We need to estimate the yellow dotted graph
    not only from the newly observed data points,
    but from the previous observations, too



From Probability to Machine Learning

- Maximum A Posteriori

- **Bayes' theorem**

$$P(\mu|[x_1, x_2, \cdots x_5]) = \frac{P([x_1, x_2, \cdots x_5]|\mu)P(\mu)}{P([x_1, x_2, \cdots x_5]|)}$$

{A Posteriori}= {Likelihood} X {A Priori} / {Normalizing Constant}

- MAP estimation:
  - To find the parameter that maximizes the **a posteriori distribution**
- We can find an analytic MAP estimation with certain conditions

$$\mathcal{LL} = \ln P([x_1, x_2, \cdots x_5]|\mu)P(\mu) = \sum_{i=1}^{5} \ln P(x_i|\mu) + \ln P(\mu)$$

$$= \sum_{i}^{5} \ln \frac{1}{\sqrt{2\pi\sigma^2}} \mathrm{e}^{-\frac{(x_i-\mu)^2}{2\sigma^2}} + \ln \frac{1}{\sqrt{2\pi\sigma_0^2}} \mathrm{e}^{-\frac{(\mu-\mu_0)^2}{2\sigma_0^2}}$$

$$= -\sum_{i}^{5} \frac{(x_i - \mu)^2}{2\sigma^2} - \frac{(\mu - \mu_0)^2}{2\sigma_0^2} + Constant$$

$$\frac{\partial \mathcal{LL}}{\partial \mu} = \sum_{i}^{5} \frac{x_i - \mu}{\sigma^2} - \frac{\mu - \mu_0}{\sigma_0^2} = \frac{\sigma_0^2 \sum_{i=1}^{5} x_i + \sigma^2 \mu_0 - (5\sigma_0^2 + \sigma^2)\mu}{\sigma^2 \sigma_0^2}$$

$$\mu^* = \frac{\sigma_0^2 \sum_{i=1}^{5} x_i + \sigma^2 \mu_0}{5\,\sigma_0^2 + \sigma^2}$$

Just another kind of average

- Does it always work like this?

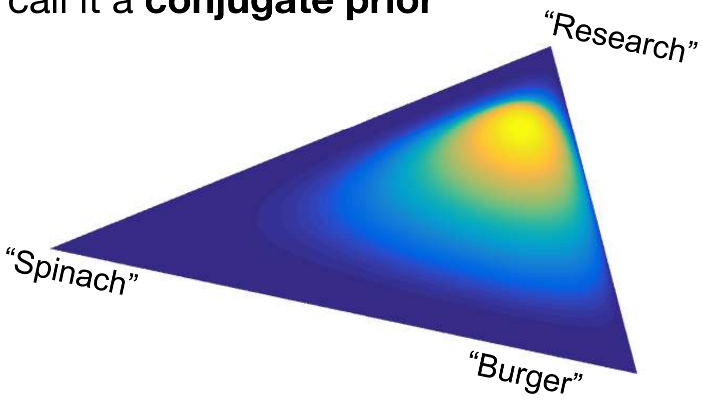
From Probability to Machine Learning

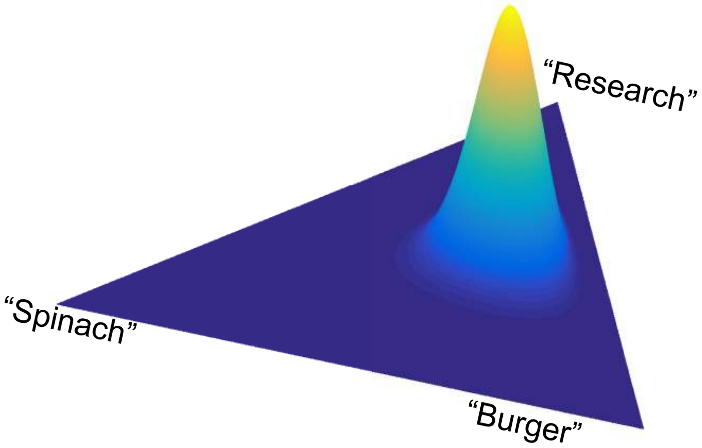
- Conjugate Priors

- If the a priori distribution has the similar algebraic form with the likelihood, we call it a **conjugate prior**
  - This ensures the posterior distributions have the same algebraic forms, too
- You observed in my tweets that I used
  - "Research": 5 times
  - "Burger": 4 times
  - "Spinach": 2 times
- What if you've watched them for a month and accumulated more evidence?
  - "Research": 19 times
  - "Burger": 15 times
  - "Spinach": 7 times
- One of these is the distribution where I sample from to decide what to talk about on the tweet on a given day
- It looks like multinomial, but it's not
  - The RV is not for the counts, but for the parameter

    $$P(\Theta_1 = p_1, \Theta_2 = p_2, \cdots, \Theta_K = p_K)$$

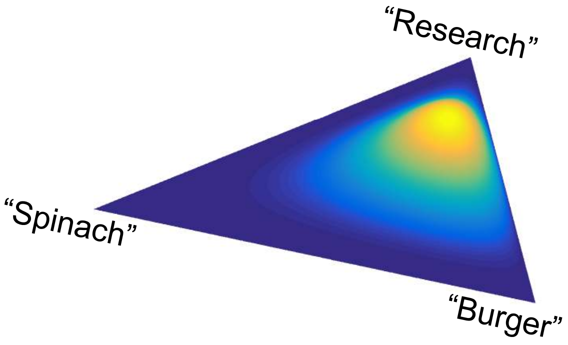
  - The counts don't have to be integers (what?)





From Probability to Machine Learning
- Conjugate Priors

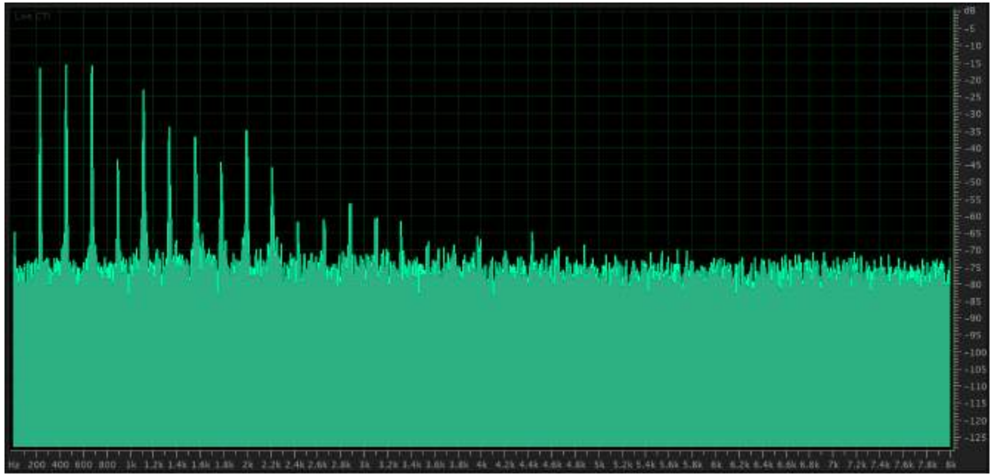
- **Dirichlet distribution**
  - $$P(\Theta_1 = p_1, \Theta_2 = p_2, \cdots, \Theta_K = p_K | \alpha_1, \alpha_2, \cdots, \alpha_K) = \frac{\Gamma\left(\sum_{k=1}^K \alpha_k\right)}{\prod_{k=1}^K \Gamma(\alpha_k)} \prod_{k=1}^K p_k^{\alpha_k - 1}$$
    $$\sum_{k=1}^K p_k = 1, \quad p_k \geq 0 \quad \forall p_k, \quad \alpha_k > 0 \quad \forall \alpha_k$$
  - We call α_k a **hyperparameter** or **pseudo count**
  - The conjugate prior of multinomial distribution
    - Because… $$P(X_1 = x_1, X_2 = x_2, \cdots, X_K = x_K | \Theta_1 = p_1, \Theta_2 = p_2, \cdots, \Theta_K = p_K)$$
      $$\cdot P(\Theta_1 = p_1, \Theta_2 = p_2, \cdots, \Theta_K = p_K)$$
      $$= \frac{N!}{\prod_{k=1}^K x_k!} \prod_{k=1}^K p_k^{x_k} \cdot \frac{\Gamma\left(\sum_{k=1}^K \alpha_k\right)}{\prod_{k=1}^K \Gamma(\alpha_k)} \prod_{k=1}^K p_k^{\alpha_k - 1}$$
      $$= \frac{N!}{\prod_{k=1}^K x_k!} \frac{\Gamma\left(\sum_{k=1}^K \alpha_k\right)}{\prod_{k=1}^K \Gamma(\alpha_k)} \prod_{k=1}^K p_k^{x_k + \alpha_k - 1}$$
      $$\propto \prod_{k=1}^K p_k^{x_k + \alpha_k - 1}$$ ←MAP is to maximize this w.r.t. $\boldsymbol{\Theta}$



Information Theory

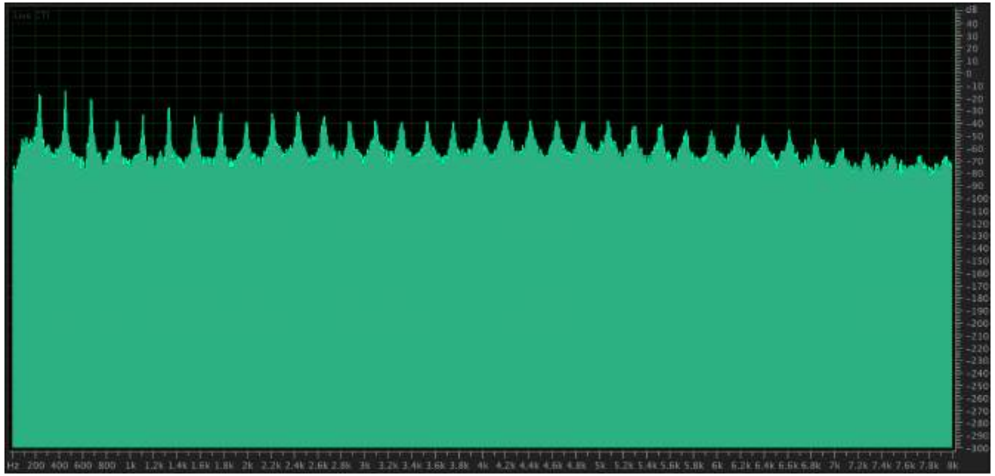
- Entropy

- Let's talk about audio a little bit
  - Flute vs. electric guitar
  - Electric guitar is noisier



Flute

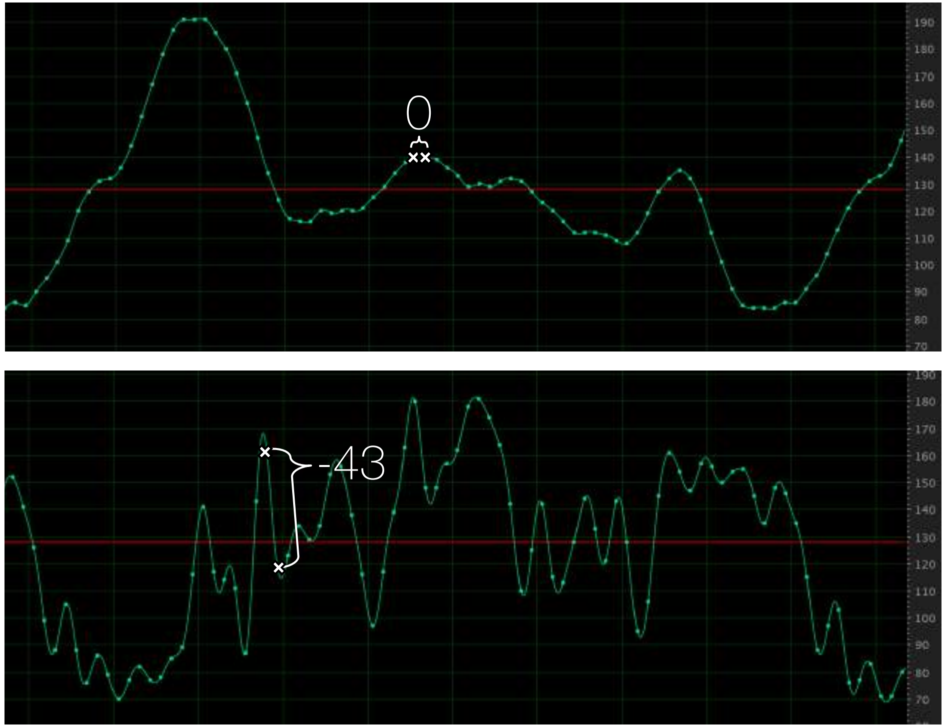
E.G.

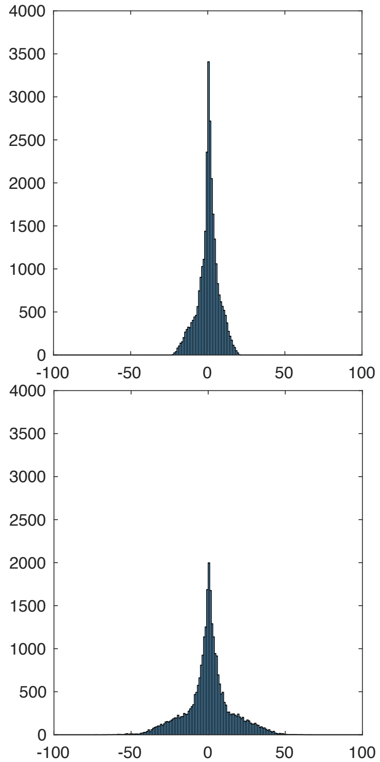


Information Theory

- Entropy

- Calculate the difference between adjacent samples
  - Flute is with a smother wave form → sharper distribution of differences
  - Electric guitar is the opposite





Information Theory

- Entropy coding

- Huffman coding (we seek minimal code length)
  - I'll assign a shorter bit string to a more frequent value
  - From the histogram of flute:
    - `100 for 0`
    - `0010001010111111111111111111111111111111111111111111111111111111111111111111111111111111111 1111111111111110 for -43`
    - From the histogram of electric guitar:
    - `0100` for 0
    - `00000111101` for -43
  - The lengths are different because -43 is rarer in flute signals
- Total number of bits used to encode the entire signal
  - Flute: `136,987`
  - E.G.: `167,732`
- Electric guitar signal spent more bits, because its **entropy** is higher

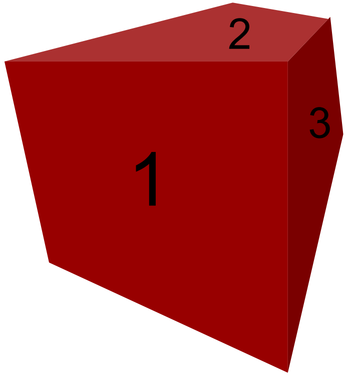
Information Theory

- Entropy coding

- The (expected) amount of information in the message
  - Represented in terms of events and their probabilities
- Less probable events have more information
  - A dice with all ones → no information
  - An unfair dice: (1, .9), (2, .05), (3, .02), (4, .015), (5, .01), (6, .005)
    - `1: 0`
    - `2: 10`
    - `3: 110`
    - `4: 1110`
    - `5: 11110`
    - `6: 11111`
  - Expected code length for an event:

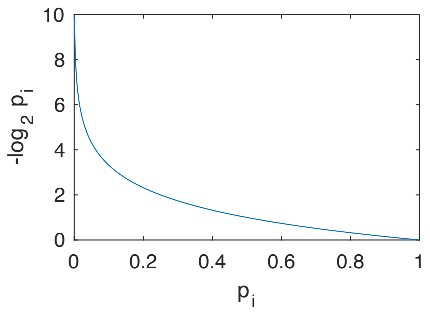
$$0.9 \times 1 + 0.05 \times 2 + 0.02 \times 3 + 0.015 \times 4 + 0.01 \times 5 + 0.005 \times 5 = 1.195 < \lceil \log_2 6 \rceil = 3$$

  - Is this the best we can do?



Information Theory

- Entropy coding



- We choose the **information function** to be the negative logarithm of the probability

$$I(x_i) = \log(1/p_i) = -\log(p_i), \quad p_i = P(x_i)$$

- $-\log(p_i)$ corresponds to the number of bits
  - More probable events carry less information (bits)
  - No information when the probability is 1
  - Events that happen at the same time carry the sum of the amount of information each of which carries, i.e.

$$-\log(p_i \cdot p_j) = -\log(p_i) - \log(p_j)$$

- If we can assign $-\log(p_i)$ bits to x_i, that's an efficient coding scheme
- (Shannon's) **Entropy** $H(X) = -\sum_i p_i \log(p_i)$
  - Lower bound of the average code length for a given distribution
  - Each distribution has a unique entropy
    - e.g. Flute's distribution of sample difference: 4.6889
    - e.g. Electric guitar: 5.7527
  - E.G. needs longer code because its distribution is less ordered

$$\frac{4.6889}{5.7527} = 0.8151$$

$$\frac{136987}{167732} = 0.8167$$

Information Theory

- Kullback-Leibler Divergence

- **Cross Entropy**
  - Calculates entropy against a different distribution
  - If p_i and q_i are same, $-\sum_i p_i \log q_i = -\sum_i p_i \log p_i$
  - Otherwise
    - For example, with a wrong $I(X)$ (two slides ago, the unfair dice example)
      $0.9 \times 1 + 0.05 \times 2 + 0.02 \times 3 + 0.015 \times 4 + 0.01 \times 5 + 0.005 \times 5 = 1.195$
    - With an optimal $I(X)$ (i.e. $-\log p_i$) , cross entropy is minimal
      $0.9 \times 0.15 + 0.05 \times 4.32 + 0.02 \times 5.64 + 0.015 \times 6.06 + 0.01 \times 6.64 + 0.005 \times 7.64 = 0.6613$
    - This starts to look like a distance (not exactly though)
- **KL Divergence**
  - The amount of information lost by an approximation
  - Frequently used to measure the difference between distributions

$$\mathcal{D}_{KL}\big(P(X)||Q(X)\big) = H\big(P(X), Q(X)\big) - H\big(P(X)\big) = -\sum_i p_i \log q_i + \sum_i p_i \log p_i = \sum_i p_i \log \frac{p_i}{q_i}$$

Thank You!