SIGMOID

Estimation Techniques: MCMC

Author Name: Arkaprova Saha

Date: 29th Dec 2022

Brief Overview of Statistical Inference

- The main idea of statistical inference is straightforward, in fact very trivial in day-to-day life.
- In real life we want to make decision about future from the experiences we had.
- In inferential statistics we assume that any phenomenon or data that we observe comes from an underlying population or theoretical distribution.
- And our target is to infer properties about the population distribution using data analysis.
- In machine learning, inference is often interchangeably used to mean, make a prediction from an already trained model.

Estimation Techniques

Some of the estimation rules or estimator are as follows

- Maximum likelihood estimators
- Bayes Estimators
- Method of moments estimators
- Cramér–Rao bound
- Least squares
- Maximum a posteriori probability (MAP)
- Minimum-variance unbiased estimator(MVUE)
- Best linear unbiased estimator(BLUE)
- Unbiased estimators
- Markov chain Monte Carlo(MCMC)

Estimation Techniques

- Estimation is a part of inferential statistics, where in order to make inference about a model, viz. the population parameters, based on the empirical data we have.
- Estimator is a rule for calculating the estimates for a population parameter.

Prediction Vs Inference

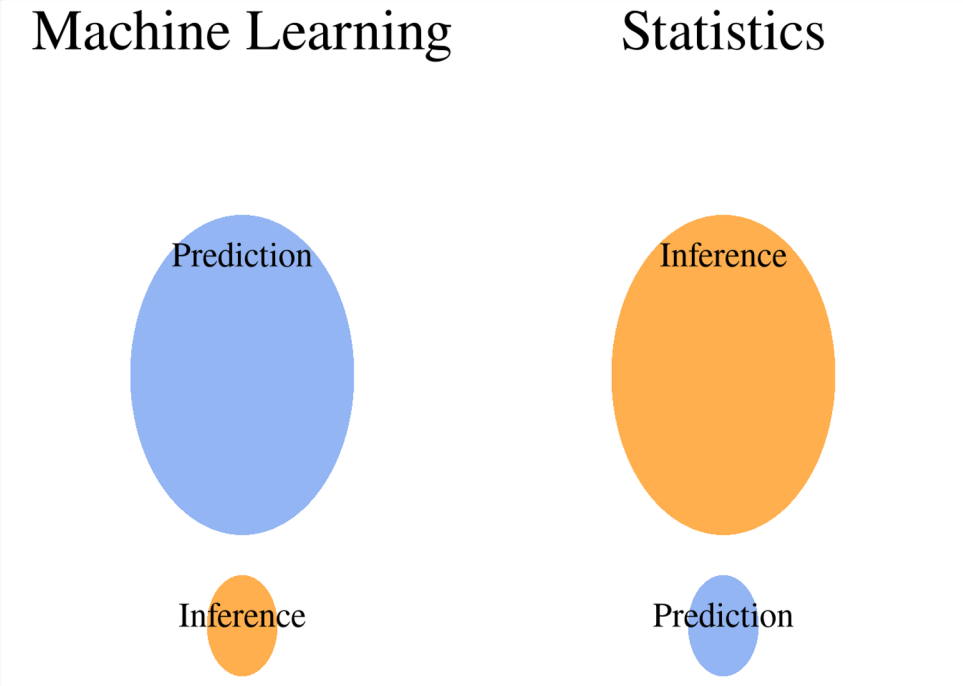
- Statistical inference is the process of drawing conclusions about the "true" properties of a phenomenon from data. Statistical inference is used to judge the individual relevance of each regressor in affecting the response of interest.
- It is slightly different from the goal of modelling for prediction. In order to do this, researchers often look for patterns in data that are likely to be meaningful, and then try to use that information to make predictions. Prediction accuracy is a key metric that researchers use to measure how well their models are doing.

For instance, consider a company that is interested in conducting a direct-marketing campaign. The goal is to identify individuals who are likely to respond positively to a mailing, based on observations of demographic variables measured on each individual. In this case, the demographic variables serve as predictors, and response to the marketing campaign (either positive or negative) serves as the outcome. The company is not interested in obtaining a deep understanding of the relationships between each individual predictor and the response; instead, the company simply wants to accurately predict the response using the predictors. This is an example of modeling for prediction.

SIGMOID

Utility of Inference and Prediction in Fields

- The difference between Inference and prediction lies in their utility mainly.
- Viz Prediction is used mostly in Machine Learning modelling.
- And in theoretical, Empirical research work, medical field where statistical significance is more important than accuracy, Inferential statistics is used vastly.



SIGMOID

Some Modern Statistical Approaches

- Increase in computing power, the reduction of its cost has revolutionized statistical inference methods. And two such methods which are more prominent in this are are namely Bootstrap and MCMC algorithms.
- These are basically numerical approaches of statistical inference, where we spend less time in understanding the underlying structure of a probability distribution of the data. And we mostly draw samples randomly from the population until desired result is met.
- Bootstrap, MCMC those are example of some such methods to name a few.

MCMC

- In Statistics Markov Chain Monte Carlo (MCMC) are methods comprising of algorithms for sampling from a probability distribution. By constructing a Markov Chain, which has the desired Steady State distribution, we can sample from a desired distribution, with minimal assumptions.
- As the name suggests there are two components Markov chain and Monte Carlo, cover both the parts briefly and why we club them together, in the next slides

Markov Chain

A **Markov chain** or **Markov process** is a Stochastic Process of a Sequence of possible events in which the probability of each event depends only on the state attained in the previous event.

Markov property

$$\Pr(X_{n+1} = x \mid X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) = \Pr(X_{n+1} = x \mid X_n = x_n),$$

Time-homogeneous Markov chain

$$\Pr(X_{n+1} = x \mid X_n = y) = \Pr(X_n = x \mid X_{n-1} = y)$$

A stationary Markov chain is a Markov chain that follows the property

$$\Pr(X_0 = x_0, X_1 = x_1, \ldots, X_k = x_k) = \Pr(X_n = x_0, X_{n+1} = x_1, \ldots, X_{n+k} = x_k)$$

Markov Chain

Transition Probability and transition matrix

The ***one-step transition probability*** is the probability of transitioning from one state to another in a single step. The Markov chain is said to be time homogeneous if the transition probabilities from one state to another are independent of time index.

$$p_{ij} = Pr\{X_n = j | X_{n-1} = i\}$$

The ***m-step transition probability*** is the probability of transitioning from state to another state in *m*-steps.

$$p_{ij}^{(m)} = Pr\{X_{n+m} = j | X_n = i\}$$

To transition from *i* to *j* in m steps, the process can first transition from *i* to *r* in m-k steps, and then transition from *r* to *j* in *m-k* steps, where $0<k<m$

$$p_{ij}^{(m)} = \sum_r p_{ir}^{m-k} p_{rj}^{k}$$

The transition matrix is basically the matrix $P = (p_{ij})$

And for the m-step in becomes

$$P^{(m)} = P^{(m-k)} P^{(k)} \quad \text{that is} \quad P^{(m)} = P \cdot P \cdot P \cdots P = P^m$$

SIGMOID

Monte Carlo Techniques

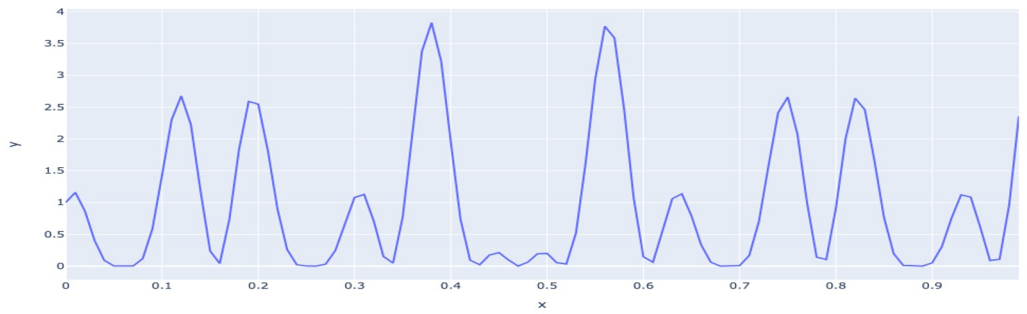
- Monte Carlo Methods are Computational Algorithms that rely on Random Sampling, in order to obtain Numerical Results.
- The underlying concept is to use randomness to solve problems that might be deterministic in principle. They are often used in mathematical problems when it is difficult or impossible to use other approaches.

- Monte Carlo sampling methods that can draw independent samples from the distribution.
- Markov Chain Monte Carlo methods draw samples where the next sample is dependent on the existing sample, called a Markov Chain, thus utilising the information from previous sample. This is why MCMC algorithms are preferable.

Example of a Monte Carlo Integration

```
x=np.arange(0,1,0.01)
def f(x):
    return ((cos(50*x)+sin(20*x))**2)
func = np.vectorize(f)
fx = func(x)

px.line(x=x,y=fx)
```

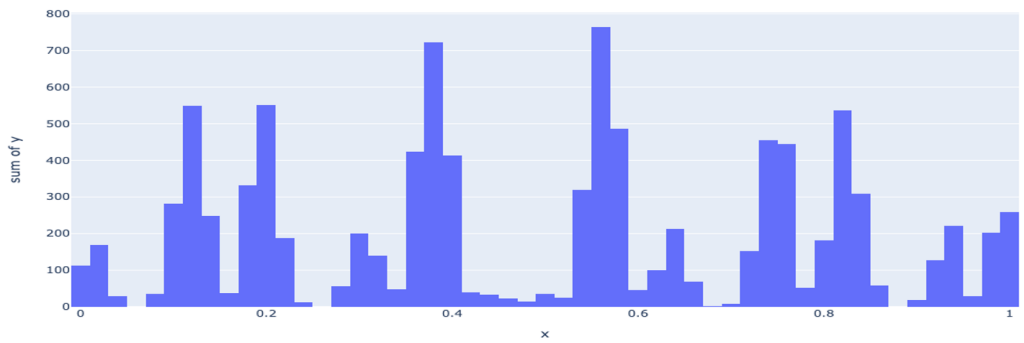
Original Function



$$\int_0^1 (cos50x + sin20x)^2 dx = 0.965$$

```
s = np.random.uniform(0,1,10000)
sampf = func(s)
px.histogram(x=s,y=sampf)
```

Monte Carlo Sample




```
[5]: value=sum(sampf)/len(sampf)

14]: round(value,2)

14... 0.98
```


SIGMOID

Markov Chain Monte Carlo

Accept Reject

Let $X_n = x$. To obtain X_{n+1}:

1. $Y \sim Q(x, .)$ and independently $U \sim \text{Unif}(0, 1)$
2. If $U < \alpha(x, y)$, then set $X_{n+1} = y$
3. Else set $X_{n+1} = x$

❖ Here α is called the acceptance function, it decides if a drawn sample to be accepted or rejected, we will next look at an important MCMC algorithm and the utility of the function α.

Markov Chain Monte Carlo

Metropolis Hastings Algorithm

Let $X_n = x$. To obtain X_{n+1}:

1. $Y \sim Q(x, .)$ and independently $U \sim \text{Unif}(0, 1)$
2. If
$$U < \min\left\{1, \frac{f(y)\, q(y, x)}{f(x)\, q(x, y)}\right\}$$
then set $X_{n+1} = y$
3. Else set $X_{n+1} = x$

The ration $r(x, y)$ is called the Hasting's ratio where

$$r(x, y) = \frac{f(y)q(y, x)}{f(x)q(x, y)}$$

Sometimes it happens that the form of a probability distribution is known only up-to a constant, viz $f(x) = c\tilde{f}(x)$ note that even in this case

$$r(x, y) = \frac{\tilde{f}(y)q(y, x)}{\tilde{f}(x)q(x, y)}$$

So in this case not knowing the constant, i.e not knowing the probability distribution explicitly does not affect the implementation of the algorithm.

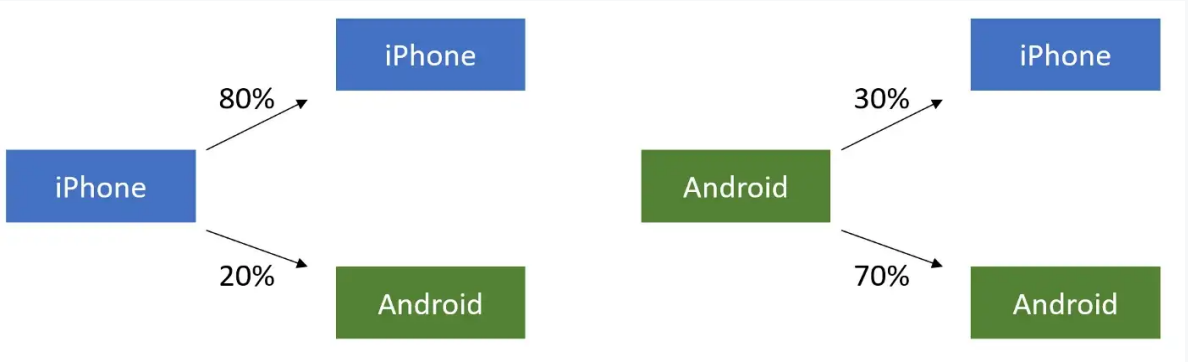
SIGMOID

Convergence of a Markov Chain

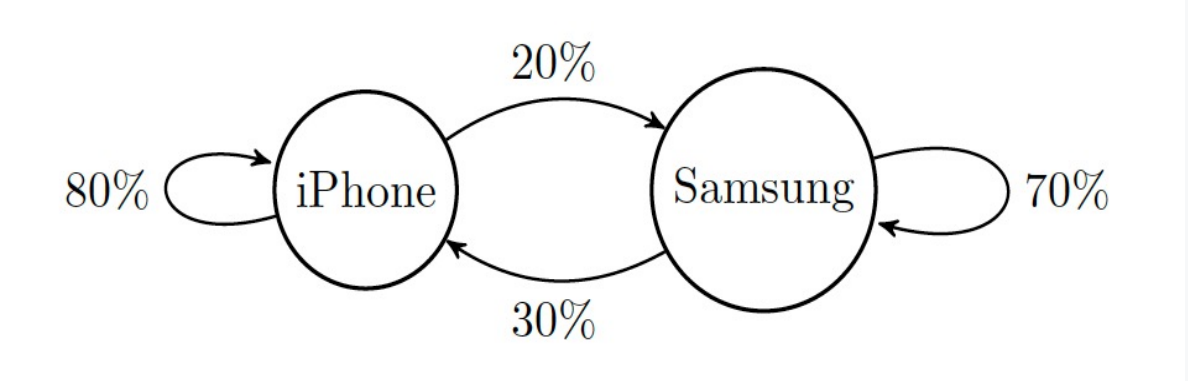
- Does Every Markov Chain Converge?
  - No only Markov Chains with certain properties converge.
  - These Markov Chains are called Ergodic Markov Chains.
- Ergodicity In Markov Chains
  - Markov chains in which any state is able to get to and from any other state eventually.
  - We can never get permanently stuck in one state or a set of states, is said to be irreducible.
  - The states can't get stuck cycling back and forth between the same set of states at regular intervals. This Markov chain is called aperiodicity.
  - When both this properties satisfy a Markov Chain is called Ergodic.

Example of a Markov Chain and its convergence

Consider the following process



The Markov Chain Diagram of the above process is following,



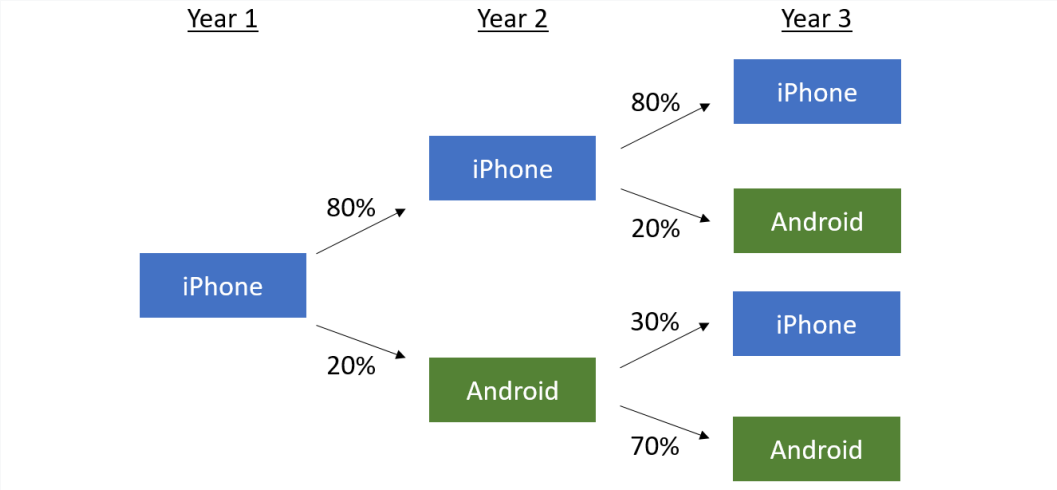
SIGMOID

Example of a Markov Chain and its convergence

So, the transition matrix with two states, viz iPhone and android is the following

$$\begin{bmatrix} 0.8 & 0.2 \\ 0.3 & 0.7 \end{bmatrix}$$

After some steps the process will start to look something like



SIGMOID

Example of a Markov Chain and its convergence

We are to find if and then where does the process converge.
The following code gives the result that the transition matrix converges to the following matrix

```
#enter code here
from sympy import MatAdd, Matrix, init_printing
P=np.array([[0.8,0.2],[0.3,0.7]])
B=P.dot(P)

err=np.inf
iter=0
while err>10E-10:
    iter+=1
    C=P.dot(B)
    err=np.linalg.norm(B-C,2)
    B=C
M=B.round(decimals=2, out=None)
print('The Limiting transition matrix is:\n\n',B.round(decimals=2, out=None),'\n')
print('The error level is:',err,'\n')
print('The transition matrix converges after ', iter, 'iterations')
```

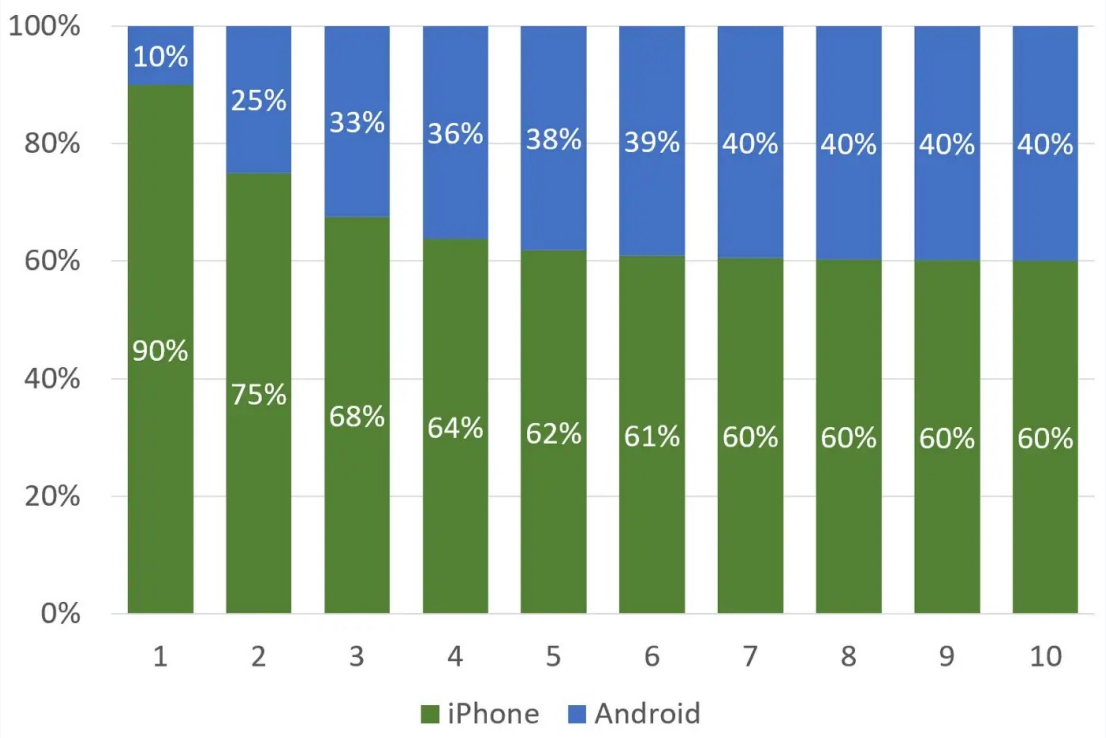
```
The Limiting transition matrix is:

 [[0.6 0.4]
 [0.6 0.4]]

The error level is: 9.497663723321688e-10

The transition matrix converges after  28 iterations
```

$$\begin{bmatrix} 0.60 & 0.40 \\ 0.60 & 0.40 \end{bmatrix}$$

SIGMOID

Convergence

We can verify it intuitively , that even if we start with 90% iPhone users and 10% Android users we will eventually converge to 60% iPhone and 40% Android users, according to the limiting distribution of the transition Matrix



SIGMOID

Convergence of the MCMC Process

- In MCMC, while sampling from an unknown distribution (say F) our purpose is to find out if the Markov Chain ultimately converge to the unknown distribution.
- In MCMC for a continuous distribution, the MTK(Markov Transition Kernel) is $P(x, A) = P(X_1 \in A | X_0 = x)$.
- For a Markov Transition kernel P the distribution F will be called invariant or stationary distribution if $FP = F$.
- $A \supset B$ where $F(B) > 0$ then for a n $P^n(x, B) > 0$, then the set A is called a F-communicating state.
- A Markov Transition Kernel is F-irreducible if for a set A such that $F(A) > 0$ for a n $P^n(x, A) > 0$

Convergence of the MCMC Process

So, for a MTK P to converge we need to ask the questions:

- Does $\lim_{n\to\infty} \|P^n(x,\cdot) - F(\cdot)\| \to 0$ for all x?
- For what n does $\|P^n(x,\cdot) - F(\cdot)\| \leq \epsilon$. So when can we say we have converged to sufficiently close to the target distribution.
- How fast does $\|P^n(x,\cdot) - F(\cdot)\| \to 0$?

The condition for a continuous MTK to converge is

If $FP = F$, P is F-irreducible and aperiodic, then for F-a.e. x

$$\lim_{n\to\infty} \|P^n(x,\cdot) - F(\cdot)\| = 0\,.$$

SIGMOID

Convergence of the MCMC Process

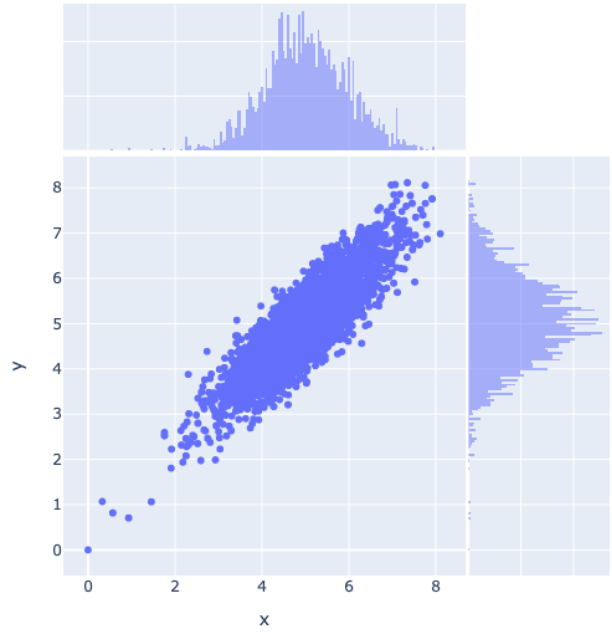
- There are different rates of convergence of Markov Chains.

1. Polynomial Ergodicity.
2. Geometric Ergodicity.
3. Uniform Ergodicity.

- Polynomial ergodicity is the weaker rate of convergence, followed by geometric ergodicity.
- If the rate of convergence is a bounded function of the starting values, then we have uniform ergodicity.

Convergence of the MCMC Process

```
mus = np.array([5, 5])
sigmas = np.array([[1, .9], [.9, 1]])

def pgauss(x, y):
    return st.multivariate_normal.pdf([x, y], mean=mus, cov=sigmas)


def metropolis_hastings(p, iter=1000):
    x, y = 0., 0.
    samples = np.zeros((iter, 2))

    for i in range(iter):
        x_star, y_star = np.array([x, y]) + np.random.normal(size=2)
        if np.random.rand() < (p(x_star, y_star) / p(x, y)):
            x, y = x_star, y_star
        samples[i] = np.array([x, y])

    return samples

samples = metropolis_hastings(pgauss, iter=10000)
fig = px.scatter(x=samples[:, 0], y=samples[:, 1],marginal_y="histogram",marginal_x="histogram",width=900, height=700)
fig.show()
```



To know more about convergence rate and convergence of MCMC follow the links below,
Geometric Ergodicity of Metropolis-Hastings Algorithms , MCMC Simulations

SIGMOID

Takeaways

- Classical statistical Methods focus on an apparently utopian idea that every data we can observe comes from a Theoretical probability distribution. In sampling terms it is called Population.
- Main idea behind statistical analysis is, after we have the data, we try to make inference about the “Ideal Theoretical Distribution” population and its parameters.
- But in reality the theoretical distribution can be complex, or impossible to know. Then MCMC or those numerical techniques come in handy, which take very less assumptions about the nature of probability distributions.
- And with the emergence of cheap computing and abundance of data to compute, these powerful statistical methods are improving inference, and data modelling altogether.

References and Links

- What is the difference between prediction and inference?
- Inference VS Prediction
- MTH707A - Markov chain Monte Carlo
- Monte Carlo Method
- Markov Chain
- Metropolis Hastings
- The Intution behind Markov Chains
- MCMC Algoritms

Thank You!

Reimagine your business with data

Email: marketing@sigmoidanalytics.com

Offices:
USA: New York, San Francisco, Dallas | Peru: Lima | India: Bengaluru