2. 데이터와 포본분포

빅데이터 시대가 되면서 더는 표본추출(표집, 샘플링)이 필요 없을 거라고 오해하는 사람들이 많다. 하지만 데이터의 질과 적합성을 일정 수준 이상으로 담보할 수도 없으면서 데이터 크기만 늘어나는 것이 오늘날 상황이다. 이런 상황에서, 오히려 다양한 데이터를 효과적으로 다루고 데이터 평향을 최소화하기 위한 방법으로 표본추출의 필요성이 더 커지고 있다. 아무리 빅데이터 프로젝트라고 해도, 결국 작은 표본 데이터를 통해 예측 모델을 개발하고 테스트한다. 샘플은 다양한 종류의 테스트에도 사용된다.

전통적인 통계학에서는 강력한 가정에 기초한 이론을 통해 모집단을 밝혀내는 데 초점을 맞춰왔다.

하지만 현대 통계학에서는 이러한 가정이 더 이상 필요하지 않은 연구로 방향이 옮겨지기 시작했다.

일반적으로 데이터 과학자들은 모집단의 이론적 측면에 대해 신경쓰기 보다는, 대신 표본추출 과정과 주어진 데이터에 집중한다. 몇 가지 주목할 만한 예외는 있다. 때론 모델링이 가능한 물리적 과정을 통해 데이터가 생성되기도 한다.

가장 간단한 예는 동전을 뒤집는 것이다. 이는 이항분포를 따른다. 실제로 생활에서 겪게되는 모든 이항 상황(구매하거나 구매하지 않거나, 사기이거나 아니거나 등등)은 이 동전 뒤집기로 설명이 가능하다. 이 경우 우리는 모집단에 대한 이해를 바탕으로 추가적인 통찰을 얻을 수 있다.

2.1 랜덤표본추출과 표본편향

용어정리

임의표집(랜덤표본추출) : 무작위로 표본을 추출하는 것

층화표집(층화표본추출) : 모집단을 층으로 나눈 뒤, 각 층에서 무작위로 표본을 추출하는 것

단순임의표본(단순랜덤 표본) : 모집단 층화 없이 랜덤표본추출로 얻은 표본

표본편향 : 모집단을 잘못 대표하는 표본

랜덤표본추출 은 대상이 되는 모집단 내의 선택 가능한 원소들을 무작위로 추출하는 과정을 말하며, 각 추첨에서 모든 원소는 동일한 확률로 뽑히게 된다. 그 결과 얻은 샘플을 **단순랜덤표본**이라고 한다.

추첨 후, 다음 번에도 중복 추출이 가능하도록 해당 샘플을 다시 모집단에 포함시키는 **복원추출**을 통해 표본을 얻을 수 있고, 아니면 한번 뽑힌 원소는 추후 추첨에 사용하지 않는 **비복원추출**을 할 수도 있다.

샘플 기반의 추정이나 모델링에서 데이터 품질은 데이터 양보다 훨씬 중요하다. 데이터 과학에서 데이터 품질이란 완결성, 형식의 일관성, 깨끗함 및 각 데이터 값의 정확성을 말한다. 통계는 여기에 **대표성**이라는 개념을 추가한다.

편향된 데이터를 **표본편향**이라 한다. 원래 대표되도록 의도된 모집단으로부터 추출되지 않고 유의미한 비임의 방식으로 표본이 추출 된 것이다. **비임의**라는 용어는 아주 중요하다. 아무리 랜덤표본이라고 해도 어떤 표본도 모집단을 정확하게 대표할 수 없다는 것을 의미한다. 모집단과 표본 사이의 차이가 유의미할 만큼 크고, 첫 번째 표본과 동일한 방식으로 추출된 다른 샘플들에서도 이 차이가 계속될 것으로 예상될 때 표본편향이 발생했다고 볼 수 있다.

Note 자기선택 표본편향

옐프와 같은 소셜 미디어 사이트에서 보는 레스토랑, 호텔, 카페 등에 대한 리뷰는 제출하는 사람들이 무작위로 선택되지 않았기 때문에 편향되기 쉽다. 오히려 작성자 스스로 리뷰 작성에 대한 주도권을 쥐고 있다. 이것은 자기 선택 편향으로 연결된다. 리뷰를 남기고자 하는 사람들은 시설에 대한 안 좋은 경험이 있거나, 해당 시설과 관련이 있거나, 리뷰를 남기지 않는 보통 사람들과는 뭔가 다른 유형의 사람일 가능성이 높다.

이와 같은 자기 선택 표본은 상황을 정확히 파악하기 위한 지표로는 사용하기 어렵지만, 어떤 시설을 비슷한 시설과 단순 비교할 때는 오히려 더 신뢰할 만 하다. 비슷한 자기 선택 편향이 각각의 경우에 똑같이 적용될 수 있기 때문이다.

2.1.1 편향

통계적 편향은 측정 과정 혹은 표본추출 과정에서 발생하는 계통적인 오차를 의미한다.

랜덤표본추출로 인한 오류와 편향에 따른 오류는 신중하게 구분해서 봐야 한다. 목표물에 총을 쏘는 과정을 생각해보자. 매번 목표물의 한가운데를 정확히 맞힌다는 것은 불가능하다.

한가운데 정확히 한 발 맞추기도 힘들것이다. 편향되지 않은 프로세스에도 오차가 있긴 하지만, 그것은 랜덤하며 어느 쪽으로 강하게 치우치는 경향이 없다(그림 2-2). 반면 (그림 2-3)에서 표시된 결과는 편향을 보여준다. x방향과 y방향 모두에서 랜덤한 오차가 있고 평향도 있다, 탄착점이 오른쪽 위 사분면에 떨어지는 경향을 볼 수 있다.

2.1.2 랜덤 선택

리터러리 다이제스트가 루즈벨트가 아닌 랜던일 거라고 예측하도록 유도한 표본편향 문제를 피하기 위해 조지 갤럽은 미국 유권자를 대표하는 포본을 얻기 위한 좀 더 과학적으로 조사자를 선정하는 방법을 채택했다. 대표성을 담보하기 위한 여러 방법이 있지만, 결국 핵심은 **랜덤표본추출**이다.

랜덤표본추출이 언제나 쉬운 일은 아니다. 접근 가능한 모집단의 적절한 정의가 매우 중요하다. 고객의 대표 프로필을 만들 목적으로 파일럿 고객 설문 조사를 준비한다고 하자. 설문 조사는 대표성을 필요로 하지만 동시에 그만큼 많은 노동력을 필요로 한다.

먼저 고객이 누구인지 정의해야 한다. 구매 금액이 0보다 큰 모든 고객의 명단을 작성할 수 있다.

이때 모든 과거 고객을 포함할 것인가?

제품을 환불한 고객도 포함할 것인가?

내부의 테스트 구매자는?

사업자는?

대금 청구 대행사와 고객을 모두 포함할 것인가?

다음으로 표본추출 절차를 정해야 한다. '무작위로 100명의 고객을 선택'하는 방법이 있겠다. 유동적인 상황에서 표본추출을 해야 할 경우(예를 들어 실시간 거래 고객이나 웹 방문자), 시기가 중요할 수 있다. (예를 들어 평일 오전 10시의 웹 방문자와 주말 오후 10시의 웹 방문자가 다를 수 있다.)

층화표본추출에서는, 모집단을 여러 층으로 나누고 각 **층**에서 무작위로 샘플을 추출한다. 정치 설만 단체에서 백인, 흑인, 라틴계 유권자들의 투표 성향을 조사한다고 가정하자. 모집단에서 취한 단순랜덤표본에서는 흑인과 라틴 인구가 지역에 따라 너무 적게 나올 수 있다. 이런 경우 해당 층에 높은 가중치를 주는 표본추출을 통해, 계층마다 동일한 표본크기를 얻을 수 있다.

2.1.3 크기와 품질 : 크기는 언제 중요해질까?

빅데이터 시대라고 해도 의외로 데이터 개수가 적을수록 더 유리한 경우가 있다. 랜덤표본추출에

시간과 노력을 기울일수록 편향이 줄 뿐만 아니라 데이터 탐색의 데이터 품질에 더 집중할 수 있다. 예를 들어 결측값이나 특잇값으로부터 유용한 정보를 얻을 수 있다. 몇 백만개 데이터 중에서 결측치를 추적하거나 특잇값을 평가하는 것은 어려울 수 있지만, 수천개의 샘플에서는 가능할 수 있다. 데이터가 너무 많을 경우, 데이터를 일일이 손으로 검사하고 조사하기는 매우 어렵다.

그렇다면 언제 정말 대량의 데이터가 필요할까?

빅데이터가 가치 있을 것이다라는 것의 일반적인 예상은 데이터가 크고 동시에 희박할 때 이다. 구글에서 입력받은 검색 쿼리를 처리한다고 가정해보자. 행렬을 만들어 열은 용어를, 행은 개별 검색 쿼리들을 의미하고 쿼리에 해당 용어가 포함되는지 여부에 따라 원소의 값이 0 또는 1이 된다고 하자.

목표는 주어진 쿼리에 대해 가장 잘 예측된 검색 대상을 결정하는 것이다.

영어 단어는 150,000개가 넘으며 구글은 연간 1조 이상의 검색어를 처리한다. 따라서 대부분 원소가 0인 거대한 행렬이 된다.

이는 방대한 양의 데이터가 누적될 때만 대부분의 쿼리에 대해 효과적인 검색 결과를 반환할 수 있는, 진정한 의미의 빅데이터 문제이다. 더 많은 데이터가 축적될수록 결과가 더 좋을 수 밖에 없다. 인기 있는 검색어의 경우, 전혀 문제가 되지 않는다. 특정 시간에 인기가 급상승하는 소수의 주제는 효과적인 데이터를 아주 신속하게 찾을 수 있다. 현대 검색 기술의 진정한 가치는 백만 번에 한 번 정도 발생하는 검색 쿼리까지도 포함하여 다양한 검색 쿼리에 대해 상세하고 유용한 결과를 얻을 수 있다는 것에 있다.

실제 **연관된** 레코드(정확한 검색 쿼리나 아주 비슷한 것이 들어 있는 레코드, 클릭한 사용자의 정보를 포함한다)의 수는 수천 개 정도만 되도 효과적일 수 있다는 것을 기억하자.

그러나 결국 이러한 연관된 레코드를 얻기 위해서는 수조의 데이터 포인트가 필요하다. 그리고 이 경우, 물론 랜덤표본 추출은 도움이 되지 않는다.

2.1,4 표본평균과 모평균

기호 $\bar{}$은 모집단의 표본평균을 나타내는데 사용되는 반면, μ 는 모집단의 평균을 나타내는 데 사용된다. 이 둘을 왜 따로 구분할까?

표본에 대한 정보는 관찰을 통해 얻어지고, 모집단에 대한 정보는 주로 작은 표본들로부터 추론한다. 통계학자들은 이렇게 다른 기호로 이 두 가지를 구분하는 것을 선호한다.

주요 개념

- ***빅데이터 시대에도 랜덤표본추출은 데이터 과학자들의 화살통에 남은 중요한 화살이다.***
- ***편향은 측정이나 관측에 계통적 오차가 있어 전체 모집단을 제대로 대표하지 못할 경우 발생한다.***
- ***데이터 품질이 데이터 양보다 중요할 때가 자주 있다. 랜덤표본추출은 편향을 줄이고, 나중에 다시 하려면 훨씬 비싼 값을 치를 수도 있는 품질 향상을 용이하게 한다.***

2.2 선택편향

선택 편향은 데이터를 의식적이든 무의식적이든 선택적으로 고르는 관행을 의미한다. 결국 오해의 소지가 있거나 단편적인 결론을 얻게 된다.

용어정리

- ***편향: 계통적 오차***
- ***데이터 스누핑: 뭔가 흥미로운 것을 찾아 광범위하게 데이터를 살피는 것***
- ***방대한 검색 효과: 중복 데이터 모델링이나 너무 많은 예측변수를 고려하는 모델링에서 비롯되는 편향 혹은 비재현성***

어떤 가설을 세우고 그것을 시험하기 위해 잘 설계된 실험을 수행한다면, 그 결과에 대해 강하게 확신할 수 있다. 하지만 이런 경우는 참 드물다. 보통은 가지고 있는 데이터를 먼저 확인한 후, 그 안에서 패턴을 찾고자 한다. 하지만 이것이 참된 패턴인지 아니면 그냥 **데이터 스누핑**을 통해 나온 결과인지 알 수 없다. 다시 말해, 뭔가 흥미로운 것이 나올 때 까지 데이터를 너무 샅샅이 뒤진 결과는 아닐까? "데이터를 충분히 오래 고문하다 보면 언젠가 뭐든 털어놓을 것이다." 라는 식의 통계학자들끼리 농담처럼 하는 얘기가 있다.

실험을 통해 가설을 테스트해서 확인한 현상과 사용 가능한 데이터를 통해 발견한 현상의 차이를, 다음 사고실험을 통해 알아보자.

빅데이터를 반복적으로 조사하는 것이 데이터 과학의 중요한 가치 명제이기에, 선택 편향에 대해 조심할 필요가 있다. 데이터 과학자들이 특별히 걱정하는 선택 편향의 한 형태는 존 엘더가 **방대한 검색 효과**라고 부르는 것이다. 큰 데이터 집합을 가지고 반복적으로 다른 모델을 만들고 다른 질문을 하다보면, 언젠가 흥미로운 것을 발견하기 마련이다. 그 결과는 정말로 의미 있는 것인가? 아니면 우연히 얻은 예외 경우인가?

성능을 검증하기 위해 하나이상의 홀드 아웃 세트를 이용하면 이를 방지할 수 있다. 또한 엘더는 데이터 마이닝 모델에서 제시하는 예측들을 검증하기 위해, **목푯값 섞기** 라는 것을 추천했다.

방대한 검색 효과 외에도, 통계에서 일반적으로 나타나는 선택 편향으로는 비랜덤표본추출, 데이터 체리 피킹, 특정한 통계적 효과를 강조하는 시간 구간 선택, '흥미로운' 결과가 나올 때 실험을 중단하는 것 등이 여기에 포함된다.

2.2.1 평균으로의 회귀

평균으로의 회귀란 주어진 어떤 변수를 연속적으로 측정했을 때 나타나는 현상이다. 예외적인 경우가 관찰되면 그 다음에는 중간 정도의 경우가 관찰되는 경향이 있다. 따라서 예외 경우를 너무 특별히 생각하고 의미를 부여하는 것은 선택 편향으로 이어질 수 있다.

스포츠를 좋아하는 사람이라면 '신인상 수상자의 2년 차 슬럼프'라는 얘기를 한번 쯤 들어봤을 것이다. 왜 그럴까? 거의 모든 스포츠에는 적어도 공이나 퍽으로 경기를 치르는 스포츠에는 매우 중요한 두 요소가 있다.

- 실력
- 행운

평균에 대한 회귀는 일정의 선택 편향으로 인해 나타나는 결과이다. 성적으로 신인을 뽑을 때, 진짜 실력도 있지만 운도 동시에 따랐을 것이다. 다음 시즌에는 실력이 그대로 유지되지만, 대부분의 경우 운은 그렇지 않다. 따라서 성적은 나빠질 것이다. 이 현상은 1886년 프랜시스 골턴이 처음 밝혔는데, 유전적 경향성과 관련하여 기술했다. 예를 들어 키가 엄청나게 큰 남성의 자식들도 아버지처럼 키가 큰 것은 아니었다.

CAUTION 여기서 회귀는 '돌아간다'는 의미로서 통계적 모델링 방법의 하나인 선형회귀와는 구별되어야 한다. 선형회귀는 예측변수와 결과변수 사이의 선형적 관ㄱ를 추정하는 방법이다.

주요개념

- **가설을 구체적으로 명시하고 랜덤표본추출을 원칙에 따라 데이터를 수집하면 평향을 피할 수 있다.**
- **다른 모든 형태의 데이터 분석은 데이터 수집/분석 프로세스에서 생기는 편향의 위험성을 늘 갖고 있다. (데이터 마이닝에서의 모델 반복 실행, 연구 시 데이터 스누핑, 흥미로운 사건의 사후 선택 등)**

2.3 통계학에서의 표본분포

통계의 **표본분포**라는 용어는 하나의 동일한 모집단에서 얻은 여러 샘플에 대한 표본통계량의 분포를 나타낸다. 고전 통계의 대부분은 (작은)표본을 가지고 (매우 큰)모집단을 추론하는 것과 관련있다.

용어 정리

- **표본통계량 : 더 큰 모집단에서 추출된 표본 데이터들로부터 얻은 측정 지표**
- **데이터 분포 : 어떤 데이터 집합에서의 각 개별 값의 도수분포**
- **표본분포 : 여러 표본들 혹은 재표본들로부터 얻은 표본통계량의 도수분포**
- **중심극한정리 : 표본크기가 커질수록 표본분포가 정규분포를 따르는 경향**
- **표본오차 : 여러 표본들로부터 얻은 표본통계량의 변량**

  (개별 데이터 값들의 변량을 뜻하는 표준편차와 혼동하지 말것!!!!!!!!!!)

일반적으로 우리는 (**표본통계량**으로) 어떤 것을 측정하거나 (통계 또는 머신러닝 모델로) 뭔가를 모델링하기 위해 표본을 뽑는다. 우리는 표본을 통해 추정이나 모델을 하기 때문에 오차가 있을 수 있다. 우리가 다른 표본을 뽑았다면 결과가 다를 수 있다. 따라서 그것이 얼마나 달라질지에 관심이 있다. 주요관심사는 **표본의 변동성**이다. 우리가 많은 양의 데이터를 가지고 있다면 추가로 표본을 얻어서 통계의 분포를 직접 관찰할 수 있다. 보통은 이미 최대한 많은 데이터를 사용하여 추가로 표본을 얻어서 통계의 분포를 직접 관찰할 수 있다. 보통은 이미 최대한 많은 데이터를 사용하여 추정치 또는 모델을 계산했을 것이므로, 모집단에서 추가 표본을 얻는 옵션은 쉽게 이용할 수 없을 것이다.

CAUTION 흔히 데이터 분포라고 알려진 개별 데이터 포인트와 분포와 표본분포라고 알려진 표본통계량의 분포를 구별하는 것이 중요하다.

평균과 같은 표본통계량의 분포는 데이터 자체의 분포보다 규칙적이고 종 모양일 가능성이 높다. 통계의 기반이 되는 표본이 클수록, 그럴 가능성이 높은 것이 사실이다. 또한 표본이 클수록 표본통계량의 분포가 좁아진다.

렌딩클럽에 대출을 신청한 사람들을 평가하기 위해 연간 소득 정보를 사용하는 예를 통해 이를 설명할 수 있다. 이 데이터에서 단순히 1,000개 값으로 이뤄진 표본, 5개 값의 평균 1,000개로 이뤄진 표본 , 20개 값의 평균 1,000개로 이뤄진 표본, 이렇게 3개의 표본을 뽑는다고 생각해보자. 각 샘플의 히스토그램을 살펴보자.

개별 데이터 값의 히스토그램은 예상대로 넓게 분산되어 있고 한쪽으로 기울어져 있다. 5와 20의 평균에 대한 히스토그램은 갈수록 좁고 벨 모양이 된다. 다음은 시각화 패키지 ggplot2를 사용하여 이러한 히스토그램을 생성하는 R코드 이다.

```
x <- seq(from=-3, to=3, length=300)
gauss <- dnorm(x)
par(mar=c(3, 3, 0, 0)+.1)
plot(x, gauss, type="l", col='blue', xlab='', ylab='', axes=FALSE)
polygon(x, gauss, col='blue')
```

```
#단순랜덤표본을 하나 취한다.
samp_data <- data.frame(
                        income=sample(loans_income, 1000),
                        type='data_dist')
```

```
# take a sample of means of 5 values
samp_mean_05 <- data.frame(
                  income = tapply(sample(loans_income, 1000*5),
                  rep(1:1000, rep(5, 1000)), FUN=mean),
                  type = 'mean_of_5')
# take a sample of means of 20 values
samp_mean_20 <- data.frame(
  income = tapply(sample(loans_income, 1000*20),
                  rep(1:1000, rep(20, 1000)), FUN=mean),
  type = 'mean_of_20')
# bind the data.frames and convert type to a factor
income <- rbind(samp_data, samp_mean_05, samp_mean_20)
income$type = factor(income$type,
                     levels=c('data_dist', 'mean_of_5', 'mean_of_20'),
                     labels=c('Data', 'Mean of 5', 'Mean of 20'))
# plot the histograms
ggplot(income, aes(x=income)) +
  geom_histogram(bins=40) +
  facet_grid(type ~ .)
```

보충학습!

R에서 그룹별로 합계 또는 평균을 구하고 싶을 때는 tapply함수, aggregate함수, subset함수 등을 이용한다.

tapply : 주어진 함수를 그룹별로 각 자료 값에 적용하는 함수

Usage

tapply(X, INDEX, FUN=NULL, ..., default = NA, simplify = T)

X : 평균을 구할 변수

INDEX : 그룹 변수

FUN = 평균을 구할 함수

ex) A,B 그룹에 나이가 15~35세인 사람들이 5명씩 있을 때 그룹별 평균 연령을 구해보자.

먼저 아래와 가이 예제 자료를 생성한다.

rep 함수를 이용해 A와 B를 5개씩 반복 입력하고 set.seed와 sample 함수를 이용해 15~35세 사이의 10개 나이를 추출한다.

```
group<-c(rep("A",5),rep("B",5))
set.seed(1); age<-c(sample(15:35),10)
mydat<-data.frame(group, age)
head(mydat)
tapply(mydat$age, mydat$group, mean)
```

2.3.1 중심극한정리

이러한 현상을 **중심극한정리**라고 한다. 모집단이 정규분포가 아니더라도, 표본크기가 충분하고 데이터가 정규성을 크게 이탈하지 않는 경우, 여러 표본에서 추출한 평균은 종 모양의 정규곡선을 따른다. 중심극한정리 덕분에, 추론을 위한 표본분포에, 즉 신뢰구간이나 가설검정을 계산하는 데에 t분포 같은 정규근사 공식을 사용할 수 있다.

중심극한정리는 전통적인 통계 교과서에서 절반 정도를 할애할 정도로 중요하게 다뤄지는 가설검정과 신뢰구간에 대한 밑바탕이 되기에 모든 참고서에서 중요하게 다뤄진다.

데이터 과학자들도 이 정도로 중요하다는 사실은 알고 있어야 한다. 하지만 형식적인 가설검정이나 신뢰구간이 데이터 과학에서는 이 정도로 중요하지 않다. 대부분의 경우 부트스트랩을 사용할 수 있기에, 데이터 과학의 관점에서는 중심극한정리가 그렇게 중요하지는 않다.

2.3.2 표준오차

표준오차는 통계에 대한 표본분포의 변동성을 한마디로 말해주는 단일 측정 지표이다. 표준오차는 표본값들의 표준편차 s와 표본크기 n을 기반으로 한 통계량을 이용하여 추정할 수 있다.

$$\text{표준오차} = SE = \frac{s}{\sqrt{n}}$$

표본크기가 커지면 표준오차가 줄어든다. 표준오차와 표본크기 사이의 관계를 때로는 **n제곱근의 법칙**이라고 한다. 즉 표준오차를 2배로 줄이려면 표본크기를 4배 증가시켜야 한다.

표준오차 공식의 유효성은 중심극한정리를 통해 증명된다. 하지만 실제 표준오차를 이해하기 위해 중심극한정리에 너무 의존할 필요는 없다. 표준오차를 측정할 때 고려할 사항은 다음과 같다.

1. 모집단에서 완전히 새로운 샘플들을 많이 수집한다.
2. 각각의 새 샘플에 대해 통계량을 계산한다.
3. 2단계에서 얻은 통계량의 표준편차를 계산한다. 이것을 표준오차의 추정치로 사용한다.

실질적으로 표준오차를 추정하기 위해 새 샘플을 수집하는 접근 방식은 일반적으로 불가능하다(통계적으로 낭비가 심하다). 다행히 새로운 샘플을 뽑을 필요가 없다는 사실이 밝혀졌다. 대신 **부트스트랩** 재표본을 사용할 수 있다. 현대 통계에서 부트스트랩은 표준오차를 추정하는 표준 방법이 되었다. 사실상 모든 통계에 사용할 수 있으며 중심극한정리 또는 기타 분포 가정에 의존하지 않는다.

CAUTION 표준편차와 표준오차!!!!!!

개별 데이터 포인트의 변동성을 측정하는 표준편차, 그리고 포본 측정 지표의 변동성을 측정하는 표준오차 둘을 혼동하지 말자

주요개념

- **표본통계량의 도수분포는 그 해당 지표가 표본마다 다르게 나타날 수 있음을 보여준다.**
- **부트스트랩 방식 혹은 중심극한정리에 의존하는 공식을 통해 표본분포를 추정할 수 있다.**
- **표준오차는 표본통계량의 변동성을 요약하는 주요 지표이다.**

2.4 부트스트랩

통계량이나 모델 파라미터(모수)의 표본분포를 추정하는 쉽고 효과적인 방법은, 현재 있는 표본에서 추가적으로 표본을 복원추출하고 각 표본에 대한 통계량과 모델을 다시 계산하는 것이다. 이러한 절차를 **부트스트랩**이라 하며, 데이터나 표본통계량이 정규분포를 따라야 한다는 가정은 꼭 필요하지 않다.

용어정리

부트스트랩 표본 : 관측 데이터 집합으로부터 얻은 복원추출 표본

재표집(재표본추출, 리샘플링) : 관측 데이터로부터 반복해서 표본추출하는 과정, 부트스트랩과 순열(셔플링) 과정을 포함한다.

개념적으로, 부트스트랩은 원래 표본을 수천, 수백만 번 복제하는 것이라고 생각할 수 있다. 그리고 이를 통해 원래 표본으로부터 얻어지는 모든 정보를 포함하는 가상 모집단을 얻게 된다.

그런 다음 이 가상 모집단으로부터 표본분포를 추정할 목적으로 표본을 수집할 수 있다.

그렇다고 표본을 실제로 엄청난 횟수로 반복 복제한다는 것은 아니다. 대신 각각의 표본을 뽑은 후 각 관측치를 다시 원래 자리에 돌려놓는다. 즉, **복원추출**한다. 이런 식의 효과적인 방법으로, 뽑을 때마다 각 원소가 뽑힐 확률은 그대로 유지하면서 무한한 크기의 모집단을 만들어낼 수 있다. 크기 n의 샘플 평균을 구하는 부트스트랩 재표본추출 알고리즘은 다음과 같다.

1. 샘플 값을 하나 뽑아서 기록하고 제자리에 놓는다.
2. n번 반복한다.
3. 재표본추출된 값의 평균을 기록한다.
4. 1~3단계를 R번 반복한다.
5. R개의 결과를 사용하여
   a. 그것들의 표준편차(표본평균의 표준오차)를 계산한다.
   b. 히스토그램 또는 상자그림을 그린다.
   c. 신뢰구간을 찾는다.

R(부트스트랩 반복 횟수)은 임의로 설정한다. 반복 횟수가 많을수록 표준오차나 신뢰구간에 대한 추정이 더 정확해진다. 이런 절차를 통해 표본통계량 혹은 추정한 모델 파라미터의 부트스트랩 집합을 얻게 되고, 결과적으로 이 집합이 얼마나 변하는지를 조사할 수 있다.

R패키지 **boot**는 이런 여러 단계를 하나의 함수로 제공한다. 예를 들어 다음은 앞에서 사용한 대출받은 사람들의 소득 데이터에 부트스트랩을 적용하는 코드다.

```
stat_fun <- function(x, idx) median(x[idx])
boot_obj <- boot(loans_income, R = 1000, statistic=stat_fun)
```

stat_fun 함수는 인덱스 idx로 지정된 표본의 중앙값을 계산한다.

```
Bootstrap Statistics :
    original  bias    std. error
t1*    62000 -85.987    235.8748
```

중간값의 원래 추정치는 62,000달러이다. 부트스트랩 분포는 추정치에서 약 -86달러만큼의 **편향**이 있고 235달러의 표준오차가 있는 것으로 나타난다.

부트스트랩은 다변량 데이터에도 적용될 수 있다. 여기서 각 행은 여러 변수들의 값을 포함하는 하나의 샘플을 의미한다. 모델 파라미터의 안정성(변동성)을 추정하거나 예측력을 높이기 위해, 부트스트랩 데이터를 가지고 모델을 돌려볼 수 있다. 분류 및 회구 트리(**의사결정트리**라고도 함)를 사용할 때, 여러 부트스트랩 샘플을 가지고 트리를 여러 개 만든 다음 각 트리에서 나온 예측값을 평균 내는 것이 (분류 문제에서는 과반수 투표를 한다) 일반적으로 단일 트리를 사용하는 것보다 효과적이다. 이 프로세스를 배깅이라고 부른다.

CAUTION 부트스트랩은 표본크기가 작은 것을 보완하기 위한 것이 아니다. 새 데이터를 만드는 것도 아니며 기존 데이터 집합의 빈 곳을 채우는 것도 아니다. 모집단에서 추가적으로 표본을 뽑는다고 할 때, 그 표본이 얼마나 원래 표본과 비슷할지를 알려 줄 뿐이다.

보충학습!!!!!

R에서 부트스트랩을 시행하는 코드를 직접 짜볼 수도 있겠지만, 기본적으로 제공되는 함수를 이용할 수도 있다.

그 과정은 아래와 같이 단순하지만 다른 함수들과 사용법에 다른 점이 많아 처음에는 많이 낯설 것이다.

그럼 rnorm() 함수로 정규분포이 100개 난수를 생성한 후, 이를 이용하여 부트스트랩 샘플의 중앙값을 계산해 히스토그램을 그려보려 한다.

```
#표본 만들기
set.seed(20)
x<-rnorm(100)
#부트스트랩 샘플 평균 구하기 및 히스토그램 그리기
bootstrap_median <- boot(x,statistic  =stat_fun, R=1000)

Bootstrap Statistics :
        original       bias
t1* 2.220446e-16 -0.004765886
     std. error
t1*   0.1720967
```

2.4.1 재표본추출 대 부트스트래핑

종종 **재표본추출**이라는 용어는 앞서 소개한 **부트스트랩**이라는 것과 비슷한 의미로 사용된다. 보통 재표본추출이라는 용어는 여러 포본이 결합되어 비복원추출을 수행할 수 있는 순열 과정을 포함한다. 부트스트랩이라는 용어는 항상 관측된 데이터로부터 복원추출한다는 것을 의미한다.

주요 개념

- **부트스트랩(데이터로부터 복원추출)은 표본통계량의 변동성을 평가하는 강력한 도구이다.**
- **부트스트랩은 표본분포의 수학적 근사치에 대한 엄청난 연구 없이도 다양한 환경에서 유사한 방식으로 적용될 수 있다.**
- **또한 수학적 근사가 어려운 통계량에 대해서도 샘플링 분포를 추정할 수 있다.**
- **예측 모델을 적용할 때, 여러 부트스트랩 표본들로부터 얻은 예측값을 모아서 결론을 만드는 것(배깅)이 단일 모델을 사용하는 것보다 좋다.**

2.5 신뢰구간

도수분표포, 히스토그램, 상자그림, 표준오차는 모두 표본추정에서 잠재적인 오차를 이해하는 방법들이다. 신뢰구간은 이들과 다른 방법이다.

용어정리

신뢰수준: 같은 모집단으로부터 같은 방식으로 얻은, 관심 통계량을 포함할 것으로 예상되는 신뢰구간 백분율

구간끝점: 신뢰구간의 최상위, 최하위 끝점

사람들은 보통 불확실성에 대해 자연스런 반감이 있다. 사람들(특히 전문가)은 '잘 모른다'는 식으로 말하는 것을 꺼린다. 분석가들이나 관리자들은 불확실성을 인정하면서도, 그것이 어떤 단일 수치(**점추정**)로 제시할 때, 추정치에 과도한 믿음을 둔다. 단일 수치가 아닌 어떤 범위로 추정치를 제시하는 것이 이러한 경향을 막는 방법이다. 신뢰구간은 통계적 샘플링 원칙에 근거한다.

신뢰구간은 항상 90% 또는 95%와 같이 (높은) 백분률도 표현되는 포함수준과 함께 나온다. 90% 신뢰구간이란, 표본통계량의 부트스트랩 표본분포의 90%를 포함하는 구간을 말한다. 더 일반적으로, 표본추정치 주위의 x% 신뢰구간이란, 평균적으로 유사한 표본추정치 x% 정도(비슷한 샘플링 절차를 따랐을 때)가 포함되어야 한다.

표본크기 n과 관심 있는 표본통계량이 주어졌을 때, 부트스트랩 신뢰구간을 구하는 법은 다음과 같다.

1. 데이터에서 복원추출 방식으로 크기n인 표본을 뽑는다. (재표본추출)
2. 재표본추출한 표본에 대해 원하는 통계량을 기록한다.
3. 1~2단계를 R번 반복한다.
4. x% 신뢰구간을 구하기 위해, R개의 재표본 결과로부터 분포의 양쪽 끝에서 [(100-x)/2]% 만큼 잘라낸다.
5. 절단한 점들은 x% 부트스트랩 신뢰구간의 양 끝점이다.

다음 그림은 대출 신청자의 평균 연간 소득에 대한 90% 신뢰구간을 보여준다. 평균이 57.573달러인 20개 표본에서 얻은 결과이다.

부트스트랩은 대부분의 통계량 혹은 모델 파라미터에 대한 신뢰구간을 생성하는데 사용할 수 있는 일반적인 기법이다. 반세기 넘도록 컴퓨터가 없던 시절 통계 교과서 및 SW에서는 수식, 특히 t-분포로 구한 신뢰구간을 사용했다.

NOTE 물론 표본 결과를 얻었을 때 정말로 궁금한 것은 '참값이 일정 구간 안에 있을 확률은 얼마인가?' 이다. 신뢰구간이 이 질문에 대한 답을 주는 것은 아니지만, 결국 대부분의 사람이 이 질문에 대한 대답을 설명하는 근거로 신뢰구간을 사용한다. 신뢰구간과 관련된 확률 문제는 '표본추출 절차와 모집단이 주어지만 ~ 할 확률은 얼마인가?' 라는 문구로 시작된다. 반대로 '표본 결과가 주어졌을 때, (모집단에 대해 어떤 것이 참일) 확률은 무엇인가?' 라는 질문은 더 복잡한 계산과 불확실한 요소를 필요로 한다.

신뢰구간과 관련된 백분율을 **신뢰수준**이라고 부른다. 신뢰수준이 높을수록 구간이 더 넓어진다. 표본이 작을수록 구간이 넓어진다. (즉, 불확실성이 더 커진다) 두 가지 모두 말이 된다. 더 확실할수록, 데이터가 적을수록, 확실히 참값을 얻기에 충분한 신뢰구간을 확보해야 한다.

NOTE 데이터 과학자는 신뢰구간을 통해 표본결과가 얼마나 달라질 수 있는지 알 수 있다. 데이터 고학자들이 학술 논문을 발표하거나 규제 기관에 결과를 보고하는 데 이런 정보들을 사용하지는 않을 것이다. 대신 추정에 대한 잠재적인 오류를 알려주거나, 더 큰 표본이 필요한지 여부를 파악하는 용도로 이것들을 사용할 것이다.

주요개념

- 신뢰구간은 구간 범위로 추정값을 표시하는 일반적인 방법이다.
- 더 많은 데이터를 보유할수록 표본추정치의 변위가 줄어든다.
- 허용할 수 있는 신뢰수준이 낮을수록 신뢰구간은 좁아진다.
- 부트스트랩은 신뢰구간을 구성하는 효과적인 방법이다.

2.6 정규분포

종 모양의 **정규분포**는 전통적인 통계의 상징이다. 표본통계량 분포가 보통 어떤 일정한 모양이 있다는 사실은 이 분포를 근사화하는 수학공식을 개발하는 데 강력한 도구가 되었다.

용어정리

- 오차 : 데이터 포인트와 예측값 혹인 평균 사이의 차이
- 표준화(정규화 하다) : 평균을 빼고 표준편차로 나눈다.
- z 점수 : 개별 데이터 포인트를 정규화한 결과
- 표준정규분포 : 평균 = 0 , 표준편차 = 1인 정규분포
- QQ그림 : 표본분포가 정규분포에 얼마나 가까운지를 보여주는 그림

정규분포에서 데이터의 68%는 평균의 표준편차 내에 속하며 95%는 표준편차 두 배수 내에 있다.

CAUTION 대부분의 데이터가 정규분포를 따르기 때문에, 즉 이게 정상이기 때문에 정상적인 분포라고 부르는 것은 아니다. 실제로 전형적인 데이터 과학 프로젝트에서 사용되는 대부분의 변수들, 실제 대부분의 원시 데이터는 전체적으로 정규분포를 따르지 않는다. 표본분포에서 대부분의 통계량이 정규분포를 따른다는 점에서 정규분포의 유용함이 드러날 뿐이다. 설령 그렇다 해도 일반적으로 정규분포 가정은 경험적 확률분포나 부트스트랩 분포를 구할 수 없는 경우 사용되는 최후의 수단이다.

NOTE _ 정규분포는 18세기 후반과 19세기 초반의 수학자 프리드리히 가우스의 이름을 따, 가우스 분포라고도 불린다. 그 전에는 '오차'분포라는 이름으로도 불린 적이 있다. 통계적 관점에서 오차는 참값과 표본평균과 같은 통계적 추정치의 차이를 말한다. 예를 들어 표준편차는 데이터 평균과의 오차에서 비롯된다. 가우스는 천문학적 측정 오차가 정규분포를 따른다는 연구를 통해 정규분포의 발전에 크게 이바지 했다.

보충학습 (R 정규분포 : rnorm, dnorm, pnorm, dnorm)

rnorm	**난수 (generates random deviates)**	**rnorm(n,mean=0,sd=1)**
dnorm	확률밀도(gives the density)	dnorm(x,mean=0,sd=1,log=False)
pnorm	누적확률밀도(gives the distribution function)	pnorm(q,mean=0,sd=1,lower.tail=T, log.p=F)
qnorm	분위점(gives the quantile function)	qnorm(p,mean=0,sd=1,lower.tail=T,log.p=F)

dnorm : mean=0, sd=1인 정규분포상에서 quantile 값이 1일때의 density 값을 구해보자.

pnorm : 임의의 quantile 이하의 면적을 구한다.

qnorm : pnorm()의 역함수라고 생각하면 된다. 역의 면적을 구한다.

2.6.1 표준정규분포와 QQ그림

표준정규분포는x축의 단위가 평균의 표준편차로 표현되는 정규분포를 말한다. 데이터를 표준정규분포와 비교하려면 데이터에서 평균을 뺀 다음 표준편차로 나누면 된다. 이를 **정규화** 또는 **표준화**라고 한다. 여기서의 '표준화'는 DB레코드의 표준화와 관련이 없다는 점에 유의하자. 이렇게 변환한 값을 **z 점수**라고 하며, 정규분포를 **z 분포**라고도 한다.

QQ 그림은 표본이 정규분포에 얼마나 가까운지를 시각적으로 판별하는 데 사용된다. QQ그림은 Z 점수를 오름차순으로 정렬하고 각 값의 z점수를 y축에 표시한다. x축은 정규분포에서의 해당 분위수를 나타낸다. 데이터가 표준화되었기 때문에, 단위는 평균으로부터 떨어진 데이터의 표준편차 수에 해당한다. 점들이 대략 대각선 위에 놓이면 표본분포가 정규분포에 가까운 것으로 간주할 수 있다. 정규분포에서 임의로 생성한 100개의 값에 대한 QQ그림을 보여준다. 예상대로 점들이 대각선에 가깝게 따라가는 것을 볼 수 있다. 이 그림은 R의 qqnorm함수를 사용해서 만들 수 있다.

```
norm_samp <- rnorm(100)
par(mar=c(3, 3, 0, 0)+.1)
qqnorm(norm_samp, main='', xlab='', ylab='')
abline(a=0, b=1, col='grey')
```

CAUTION 데이터를 z점수로 변환(즉, 데이터를 표준화 또는 정규화)한다고 해서, 데이터가 정규분포가 되는 것은 아니다. 단지 비교를 목적으로 데이터를 표준정규분포와 가은 척도로 만드는 것 뿐이다.

주요개념

- 정규분포는 불확실성과 변동성에 관한 수학적 근사가 가능하도록 했다. 이는 통계의 역사적 발전에 필수적 이었다.
- 원시 데이터 자체는 대개 정규분포가 아니지만, 표본들의 평균과 합계, 그리고 오차는 많은 경우 정규분포를 따른다.
- 데이터를 z점수로 변환하려면 데이터의 값에서 평균을 빼고 표준편차로 나눈다. 그러면 데이터를 정규분포와 비교할 수 있다.

2.7 긴 꼬리 분포

역사적으로 통계에서 정규분포에 대한 중요성에도 불구하고, 또한 그 이름에 담긴 '정상적'이라는 의미와는 달리, 데이터는 일반적으로 정규분포를 따르지 않는다.

용어정리

꼬리 : 적은 수의 극단값이 주로 존재하는, 도수분포의 길고 좁은 부분

왜도 : 분포의 한쪽 꼬리가 반ㄷ쪽 다른 꼬리보다 긴 정도

오차나 표본통계량의 분포를 볼 때 정규분포는 적절하고 유용하지만, 정규분포가 일반적으로 원시 데이터 분포의 특징을 나타내지는 않는다. 때로는 분포가 소득 데이터와 같이 비스듬(비대칭) 하게 기울어져 있거나 이항 데이터 같이 이산적일 수 있다. 대칭 및 비대칭 분포 모두 **긴 꼬리**를 가질 수있다. 분포의 꼬리는 양 극한값에 해당한다. 실무에서는 긴 꼬리를 잘 보는 것이 중요하다. 주식시장의 붕괴와 같은 이례적인 사건이, 정규분포로 예측되는 것보다 훨씬 더 자주 일어날 수 있다고 예측하는 **블랙스완 이론**이 발표된 바 있다.

주가 수익률은 데이터의 긴 꼬리 특성을 설명하기 위한 좋은 예가 된다. 넷플릭스의 일일 주식 수익률에 대한 QQ그림을 보여준다. 이 그림은 다음 R코드로 생성한다.

```
nflx <- sp500_px[,'NFLX']
nflx <- diff(log(nflx[nflx>0]))
#양수 nflx값에 대한 로그 처리 후, 미분
qqnorm(nflx, main='', xlab='', ylab='')
abline(a=0, b=1, col='grey')
```


낮은 값의 점들은 대각선보다 훨씬 낮고, 높은 값은 선보다 훨씬 위에 위치한다. 이것은 데이터가 정규분포를 따른다고 할 때 예상되는 것보다 훨씬 더 많은 극단값을 관찰할 가능성이 있음을 의미한다.

또 한가지 다른 일반적인 현상을 보여준다. 평균에서 표준편차 이내에 있는 데이터의 점들은 선에 가까이 있다. 투기는 데이터가 '중간에서는 정상'이면서 더 긴 꼬리를 갖는 현상에 대해 언급한 바 있다.

NOTE_ 관측된 자료에 적합한 통계 분포를 찾는 작업에 관한 통계학 문헌이 많이 있다. 이 일을 할때 너무 데이터만을 보고 판단하지 말라. 이는 과학 못지않은 일종의 기술이기도 하다. 데이터는 가변적이며, 보기에는 하나가 아닌 여러 유형의 분포와 일치할 수 있다. 일반적으로 주어진 상황을 묘사하기에 적합한 분포 유형을 결정하기 위해, 분야 지식과 통계 지식을 모두 활용해야 한다. 예를 들어 5초 마다 연속적으로 서버의 인터넷 트래픽 수준에 대한 데이터를 얻는다고 하자. **푸아송 분포가 '시간 주기별 이벤트'를 모델링하는 데 가장 적합한 분포라는 사전지식이 있다면 큰 도움이 될 것이다.**

주요개념

- **대부분의 데이턴느 정규분포를 따르지 않는다.**

- **정규분포를 따를 것이라는 가정은 자주 일어나지 않는 예외에 관한 과소평가를 가져올 수 있다.**

2.8 스튜던트의 t분포

t분포는 정규분포와 생김새가 비슷하지만, 꼬리 부분이 약간 더 두껍고 길다. 이것은 표본통계량의 분포를 설명하는 데 광범위하게 사용된다. 표본평균의 분포는 일반적으로 t분포와 같은 모양이며, 표본 크기에 따라 다른 계열의 t분포가 있다. 표본이 클수록 더 정규분포를 닮은 t분포가 형성된다.

용어정리

n : 표본크기

자유도 : 다른 표본크기, 통계량, 그룹의 수에 따라 t분포를 조절하는 변수

t분포를 처음 주창한 윌리엄 고셋은 '더 큰 모집단에서 추출한 표본평균의 표본분포는 무엇인가?'라는 질문에 대한 답을 찾고 싶어했다. 그는 재표본추출 실험을 시작했다. 3,000명의 범죄자들에 대한 신장과 왼손 가운뎃 손가락 길이 데이터에서 무작위로 4개의 표본을 추출한다.(우생학 시대가 되면서, 범죄자들에 대한 데이터와 범죄 경향과 신체적 또는 심리적 특징들 사이의 상관관계를 밝히는 데 관심이 많아졌다). 그는 x축에 표준화된 결과(z 점수)를, y빈도를 나타내는 도표를 만들었다. 별도로

그는 오늘날 스튜던트의 t로 알려진 함수를 유도해냈고, 표본 결과에 가장 적합한 함수를 구했으며,

으로 둘을 비교해 봤다.

표준화된 여러 통계자료를 t분포와 비교하여 신뢰구간을 추정할 수 있다. 표본평균이 x인, 크기 n의 표본이 있다고 가정하자. s가 표본 표준편차라면, ㄹ표본평균 주위의 90% 신뢰구간은 다음과 같이 주어진다.

$\overline{x} \pm t_{n-1}(.05) \times \frac{s}{\sqrt{n}}$

여기서 $t_{n-1}(.05)$은 (n-1) 자유도를 갖는 t분포의 양쪽 끝에서 5%를 '잘라내는' t 통계량을 의미한다. 표본평균, 두 표본평균 간의 차이, 회귀 파라미터, 그 외 다른 통계량들의 분포를 구할 때 t분포를 사용한다.

 지금과 같은 성능의 컴퓨터가 1908년 부터 널리 보급됐다면, 처음부터 통계학은 계산을 많이 필요로 하는 재표본추출 방법을 훨씬 더 많이 이용했을 것이다. 컴퓨터가 없던 시절, 통계학자들은 표본분포를 근사화하기 위한 수학적 기법과 t분포와 같은 함수로 눈을 돌렸다. 1980년대 들어오면서 컴퓨터로 실용적인 재표집 실험이 가능해졌지만, 그땐 이미 교과서와 SW에 t 분포나 그와 유사한 다른 분포함수들을 사용하는 것이 깊이 자리 잡혀 있었다.

표본통계량의 상태를 묘사할 때 t분포의 정확도는 표본에 대한 통계량의 분포가 정규분포를 따른 다는 조건을 필요로 한다. 원래 모집단이 정규분포를 따르지 않을 때 조차도, 표본통계량은 보통 정규분포를 따르는 것으로 나타났다. 이는 t분포가 널리 적용되는 이유이기도 하다. 이런 현상을 **중심극한 정리**라고 부른다.

NOTE 데이터 과학자가 t분포와 중심극한정리에 대해 알아야 할 것은 무엇일까? 실은 별로 없다. 이러한 분포는 고전적인 통계적 추론에 사용되기는 하지만, 데이터 과학이 주로 추구하는 목적과는 조금 거리가 있다. 데이터 과학자에게는 불확실성과 변동성을 이해하고 정량화하는 것이 중요하다. 이러한 목적을 위해서라면, 경험적 부트스트랩 표본추출을 통해서도 표본 오차에 대한 대부분의 질문에 답을 얻을 수 있다. 하지만 데이터 과학자들은 R과 같은 통계 SW 혹은 A-B 테스트 나 회귀분석 같은 통계 절차를 통해 나온 t통계량을 매일 만나게 될 테니 알아둬야 한다.

주요개념

- **t분포는 사실 정규분포와 비슷한데 꼬리만 조금 더 두꺼운 형태이다.**
- **이것은 표본평균, 두 표본평균 사이의 차이, 회귀 파라미터 등의 분포를 위한 기준으로 널리 사용된다.**

2.9 이항분포

용어정리

- 시행: 독립된 결과를 가져오는 하나의 사건 (예 : 동전던지기)
- 성공 : 시행에 대한 관심의 결과 (유의어 : 1, 즉 0에 대한 반대)
- 이항식 : 두 가지 결과를 갖는다. (유의어 : 예/아니오, 0/1, 이진)
- 이항시행 : 두 가지 결과를 가져오는 시행 (유의어: 베르누이 시행)
- 이항분포 : x번 시행에서 성공한 횟수에 대한 분포 (유의어 : 베르누이 분포)

예/아니오 (이항의) 식의 결론은 구매/구매하지 않음, 클릭/클릭하지 않음, 생존/사망 등과 같은 의사결정 과정에서 아주 중요하기 때문에, 분석에서 핵심이라고 할 수 있다. 이항분포를 이해할 때 핵심은 일련의 **시행**들이라는 아이디어인데, 각 시행은 정해진 확률로 두 가지 결과를 갖는다.

이항분포란, 각 시행마다 그 성공확률(p)이 정해져 있을 때, 주어진 시행 횟수(n) 중에서 성공한 횟수(n)의 도수분포를 의미한다. x,n,p값에 따라 다양한 이항분포들이 있다. 이항분포로 답하고자 하는 것은 다음과 같은 질문이다.

한 번의 클릭이 판매로 이어질 확률이 0.02일 때, 200회 클릭으로 0회 매출을 관찰할 확률은 얼마인가?

R함수 **dbinom**은 이항 확률을 계산할 때 사용한다.

```
> dbinom(2,5,0.1)
[1] 0.0729
```

이는 n=5인 시행에서 각 시행의 성공 확률이 p=0.1일 때 정확히 x = 2인 성공이 나올 확률을 의미한다. 보통은 n번의 시도에서 x번 또는 그 이하로 성공할 확률이 얼마일지 알아보는 데 관심이 있다.

```
> pbinom(2,5,0.1)
[1] 0.99144
```

이는 성공 확률이 0.1인 시행을 다섯 번 했을 때, 두 번 이하의 성공을 관찰할 확률을 의미한다.

이항분포의 평균은 n x p 이다. 성공 확률이 p 일 경우, n번의 시행에서 예상되는 성공 횟수로 생각할 수도 있다. 분산은 n x p(1-p)이다. 시행 횟수가 충분할 경우(특히 p가 0.50에 가까울 때) 이항분포는 사실상 정규분포와 구별이 어렵다. 실제로 표본크기가 커질수록 이항 확률을 구하기 위해선 많은 계산이 필요하다 보니, 대부분의 통계 절차에서는 평균과 분산으로 근사화한 정규분포를 사용한다.

주요개념

- **이항 결과는 무엇보다도 중요한 결정 사항들(구매 또는 구매하지 않거나 클릭하거나 클릭하지 않거나 등등)을 나타내기에, 모델을 만드는 데 매우 중요하다.**
- **이항 시행은 두 가지 결과, 즉 하나는 확률 p, 다른 하나는 확률 (1-p)인 실험을 말한다.**
- **n이 크고 p가 0또는 1에 너무 가깝지 않은 경우, 이항분포는 정규분포로 근사할 수 있다.**

2.10 푸아송 분포와 그 외 관련 분포들

많은 작업이 주어진 어떤 비율에 따라 임의로 사건들을 발생시킨다. 웹사이트에 방문한 방문객 수, 톨게이트에 들어오는 자동차(시간에 따른 사건), 혹은 1제곱미터당 건물의 결함, 코드 100줄당 오타(공간에 대한 사건) 같은 예를 들 수 있다.

용어정리

- **람다 : 단위 시간이나 단위 면적당 사건이 발생하는 비율**

- **푸아송 분포 : 표집된 단위 시간 혹은 단위 공간에서 발생한 사건의 도수분포**
- **지수분포 : 한 사건에서 그다음 사건까지의 시간이나 거리에 대한 도수분포**
- **베이불 분포 : 사건 발생률이 시간에 따라 변화하는 지수분포의 일반화된 버전**

2.10.1 푸아송 분포

이전에 발생한 데이터를 통해, 시간 단위 혹은 공간 단위에서의 평균적인 사건의 수를 추정할 수 있다. 하지만 시간별 혹은 공간별로 사건 발생이 얼마나 다른지 알고 싶을 때가 있다. 푸아송 분포는 시간 단위 또는 공간 단위로 표본들을 수집할 때, 그 사건들의 분포를 알려준다. '5초 동안 서버에 도착한 인터넷 트래픽을 95%의 확률로 완벽하게 처리하는 데 필요한 용량은 얼마일까?' 같은 대기행렬 관련 질문을 처리할 때 유용하다.

푸아송 분포의 핵심 파라미터는 λ(람다) 이다. 이는 어떤 일정 시간/공간 구간 안에서 발생한 평균 사건 수를 의미한다. 푸아송 분포의 분산 역시 λ 이다.

잘 알려진 기술은 대기행렬 시뮬레이션에서 푸아송 분포를 따르는 난수를 생성하는 것이다. R의 **rpois 함수**는 발생시킬 난수의 개수와 람다, 이 두 인수를 입력받는다.

```
> rpois(100,2)
  [1] 8 2 2 0 2 1 2 1 6 3 2 1 2 1 2 4
 [17] 2 4 2 2 2 1 3 1 0 0 3 1 4 0 3 2
 [33] 2 2 3 4 4 3 0 4 4 2 5 1 3 0 2 1
 [49] 7 1 5 0 4 2 2 5 5 2 1 2 1 2 3 3
 [65] 3 0 3 1 1 3 4 2 1 2 1 1 1 0 2 5
 [81] 4 3 4 2 1 3 2 1 1 4 1 1 1 1 3 3
 [97] 0 1 3 4
>
```

이 코드는 $\lambda = 2$ 인 푸아송 분포에서 100개의 난수를 생성한다. 예를 들어 고객 서비스 센터에 접수되는 문의 전화가 분당 평균 2회이면, 이 코드는 100분을 시뮬레이션 하여 100분 당 문의전화 횟수를 알려준다.

2.10.2 지수분포

푸아송 분포에 사용된 것과 동일한 변수 λ를 사용하여 사건과 사건 간의 시간 분포를 모델링 할 수 있다. 웹사이트 방문이 일어나는 시간 사이 또는 톨게이트에 자동차가 도착하는 시간 사이를 예를 들 수 있다. 또한 공학 분야에서는 고장이 발생하는 시간을 모델링한다거나 프로세스 관리에서는 개별 고객 상담에 소요되는 시간을 모델링하는 데 사용한다. 지수분포에서 난수를 생성하기 위한 R코드에는 두 개의 인수 ,n(난수 발생 개수)과 **비율**(시간 주기당 사건 수)을 사용한다.

```
> rexp(100,.2)
  [1]  0.3970743  2.7434158 11.7478696  2.3260710  2.8362567  5.2224648
  [7]  0.5896602  2.5358210  1.8875535 11.5218486  0.5537176  0.3522091
 [13]  3.4533959  2.6012404  0.6423687  6.2115951  1.2578362  1.6959717
 [19] 10.5841685  0.4080910  1.8548950  0.3066226 17.6621244  0.6030490
 [25]  6.1016766  5.1141988  1.5971308  4.0210735  0.8534677  6.9456744
 [31]  3.2309492 14.8777604  1.1530001 15.3774535 10.5907641  8.6971796
 [37]  1.1935036  6.6009253  4.0283819  8.5876166  7.0104114  4.3908955
 [43] 24.9209225  7.1300282  7.2953379  7.2098671  3.3989707  5.3446159
 [49]  6.3478813 16.2268334  4.8402119  0.4567260  8.1177269  1.9425552
 [55]  5.2452872  4.2811460  1.0373652  1.0226975  1.9481185  3.3016827
 [61]  6.2025476 23.9648103 12.8797549 17.9745989  0.2807424  5.8122680
 [67]  4.5799960  0.9550523 10.6498233 11.5636143  1.8041855  7.9303090
 [73]  7.1347099  0.1337033  1.5561814  1.7245872 14.7884891  1.2249053
 [79]  0.6747239  8.7352290  1.6512882  3.3404950  3.9173969  2.4485859
 [85] 14.5924842  1.5809468  2.9628690  3.2334129 11.1849641  5.7349282
```

```
 [91]  3.6133551  5.2549621  6.6956560  3.2601898 14.1027791  8.0808233
 [97]  0.2834473  9.9970684  0.2379500  0.8257706
```

이 코드는 주기별 평균 사건 수가 0.2인 지수분포에서 100개의 난수를 생성한다. 따라서 분당 평균적으로 0.2회 서비스 문의 전화가 걸려오는 경우, 100분 동안의 서비스 센터 문의 전화를 시뮬레이션 할 수 있다.

푸아송이나 지수분포에 대한 시뮬레이션 연구에서 핵심은 λ가 해당 기간 동안 일정하게 유지된다는 가정이다. 전반적으로 이는 거의 적절하지 않다. 예를 들어 도로의 교통 상황이나 데이터 망의 트래픽은 시간대와 요일에 따라 같을 수 없다. 그러나 시간 주기 또는 공간을, 일정 기간 충반히 동일하도록 영역을 잘 나눈다면, 해당 기간 내의 분석 및 시뮬레이션이 가능하다.

2.10.3 고장률 추정

많은 응용 분야에서 사건 발생 비율 λ는 이미 알려져 있거나, 이전 데이터를 통해 추정할 수 있다. 하지만 드물게 발생하는 사건의 경우 반드시 그렇지는 않다. 예를 들어 항공기 엔진 고장은 감사하게도 정말 드물게 일어나는 사건이다. 주어진 엔진 유형에 대해, 고장이 발생하는 사건 사이의 시간을 예측하기 위한 데이터가 거의 없다. 데이터가 전혀 없다면 사건 발생률을 추정할 아무런 근거도 없다.

그러나 몇 가지 추측을 할 수는 있다. 20시간 후에도 아무런 일도 일어나지 않았다면, 시간당 발생률이 1이 아니라는 것은 분명히 알 수 있다. 이렇게 시뮬레이션 또는 확률의 직접 계산을 통해 다른 가상 사건 발생률을 평가하고, 그 이하로 떨어지지 않을 임계값을 추산할 수 있다. 데이터가 있긴 하지만 정확하고 신뢰할 만한 발생률을 추정하기에 충분하지 않은 경우, 적합도검정을 통해 적용한 여러 발생률 중 어떤 것이 관찰된 데이터에 가장 적합한지를 알 수 있다.

2.10.4 베이불 분포

많은 경우, 사건 발생률은 시간에 따라 일정하지 않다. 변화 주기가 일반적인 사건 발생 구간 보다 훨씬 길다면 문제가 안된다. 앞서 언급했듯이 비율이 상대적으로 일정한 구간으로 분석을 세분화하면 되기 때문이다. 그러나 사건 발생률이 시간에 따라 지속적으로 변한다면 지수 (또는 푸아송) 분포는 더 이상 유용하지 않다. 기계 고장이 대표적인 예이다. 시간이 지날수록 고장 위험은 증가한다. **베이블 분포**는 지수분포를 확장한 것으로, 형상 파라미터 β(베타)로 지정된 대로 발생률이 달라질 수 있다. β>1 일 경우, 발생률은 시간이 지남에 따라 증가하며, β < 1이면 감소한다. 베이불 분포는 사건 발생률 대신 고장 시간 분석에 사용되기에 두 번째 인수는 구간당 사건 발생률보다는 특성 수명으로 표현된다. 기호로 그리스 문자 η (에타, eta) 를 사용한다. **척도**변수 라고도 한다.

베이불을 사용할 때는 두 변수 β, η의 추정이 포함된다. 가장 적합한 베이불 분포를 추정하고 데이터를 모델링하는 데에는 SW를 사용한다.

베이블 분포에서 난수를 생성하는 R코드는 n(발생 개수), shape, scale 세 가지 인수를 사용한다. 예를 들어 다음 코드는 1.5의 형상 파라미터와 5,000의 특성 수명을 갖는 베이불 분포에서 100개의 난수(수명)을 생성한다.

```
> rweibull(100,1.5,5000)
  [1]  1997.2183  2209.5406  9309.6019  9612.1220  5007.8764   803.3844
  [7]  1631.2535  5708.3693  5086.0295  6958.6581 15065.6922  3333.5976
 [13]  9460.5955  1694.2787  6922.8671 11864.3462  9252.4260  2841.2133
 [19]  6660.1460  3492.3189  5460.2208 12455.0691  3359.3771  8936.8646
 [25]   622.8848  1470.5530  5403.9980  5448.3568  5430.7505  6471.4521
 [31]  6754.7353  7864.6278  5001.9052  7366.0651  5937.5501  3374.2524
 [37]  6768.2659  3285.2722  4071.7505  3003.2587   222.9509  7238.6873
 [43]  1414.7661  4250.1625  2173.6494  1314.2445  6178.3603  3838.4223
 [49]  5151.1005  2719.4471 13122.9951 13927.0179  4214.6389 14847.8429
 [55]  2726.6176  1608.5025  3565.1836  5019.0228  4970.0029  2226.2344
 [61]  2730.1259  2321.7246   509.2774  4759.4015  1577.9675 10580.4878
 [67]  7987.4562  1594.6161  1878.4547  9510.5843  4065.6947  8616.3447
```

```
 [73]  4326.0470  4022.2734 18911.4381  8890.9855  5512.1978  6085.2867
 [79]  7351.0059 10627.9577  9382.9017  2941.3522  8908.1852  6291.3307
 [85] 12485.3912  7039.3477  4140.0718 10645.8452  1715.4752  3119.3486
 [91]  2122.8440   110.3867  2540.9343  5319.5993  1923.9482  2721.9676
 [97]  7226.4303  4544.0350  1390.4270  4986.0297
```

주요개념

- *일정 비율로 발생하는 사건의 경우, 시간 단위또는 공간 단위 당 발생하는 사건의 수를 푸아송 분포로 모델링 할 수 있다.*
- *이 시나리오에서 한 사건과 다음 사건 간의 시간/거리를 지수분포로 모델링할 수도 있다.*
- *시간에 따라 변화하는 사건 발생률(예를 들어 증가하는 고장률)은 베이불 분포로 모델링할 수 있다.*

2.11 마치며

빅데이터 시대에 정확한 추정이 요구되는 경우, 랜덤표본추출의 원칙을 지키는 것이 매우 중요하다. 데이터의 무작위 선택을 통해 주어진 데이털르 그냥 사용하는 것보다 편향을 줄이고 질적으로 더 좋은 데이터로 만들어서 사용해야 한다. 다양한 표본추출 및 데이터 생성 분포에 대한 지식을 바탕으로, 랜덤 변이로 인한 추정치의 잠재정 오차를 정량화 할 수 있다. 동시에 부트스트랩(관찰된 데이터로부터 복원추출하는 방법)은 표본추정에서 잠재적 오차를 판별할 때, 유용한 '모든 문제에 적용 가능한' 방법이다.