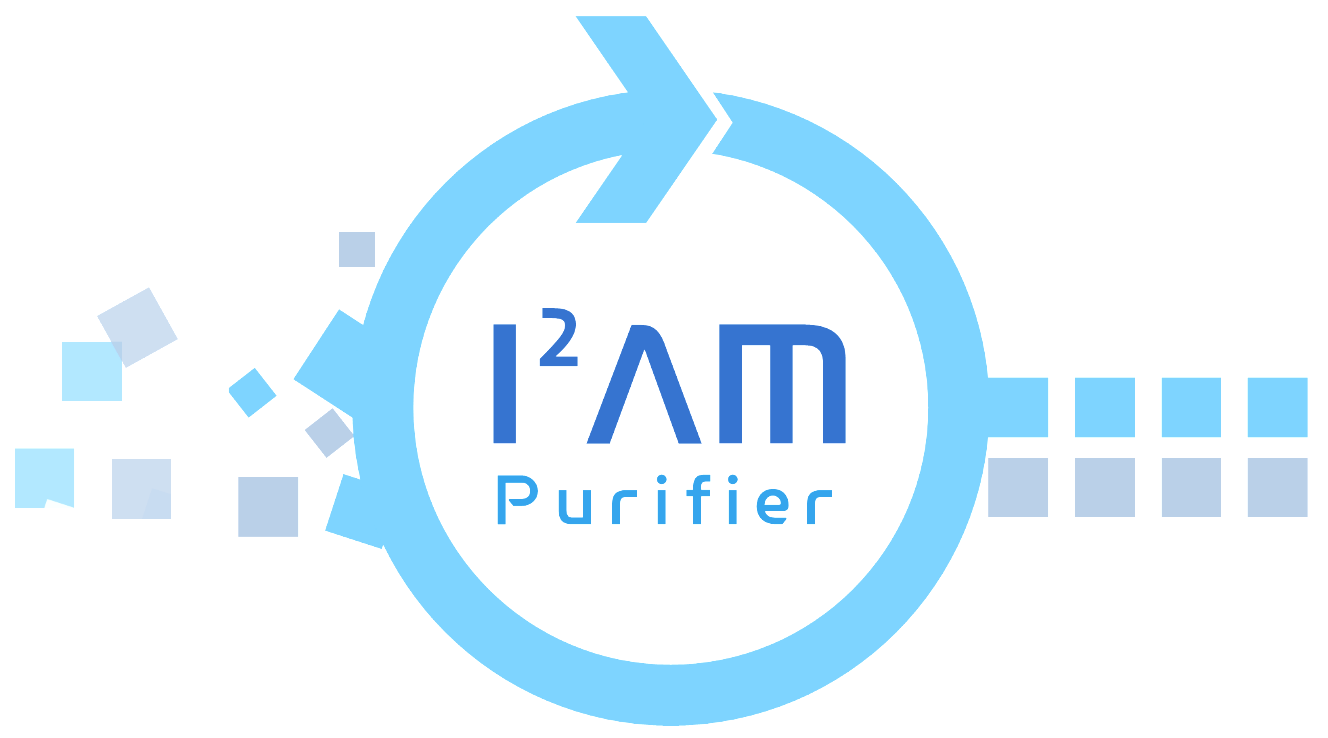
****

2016. 11.

강원대학교, 연세대학교

샘플링 기술 분석서

문서 정보

|  |  |
| --- | --- |
| **버 전** | 0.1 |
| **작성일** | 2016-11-21 |
| **상 태** | 🞎 완료 🞎 진행 중 🗹 초안 |
| **작성자** | 김하진(강원대학교) |
| **검토자** | 이름(소속) |
| **승인자** | 이름(소속) |

**목 차**

[샘플링 기술 분석서 1](file:///D:\대학원\1.연구실\2.프로젝트\IITP_I2AM\2팀\평가자료\샘플링%20기술분석서%20161110.docx#_Toc467486513)

[문서 정보 2](#_Toc467486514)

[1. 개요 1](#_Toc467486515)

[1.1. 정의 1](#_Toc467486516)

[1.2. 응용 분야 1](#_Toc467486517)

[2. 샘플링 기법 3](#_Toc467486518)

[2.1. 통계학 기반 샘플링 3](#_Toc467486519)

[2.1.1. 확률적 샘플링 3](#_Toc467486520)

[2.1.2. 비확률적 샘플링 7](#_Toc467486521)

[2.2. 스트림 환경 기반 샘플링 8](#_Toc467486522)

[2.2.1. 통계학 기반 샘플링 기법 8](#_Toc467486523)

[2.2.2. 스트림 처리에 특화된 샘플링 기법 10](#_Toc467486524)

[3. 기술 동향 13](#_Toc467486525)

[3.1. 관련 제품군 13](#_Toc467486526)

[3.2. 관련 기술연구 14](#_Toc467486527)

# 개요

## 정의

연구자가 관심을 갖고 연구하고자 하는 전체 대상 또는 집단을 모집단이라고 한다. 이런 모집단전체를 대상으로 조사하는 것을 전수조사라고 부르며, 이 방법은 데이터 분석에서 가장 이상적인 처리방법이다. 하지만 모집단은 일반적으로 매우 크기 때문에 전수조사를 할 경우 많은 시간과 비용이 요구된다는 단점이 있다. 특히 데이터가 끊임없이 유입되는 스트림 환경에서는 데이터가 무한하기 때문에 전수조사가 불가능하다. 이처럼 전수조사를 하기 어려운 경우, 모집단에 속한 데이터 일부를 샘플로 추출하고 샘플만을 사용하여 데이터를 분석한다. 이 때 모집단에서 데이터 일부를 샘플로 추출하는 것을 샘플링이라 한다.

샘플은 포괄성, 효율성, 대표성을 가져야 한다. 포괄성은 표본프레임[[1]](#footnote-1)이 모집단 전체를 포함 하는 특징이다. 효율성은 추출을 원하지 않은 요소들이 표본프레임 속에 포함되지 않는 특징이다. 대표성은 샘플의 결과가 모집단의 결과로 일반화 할 수 있는 특징이다. 샘플의 대표성을 확보하기 위해서는 샘플링 수행시 우연성을 최소화해야 하고, 샘플이 모집단과 유사한 분포를 가져야 한다. 대표성은 샘플의 가장 중요한 특징이기 때문에, 모집단을 대표할 수 있는 샘플을 분석 목적과 모집단 데이터의 특성에 맞게 추출해야 한다.

## 응용 분야

샘플링은 통계학, 스트림 환경, 주파수, 영상, 음악 등 다양한 응용분야에서 사용된다. 통계학은 샘플로부터 모집단을 추정하기 위한 학문이고, 통계학 분야에서 샘플링 기법들이 많이 연구되었다. 통계학에서는 여론조사, 불량품 비율 조사, 신약 효과 조사 등 다양한 도메인에서 모집단을 대표할 수 있는 샘플을 추출하여 분석하고 이를 토대로 통계적 가설 검정을 통해 모집단을 추정한다. 예를 들어, 신약의 효과 여부를 알고 싶다면 신약을 투약한 사람과 투약하지 않은 사람을 대표할 수 있는 샘플을 추출한 후, 통계적 가설 검정을 실시한다.

스트림 기반 샘플링은 모집단에 속한 데이터가 끊임없이 유입되는 환경에서의 샘플링을 말한다. 최근 IoT(internet of things), SNS(social network service) 등에서 데이터 스트림이 폭발적으로 증가함에 따라 스트림 기반 샘플링 기법의 필요성이 높아지고 있다. 데이터 스트림의 경우 숫자뿐만 아니라 텍스트, 로그, 멀티미디어와 같은 다양한 형태를 가지며 실시간 대용량으로 생성되기 때문에 기존 통계적 샘플링 기법들로 처리하기에는 어려움이 있다. 따라서 스트림 환경에 특화된 샘플링 기법이 필요하다.

신호처리에서 샘플링은 아날로그 데이터를 디지털 데이터로 추출하는 것을 의미한다. 아날로그 오디오가 가진 모든 파형을 디지털로 바꾸는 것은 비효율적이기 때문에, 적절한 샘플링 간격(주기)을 두고 파형을 바꿔야 한다. 이 때 나이퀴스트 샘플링(Nyquist Sampling) 이론이 사용되는데, 이는 샘플링 하려는 소리 중 가장 높은 주파수의 2배 크기의 주파수를 샘플링 간격으로 사용하면 정확하게 소리를 변환할 수 있다는 이론이다. 예를 들어, 1kHz 주파수를 디지털로 변환하고 싶다면 최소 2kHz 주파수를 샘플링 간격으로 사용하면 유실없이 변환할 수 있다.

영상처리에서 샘플링은 영상을 압축하는 방식을 말한다. 영상 데이터의 크기를 줄이기 위해 샘플링을 해야 하며, 이 때 크로마 서브샘플링(Chroma Subsampling)이 사용된다. 크로마 서브샘플링이란 명도 정보에 비해 색차 정보를 줄여 영상을 압축하는 방법이다. 사람의 눈은 밝기 변화를 느끼는 간상세보 보다 색상을 인식하는 원추세포가 적기 때문에 사람은 밝기 변화에 비해 색상 정보에 둔하다. 따라서 영상 데이터를 압축할 때 밝기 정보와 색상 정보를 분리하여 부호화하여 압축하면 효율을 높일 수 있다.

음악에서 샘플링은 기존에 있던 곡의 일부 음원을 잘라내 새롭게 가공하고 배치하는 행위를 말한다. 가장 많이 사용되는 장르는 힙합음악이지만 요즘은 다양한 장르에서 광범위하게 사용되고 있다.

이상으로 다섯 가지 샘플링의 응용분야를 설명하였다. 통계학 기반 샘플링과 스트림 기반 샘플링은 모집단을 추정하기 위해 다양한 분야의 데이터에서 모집단을 대표하는 샘플을 추출하는 방법이다. 반면 신호처리, 영상처리, 음악 분야에서의 샘플링은 특정 분야에 특화된 샘플링 기법이다. 따라서 보편적인 샘플링 기법을 소개하는 본 보고서에서는 신호처리, 영상처리, 음악 분야에서 쓰이는 샘플링 기법을 다루지 않는다.

# 샘플링 기법

본 절에서는 통계학 기반 샘플링 기법과 스트림 기반 샘플링 기법에 대해 설명한다. 통계학은 샘플로부터 모집단을 추정하는 학문으로, 샘플 추출 방법을 오랫동안 연구해왔다. 따라서 통계학 기반 샘플링 기법들은 가장 보편적인 샘플링 기법이라 할 수 있다. 스트림 기반 샘플링은 최근 데이터 스트림이 폭발적으로 증가함에 따라 스트림 환경에서 샘플링 하기 위해 연구된 기법이다.

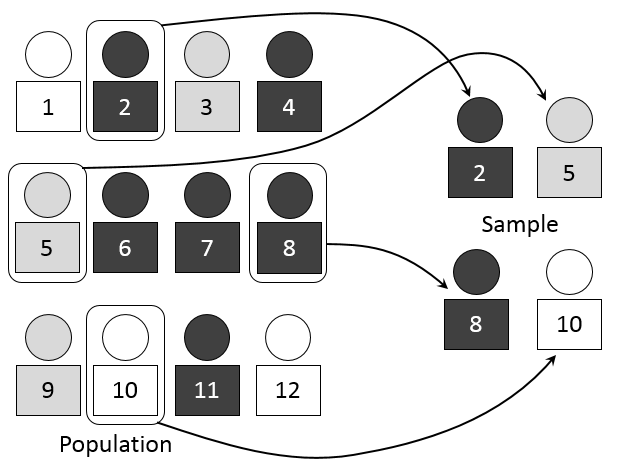
## 통계학 기반 샘플링

통계학 기반 샘플링은 확률적 샘플링과 비확률적 샘플링으로 나뉜다. 확률적 샘플링은 모집단에 속한 데이터가 샘플로 추출될 확률이 알려져 있지만, 비확률적 샘플링은 모집단에 속한 데이터가 샘플에 추출될 확률이 알려져 있지 않다. 예를 들어, 모집단에 속한 데이터가 1000개이고 이 중 100개를 샘플로 추출한다면 추출 확률은 10%로 알려지게 되고, 이는 확률적 샘플링이 된다. 반면 모집단의 속한 데이터의 수를 정확히 모를 경우 이 중 100개를 샘플로 추출한다면 추출 확률을 모르기 때문에 비확률적 샘플링이 된다. 즉, 확률적 샘플링은 모집단의 표본프레임이 확보되어, 무작위로 표본이 추출되는 방법이며 비확률적 샘플링은 모집단의 표본프레임을 확보할 수 없어 작위적으로 표본을 추출하는 방법이다. 따라서 확률적 샘플링은 샘플 분석결과의 일반화가 가능한 반면, 비확률적 샘플링은 일반화가 어렵다

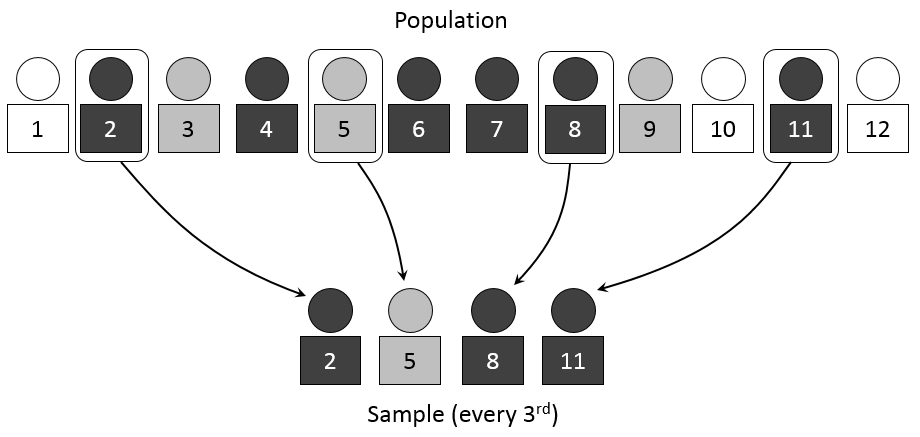
### 확률적 샘플링

확률적 샘플링은 모집단에 속한 데이터가 샘플로 추출될 확률이 알려져 있을 때 사용하는 샘플링 기법이다. 확률적 샘플링은 샘플 분석결과의 일반화가 가능하다는 장점이 있다. 확률적 샘플링은 모집단의 표본프레임이 확보되었을 때 사용되며 조사자의 주관이 개입되지 않아 샘플의 객관성이 보장된다는 장점이 있다. 확률적 샘플링에 속하는 기법은 단순무작위, 계통, 층화, 군집 등의 샘플링 있다.

단순무작위 샘플링은 무작위로 샘플을 추출하는 방법으로, 난수표를 이용해 추출하는 것이 일반적인 방법이다. 표본프레임에 속한 데이터에 각각 고유번호를 부여한 후, 난수표(table of random numbers)를 이용해 숫자를 무작위로 선택하여 표본프레임의 데이터에 부여한 고유번호와 난수표에서 선택한 숫자가 일치할 경우 그 데이터를 샘플로 추출한다. [그림 1]은 단순무작위 샘플링을 나타낸 예제이다. 그림에서, 12명의 사람 중 4명을 샘플로 추출할 경우, 표본프레임에 속한 12명에게 고유번호를 부여하고 난수표에서 숫자 4개를 무작위로 선택한다. 그림에서는 선택된 숫자가 2, 5, 8, 10이기 때문에 그 번호에 해당하는 고유번호를 가진 사람이 샘플로 추출된다. 난수는 컴퓨터로 생성하기 때문에 연구자의 자의적 개입 없이 샘플을 추출할 수 있다는 특징이 있다. 또한 모집단에 대한 사전지식이 요구되지 않고, 구현이 간단하여 샘플링 속도가 빠르다는 장점이 있다. 하지만 모집단특성을 충분히 활용할 수 없고, 샘플의 크기가 커야 대표성을 확보할 수 있으며, 동일한 크기의 샘플일 경우 층화 샘플링에 비해 표본오차[[2]](#footnote-2)가 크다는 단점이 있다.



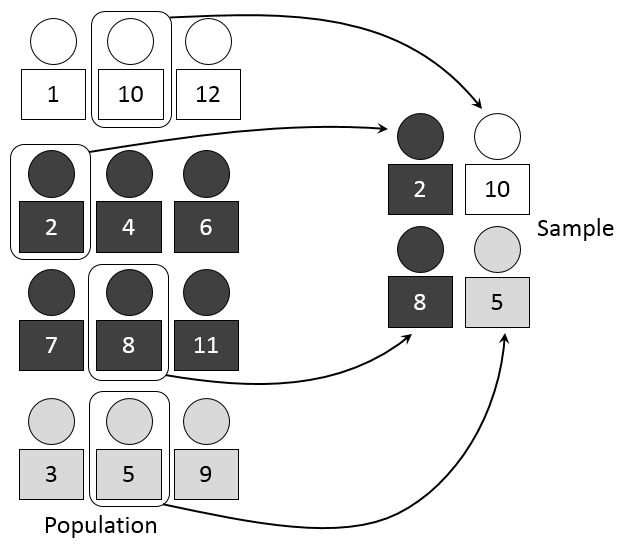
[그림 1]. 단순무작위 샘플링 예시.

계통 샘플링은 표본프레임에서 첫 번째 데이터를 무작위로 선정한 후, 매 k번째 데이터를 체계적으로 추출하는 방법이다. 이 방법은 선거에서 실시하는 출구조사(exit poll)에서 자주 사용된다. [그림 2]는 계통 샘플링을 나타낸 예제이다. 그림을 보면, 총 12명으로 구성된 모집단에서 4명의 샘플을 추출할 경우, 샘플 추출 간격 k는 전체 모집단 크기 12를 샘플크기 4로 나눈 값인 3이 된다. 다음으로 표본프레임에서 1번부터 6번 사이에서 무작위로 한 명을 선택한다. 만약 2번 대상자가 무작위로 선택됐다고 할 때, 2번 대상자부터 매 k번째(=3)에 있는 사람을 샘플로 추출한다. 여기서는 2번, 5번, 8번, ··· 에 해당하는 사람이 차례로 샘플로 추출된다. 계통 샘플링은 구현이 쉽고 비용이 적게 들고, 데이터에 따라 단순무작위보다 대표성이 좋은 장점이 있다. 하지만 모집단이 주기적인 패턴을 갖는 경우나 조사자의 주관이 개입될 경우 표본의 대표성이 현저하게 떨어지는 단점이 있다.

[그림 2]. 계통 샘플링 예시.

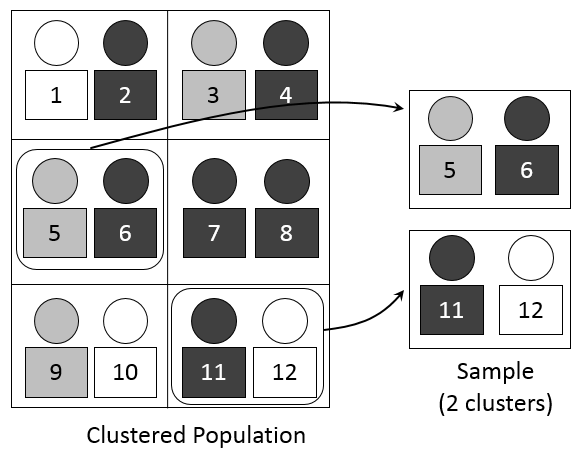
층화 샘플링은 표본프레임을 특정 기준에 따라 상이한 2개 이상의 층화집단으로 분류하고 각 소집단이 모집단에서 차지하는 비율에 따라 샘플을 무작위 추출하여 하나의 샘플을 만드는 방법이다. 각 층화집단 내부는 동질적인 속성을 갖는 반면 층화집단 간에는 이질적인 배타적 속성을 갖는다. [그림 3]은 층화 샘플링을 나타낸 예제이다. 그림에서, 모집단을 소득에 따라 고소득층(흰색), 중산층(검은색), 저소득층(회색)으로 분류했을 때, 각 층의 비율은 각각 25%, 50%, 25%이다. 만약 이 중 4명을 샘플로 추출한다면 각 소집단이 모집단에서 차지하는 비율에 따라 고소득층과 저소득층은 1명, 중산층은 2명을 샘플로 무작위 추출한다.

층화 샘플링의 장점은 샘플의 크기가 작은 경우에도 모집단의 대표성이 보장된다는 점과 이질적인 속성을 갖는 각 소집단이 모집단을 구성하는 비율에 따라 샘플을 추출하기 때문에 단순무작위 또는 계통 샘플링보다 대표성이 높다는 점, 각 소집단의 특수성을 알 수 있다는 점이다. 하지만 모집단의 각 층에 대한 정확한 정보가 필요하고, 각 층이 너무 복잡하거나 잘못될 경우 표본오차가 커질 수 있고, 시간과 비용이 많이 발생하는 단점이 있다.



[그림 3]. 층화 샘플링 예시.

군집 샘플링은 모집단을 구성하는 여러 군집들 중 무작위로 k개의 군집을 선택한 후, 그 군집 내에 있는 모든 데이터를 샘플로 추출하는 방법이다. 이 때 각 군집 내부는 이질적인 속성을 갖고 군집 간에는 동질적인 속성을 갖는 특징이 있다. 예를 들어, A 커피 전문점을 이용하는 고객들의 만족도를 연구한다고 할 때, 무작위로 특정 매장을 선정한 후 이 매장을 찾는 모든 고객들을 샘플로 추출하면 된다. [그림 4]는 군집 샘플링을 나타낸 예제이다. 그림에서는 모집단이 총 8개의 군집으로 분류되었고, 이 중 2개의 군집을 샘플로 추출하였다. 군집 샘플링은 앞서 소개한 세 가지 샘플링 기법과는 달리 표본프레임 없이도 실행할 수 있다는 특징을 갖는다. 또한 군집 샘플링은 군집 간에는 동질적인 속성을 갖는 반면, 군집 내부는 이질적인 속성을 갖기 때문에 층화 샘플링과 반대의 작동 매커니즘을 갖는다. 군집 샘플링의 장점은 광범위한 모집단에 적용 가능한 점과 상대적으로 비용이 적게 발생한다는 점이다. 하지만 군집 내의 이질성 확보가 어렵고, 다른 샘플링 기법에 비해 샘플의 크기가 커야 하며, 효율성이 증가하는 대신 대표성이 떨어진다는 단점이 있다.



[그림 4]. 군집샘플링 예시.

### 비확률적 샘플링

비확률적 샘플링은 모집단에 속한 데이터가 샘플로 추출될 추출 확률이 알려져 있지 않을 때 사용하는 샘플링 기법이다. 연구자가 임의로 샘플을 추출하기 때문에 결과의 일반화가 어렵다는 단점이 있다. 비확률적 샘플링은 모집단의 정보가 부족할 때 사용되며, 조사자의 주관이 개입될 가능성이 높지만 확률적 샘플링에 비해 시간과 비용이 적게 들고, 연구자가 모집단에 대한 사전지식을 가질 경우 효율적으로 샘플을 추출할 수 있다는 특징이 있다. 비확률적 샘플링에 속하는 기법은 편의, 판단, 할당, 스노우볼 등의 샘플링이 있다.

편의 샘플링은 길거리에서 행인들의 생각을 묻는 것과 같이 조사자가 임의로 샘플을 추출하는 샘플링 기법이다. 모집단의 추정이 불가능하고 대표성이 결여된다는 단점이 있다. 하지만 경제적이고 신속한 결과도출이 가능한 장점이 있다.

판단 샘플링은 조사자가 특정 목적에 따라 편중되게 샘플을 추출하는 방법으로, 조사 목적에 가장 적합한 데이터를 샘플로 추출한다. 예를 들어, 여행객을 대상으로 조사를 할 경우, 여행객이 많을 것으로 판단되는 공항에서 임의로 샘플로 선택하여 조사하는 것이다. 샘플 크기가 적어도 모집단의 특성을 추출할 수 있지만 모집단이 커질수록 조사자가 샘플에 대한 정확한 정보를 얻기 어려운 단점이 있다.

할당 샘플링은 모집단의 특성을 고려하여 층화집단을 선정하고 층화집단에 속한 데이터 수에 비례하게 샘플 크기를 할당한 후, 작위적으로 조사하는 방법이다. 예를 들어 고등학교, 중학교, 초등학교에 속한 대상자들의 학업성취도를 연구하는 경우, 전체 모집단에서 고등학생이 25%, 중학생이 25%, 초등학생이 50%를 차지한다면 표본프레임을 이용해 모집단을 계층에 따라 3개의 소집단으로 분류한 뒤, 모집단에 대한 각 소집단의 비율을 고려해 일정 크기의 샘플을 각 층에서 작위적으로 추출하는 것이다. 각 층화집단에서 샘플을 작위적으로 추출한다는 점이 층화 샘플링과 다른 점이다.

스노우볼 샘플링은 조사자가 작위적으로 추출한 샘플을 조사한 뒤, 응답자로부터 추천을 받아 샘플을 확대하는 과정을 되풀이하여 마치 눈덩이를 굴리듯이 누적해가며 샘플을 추출하는 방법이다. 샘플링에 소요되는 비용 및 시간이 적고, 일반화가 어렵다는 특징이 있다.

## 스트림 기반 샘플링

스트림 환경이란 IoT, 센서, SNS 등에서 끊임없이 데이터가 유입되는 환경으로, 스트림 환경에서 데이터를 처리하는 방법은 크게 배치(batch)와 실시간 스트리밍(real-time streaming) 두 가지가 있다. 첫째, 배치는 끊임없이 들어오는 데이터를 일정 시간 단위로 모은 후 처리하는 방식이다. 이 방식은 데이터(모집단)가 고정되어 있기 때문에 기존 통계학 기반 샘플링 기법을 적용할 수 있다. 하지만 데이터가 수집된 후 처리를 시작하기 때문에 실시간성이 떨어진다. 둘째, 실시간 스트리밍은 유입되는 데이터를 즉시 처리하는 방식이다. 이 경우 데이터(모집단)가 유동적이기 때문에 기존의 샘플링 기법과는 다른 스트림 환경에 적합한 샘플링 기법을 사용해야 한다. 스트림 환경에 적합한 샘플링 기법으로는 통계학 기반 샘플링 기법을 스트림 환경에 맞게 수정한 샘플링 기법과 스트림 처리를 위해 만들어진 샘플링 기법이 있다.

### 통계학 기반 샘플링 기법

본 절에서는 스트림 기반 샘플링 기법 중 통계학을 기반으로하는 샘플링 방법들을 설명한다. 통계학 기반 스트림 샘플링 기법으로 Reservoir, 계통, 해시 등의 샘플링이 있다.

Reservoir 샘플링은 단순무작위 샘플링 기법을 스트림 환경에 적용한 샘플링 기법이다. 끊임없이 유입되는 데이터 스트림 중 k개를 샘플로 단순무작위 추출하는 것이다. [그림 5]는 Reservoir 샘플링 알고리즘을 나타낸다. 라인 (1)~(3)에서는 k개의 초기 샘플을 추출한다. 다음으로, 라인 (5)에서는 k번째 이후 데이터에 대해 1부터 현재 데이터 인덱스 사이의 임의의 정수 j를 입력받는다. 그리고 라인 (6)~(8)에서는 j가 k보다 작다면 샘플의 j번째 데이터를 현재 데이터로 교체한다. 이 과정을 k번째 이후 모든 데이터에 반복 적용하여 샘플 R을 구성하고, 이를 결과로 반환한다(라인 (10)). Reservoir 알고리즘은 단순무작위 샘플링과 같이 구현이 쉽고 수행시간이 짧고 모집단에 대한 사전지식이 불필요하다는 장점이 있다. 마찬가지로 모집단의 분포를 잘 반영하기 위해서는 샘플의 크기가 커야 한다는 단점이 있다.



[그림 5]. Reservoir 샘플링 알고리즘.

계통 샘플링은 통계학 기반의 계통 샘플링을 스트림 환경에 그대로 적용한 기법이다. 모집단에 속해있는 데이터에 순번을 부여한 후 매 k번째 데이터를 체계적으로 추출하는 계통 샘플링의 특성상 끊임없이 데이터 스트림이 유입되는 스트림 환경에 바로 적용할 수 있다.

해시 샘플링은 군집 샘플링을 스트림 환경에 적용한 샘플링 기법이다. 군집 샘플링에서는 이미 나뉘어져있는 군집들 중 한 개 또는 여러 개를 샘플로 선택하는 반면, 해시 샘플링은 데이터 스트림에 해시함수를 적용하여 나온 결과 값에 따라 버킷(군집)을 나누고, 그 중 한 개 또는 여러 개의 버킷을 샘플로 선택한다. 군집 샘플링에서는 각 군집들이 군집 간에는 동질적, 군집 내부는 이질적 속성을 갖도록 군집을 나눈다. 하지만 해시샘플링에서는 각 데이터 스트림의 해시 함수 결과 값에 따라 버킷을 나누기 때문에 데이터의 특성에 따라 버킷 간 동질적, 버킷 내부는 이질적 속성을 갖기 어려울 수 있다. 예를 들어, 로그데이터와 같이 (id, value)쌍으로 된 데이터를 해시 샘플링 할 경우, id를 해시 함수의 인자로 설정하면 동일한 id를 갖는 모든 로그데이터는 동일한 해시 함수 결과 값을 갖게 되고, 따라서 모두 같은 버킷에 속하게 된다. 특정 사용자의 로그데이터가 지나치게 많을 경우, 버킷 간 이질적 속성을 갖게 되고 어떤 버킷을 샘플로 선택하느냐에 따라 분석 결과가 크게 달라질 수 있다. 따라서 분석 목적에 맞는 샘플을 추출할 수 있도록 해시 함수의 인자를 설정해주어야 한다. [그림 6]은 해시 샘플링의 알고리즘을 나타낸다. 라인 (1)~(3)에서는 모든 입력 데이터 스트림에 해시 함수를 적용하고, 해시 결과에 해당하는 버킷으로 분류한다. 그리고 라인 (4)에서는 사용자가 추출을 원하는 버킷을 샘플 결과 R로 선택하고 이를 결과로 반환한다. 해시 샘플링은 구현이 쉽고 수행속도가 빠르다는 장점이 있지만 데이터의 특성에 따라 버킷 간 동질성이 보장되지 않을 수 있다는 단점이 있다.



[그림 6]. 해시샘플링 알고리즘.

### 스트림 처리에 특화된 샘플링 기법

본 절에서는 스트림 기반 샘플링을 위해 새롭게 연구된 샘플링 기법들을 설명한다. 스트림 처리에 특화된 샘플링 기법으로는 우선순위 샘플링과 윈도우 기반 샘플링 등이 있다.

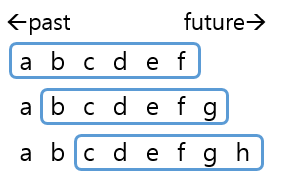
우선순위 샘플링은 가중치와 함께 유입되는 데이터 스트림에 대한 샘플링 방법이다. 데이터와 함께 유입되는 가중치를 바탕으로 우선순위를 구하고, 우선순위가 높은 k개의 데이터를 샘플로 추출하는 방법이다. 가중치는 해당 데이터 스트림의 출현 빈도이며, 우선순위는 가중치에 0과 1사이의 난수를 제곱한 값이다. [그림 7]은 우선순위 샘플링 알고리즘을 나타낸다. 라인 (1)~(4)에서는 k개의 초기 샘플을 우선순위 p와 함께 추출한다. 라인 (6)에서는 k번째 이후 데이터에 대해 우선순위를 계산한다. 그리고 만약 현재 데이터의 우선순위 p가 샘플의 최소 우선순위보다 크다면(라인 (7)), 해당 데이터와 현재 데이터를 교체한다(라인 (8)~(9)). 이와 같은 과정을 모든 데이터에 반복 적용하여 샘플 R을 구성하고 이를 결과로 반환한다(라인 (13)). 가중치가 높은 데이터에는 우선순위를 높게, 가중치가 낮은 데이터에는 우선순위를 낮게 하여 가중치가 높은 데이터가 비교적 샘플에 많이 추출된다. 우선순위 샘플링은 가중치를 저장한 배열을 유지해야 하고, 끊임없이 유입되는 데이터에 대한 우선순위를 빠르게 계산해야 하기 때문에 시스템 자원이 많이 필요하고 실행속도가 느리다는 단점이 있다. 하지만 앞서 설명한 세 가지 샘플링 기법보다 모집단과 유사한 샘플을 추출한다는 장점이 있다.



[그림 7]. 우선순위 샘플링 알고리즘.

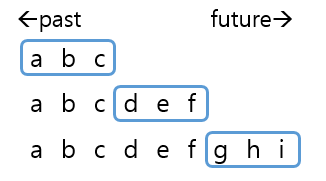
윈도우 기법은 스트림 데이터를 처리하기 위해 사용하는 스트림 데이터 처리 기법 중 하나이다. 데이터 스트림은 끊임없이 유입되기 때문에 모집단은 계속 증가한다. 따라서 계속 증가하는 모집단에서 샘플을 추출할 수도 있지만, 최근 유입된 데이터가 더 높은 중요도를 가지고 있을 경우, 모집단을 최근 유입된 데이터로 한정시킬 필요가 있다. 이 때 모집단을 어떤 방법으로 한정시킬 것인지가 중요하고, 이 때 사용되는 방법이 윈도우 기법이다. 윈도우 기법은 특정 간격(윈도우)에 속한 데이터 스트림만 처리하는 기법이다. 윈도우에 속한 데이터를 모두 샘플로 추출할 수도 있고, 앞에서 설명한 샘플링 기법들을 사용하여 윈도우에 속한 데이터 중 일부를 샘플로 추출할 수도 있다. 윈도우 기법은 윈도우가 이동하는 단위에 따라 슬라이딩 윈도우와 디스조인트 윈도우로 나뉜다.

슬라이딩 윈도우는 윈도우가 데이터 단위로 이동하는 윈도우이다. [그림 8]은 슬라이딩 윈도우를 나타낸 그림이다. 그림에서 윈도우가 이동하는 단위는 데이터 1개이고, 윈도우의 크기는 6이다. 데이터 1개가 유입되면 윈도우는 최신 데이터를 유지하기 위해 데이터 1개만큼 이동한다.



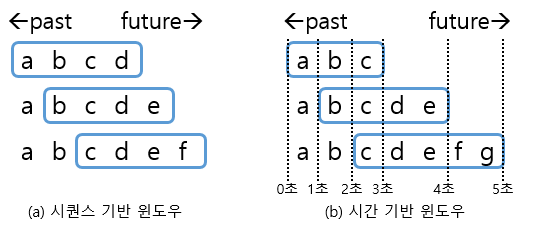
[그림 8]. 슬라이딩 윈도우.

디스조인트 윈도우는 윈도우가 윈도우 단위로 이동하는 윈도우이다. 그림 9는 디스조인트 윈도우를 나타낸 그림이다. 그림에서 윈도우가 이동하는 단위는 윈도우 1개이다. 그림에서 윈도우 크기는 3이고, 새로운 데이터가 윈도우 크기인 3개만큼 들어오면, 윈도우가 이동한다.



[그림 9]. 디스조인트 윈도우.

윈도우는 윈도우 크기를 결정하는 방법에 따라 시퀀스 기반 윈도우와 시간 기반 윈도우로 나뉜다. 시퀀스 기반 윈도우는 앞서 설명한 예제와 같이 윈도우 크기를 데이터의 개수로 설정하고, 시간 기반 윈도우는 윈도우 크기를 시간으로 설정한다. [그림 10]은 시퀀스 기반 윈도우와 시간 기반 윈도우를 슬라이딩 윈도우에 적용한 예를 나타낸 그림이다. [그림 10(a)]는 크기 4인 시퀀스 기반 윈도우이다. 윈도우는 최근 유입된 데이터 4개로 구성된다. 새로운 데이터가 하나씩 유입될 때 윈도우가 이동하는 것을 알 수 있다. [그림 10(b)]는 크기 3초인 시간 기반 윈도우이다. 윈도우는 최근 3초동안 유입된 데이터로 구성되기 때문에 절대적인 크기를 갖지 않는다. 시퀀스 기반 윈도우와 시간 기반 윈도우는 디스조인트 윈도우에도 똑같이 적용될 수 있다.



[그림 10]. 시퀀스 기반 윈도우와 시간 기반 윈도우

윈도우 기법을 사용하면 끊임없이 유입되는 데이터 중 최신 데이터를 모집단 또는 샘플로 사용할 수 있다는 장점이 있다. 윈도우에 속한 데이터 중 일부를 다양한 샘플링 기법을 활용해 샘플을 추출할 수도 있고, 윈도우에 속한 데이터 그 자체를 샘플로 추출하여 사용할 수도 있다. 하지만 윈도우 크기 설정, 윈도우를 벗어난 데이터 처리 등은 향후 연구로 남아있다.

# 기술 동향

본 절에서는 샘플링 기법의 기술 동향에 대해 설명한다. 어떤 제품이 어떤 샘플링 기법을 제공하는지에 대해 설명하고 샘플링 기법에 대한 기술 연구에 대해 설명한다.

## 관련 제품군

본 절에서는 샘플링 기법을 제공하는 다양한 관련 제품들을 소개한다. SPSS는 IBM사가 개발한 빅데이터 분석 소프트웨어로, 통계 분석 및 데이터 마이닝에 사용된다. SPSS는 단순무작위, 층화, 군집, 다단계[[3]](#footnote-3) 샘플링 기법을 제공한다.

SAS는 빅데이터 분석 플랫폼 개발 기업으로 다양한 분석 소프트웨어를 개발하였다. 이 중 SAS Enterprise Miner는 데이터 분석 모델을 생성하는 소프트웨어로 단순무작위, 층화, 군집, 계통 등 다양한 샘플링 기법을 제공한다.

R은 뉴질랜드 오클랜드 대학의 로버트 젠틀맨과 로스 이하카가 시작하여 현재 R 코어 팀이 개발하고 있는 통계 계산과 시각화를 위한 프로그래밍 언어이자 소프트웨어 환경이다. R은 통계 소프트웨어 개발과 자료 분석에 사용되고 있으며, 패키지 개발과 사용이 용이하다는 장점이 있다. R에서는 특정 샘플링 알고리즘을 함수로 구현하여 사용할 수도 있고, 다른 사람이 개발한 패키지를 다운받아 사용할 수도 있다. 현재 R은 기본 함수로 단순무작위 샘플링을 제공하고, 군집샘플링 등 다양한 통계적 샘플링 함수가 있는 ‘Sampling’ 패키지를 제공한다.

에스퍼는 CEP(complex event processing)을 구현한 오픈소스 DSMS(data stream management system)이다. 에스퍼에서는 데이터 스트림의 실시간 처리를 위해 EPL(event processing language)을 사용한다. 즉, 에스퍼 환경에서 샘플링 기법들을 사용하기 위해서는 샘플링 알고리즘을 EPL로 구현해야 한다.

오라클 Complex Event Processing은 오라클사에서 개발한 CEP이다. CEP엔진이 경량 서버이고, SQL의 확장판인 오라클 CQL(continuous query language)을 사용하여 사용이 용이하다는 특징이 있다. 오라클 CEP에서는 CQL에 샘플링 알고리즘을 등록하여 사용할 수 있다.

## 관련 기술연구

본 절에서는 샘플링 기법에 대한 기술 연구에 대해 설명한다. Babcock 외 2명은 데이터가 끊임없이 유입되는 스트림 환경에서 윈도우가 이동할 때의 샘플링 기법을 연구하였다.[[4]](#footnote-4) Reservoir 샘플링에서 샘플에 포함된 데이터가 윈도우에서 제외되었을 경우를 처리하는 “chain-sample”과 윈도우 내 데이터의 개수가 유동적인 시간 기반 윈도우에서 공간복잡도가 을 넘지 않는 우선순위 샘플링을 개발하였다.

Nick Duffield 외 2명은 가중치와 함께 데이터가 유입되는 스트림 환경에서, 임의의 샘플 부분집합의 가중치 합을 대략적으로 추정할 수 있는 우선순위 샘플링을 개발하였다.[[5]](#footnote-5) 이 기법을 네트워크 트래픽 분석에 적용해보면, 라우터가 패킷의 흐름을 요약하여 저장할 수 있다.

Mohammed Al-Kateb 외 2명은 스트림 환경에서 Reservoir 샘플링을 수행할 때, 샘플 크기를 고정시키지 않고 데이터 특성과 샘플링 상황에 따라 샘플 크기를 적절하게 결정하는 multi-reservoir 샘플링 샘플링 알고리즘을 개발하였다.[[6]](#footnote-6) Reservoir 샘플링 수행 도중 적절한 샘플 크기를 정하는 이론 연구를 수행한 후, 실제 센서 네트워크 데이터셋을 이용해 경험적으로 실험하였다.

Graham Cormode 외 3명은 데이터 스트림이 분산 환경에서 유입될 때 Reservoir 샘플링을 적용하는 방법을 연구하였다.[[7]](#footnote-7) 이 연구에서는 분산 스트림 환경에서 통신 오버헤드를 줄인 프로토콜을 제안하였다.

1. 모집단 내에 포함된 데이터(조사대상들)의 명단 리스트. [↑](#footnote-ref-1)
2. 모집단 중 일부만 조사함에 따라 발생하는 오차로 샘플이 모집단을 얼마나 잘 설명하는지 나타내는 척도. [↑](#footnote-ref-2)
3. 여러 번 샘플링을 하여 최종 샘플을 추출하는 기법 [↑](#footnote-ref-3)
4. B. Babcock, M. Datar, and R. Motwani, “Sampling from a moving window over streaming data.” In *Proc. of the 13th annual ACM-SIAM symposium on Discrete algorithms,* San Francisco,pp. 633-634, Jan. 2002. [↑](#footnote-ref-4)
5. N. G. Duffield, C. Lund, and M. Thorup. "Priority sampling for estimation of arbitrary subset sums", *Journal of the ACM, vol.* 54, No.32, Dec. 2007. [↑](#footnote-ref-5)
6. M. Al-Kateb, B. S. Lee, and X. S. Wang, "Adaptive-size reservoir sampling over data streams." *Scientific and Statistical Database Management,* IEEE, July 2007. [↑](#footnote-ref-6)
7. G. Cormode, et al. "Optimal sampling from distributed streams." *Proceedings of the twenty-ninth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*. ACM, 2010. [↑](#footnote-ref-7)