랜덤대치 기반 프라이버시 보호 기법의 정확성 개선 방법[1)]

이창우, 강주성[2)]
국민대학교 수학과
e-mail : jskang@kookmin.ac.kr

A method for improving the accuracy of privacy-preserving technique based on random substitutions

Chang Woo Lee, Ju-Sung Kang
Dept. of Mathematics, Kookmin University

요 약

랜덤대치 기법은 프라이버시 손상 관점에서 높은 프라이버시를 보존하면서 원본 데이터의 분포를 재구축하여 데이터 유용성을 확보한다. 데이터 유용성을 위한 랜덤대치 기법의 정확성을 높이는 문제는 그동안 면밀히 연구되지 않았다. 본 논문에서는 랜덤대치 기법이 대부분의 데이터에 대해서 상대적으로 낮은 정확성을 보임을 실험을 통해 밝히고, 이론적인 분석과 실험을 바탕으로 정확성을 높일 수 있는 실용적인 알고리즘 개선 방법을 제안한다.

1. 서론

실용적인 프라이버시 보호 기술의 대표적인 응용 분야인 프라이버시 보존형 데이터 마이닝에서는 정보제공자의 비밀 데이터를 보호하기 위해서 변형된 데이터를 마이너에게 제공한다. 데이터의 변형은 프라이버시 관련 정보를 노출시키지 않기 위함이며, 데이터 변형의 가장 실용적인 방법이 랜덤화 기법이다. 최근에 발표된 랜덤대치(random substitutions)[1]는 랜덤화 기법 중의 하나로 안전성과 효율성이 높고, 다양한 분야에 응용 가능한 방법으로 알려져 있다. 하지만 데이터 유용성을 위한 랜덤대치 기법의 정확성을 높이는 문제는 지금까지 면밀히 연구되지 않았다. 본 논문에서는 랜덤대치 기법의 정확성에 대하여 심도 있는 분석을 실시한다. 이를 통하여 랜덤성에 의존하는 랜덤대치 기법의 특성상 특정 데이터 집합에 대한 재구축 과정에서의 취약점을 밝히고, 이를 개선하여 보다 정확성을 높일 수 있는 개선된 랜덤대치 방법을 제시한다.

2. 랜덤대치 기법

먼저 Agrawal-Haritsa[1]과 Dowd-Xu-Zhang[2]에 의해서 제안된 랜덤대치 기법에 대하여 간략히 소개한다.

2.1 원본 데이터의 변형 과정

랜덤대치 기법의 기본적인 아이디어는 각 데이터 레코드의 속성 값을 어떤 확률 모델에 따라 속성의 정의역으로부터 랜덤하게 선택된 다른 값으로 바꾸는 것이다. 이 확률 모델은 각 속성 값이 바뀔 확률을 나타내는 전환행렬(transition matrix)을 생성하여 정의할 수 있다. 속성의 정의역을 $U=\{u_1,\cdots,u_N\}$라 가정하고 한 데이터의 속성 값 u_k가 u_h로 바뀔 확률을 다음과 같이 정의한다.

$$\Pr[u_k \to u_h] = m_{h,k} \ .$$

이렇게 정의된 확률 값 $m_{h,k}$를 성분으로 하는 $N\times N$ 크기의 행렬을 M이라 놓는다.

랜덤대치 기법에서 데이터를 변형하는 방법을 알고리즘으로 표현하면 다음과 같다.

알고리즘 1. 랜덤대치 기법의 데이터 변형 방법

입력 : n개의 레코드로 이루어진 원본 데이터 집합 O,
속성 A에 대한 정의역 $U=\{u_1,\cdots,u_N\}$,
U에 대한 전환행렬 $M_{N\times N}$.

결과 : 변환된 데이터 집합 P

수행 과정 :

모든 레코드 $o \in O$에 대해 다음을 실행한다.

1. o가 가지는 속성 값의 인덱스 값 k를 구한다.
   즉, o가 가지는 속성 값은 u_k이다.
2. (0, 1] 상의 균등분포로부터 랜덤수 r을 선택한다.
3. 다음을 만족하는 정수 $1 \leqq h \leqq N$ 를 찾는다
$$\sum_{i=1}^{h-1} m_{ik} < r \leqq \sum_{i=1}^{h} m_{ik}$$
4. o에 대응되는 변환된 레코드의 속성 값을u_h로 결정한다.

1) 본 연구는 지식경제부 및 정보통신연구진흥원의 IT신성장동력 핵심기술개발사업의 일환으로 수행하였음. [2005-Y001-04, 차세대 시큐리티 기술 개발]

2) 교신 저자

$X=(X_1,\cdots,X_N)^T$와 $Y=(Y_1,\cdots,Y_N)^T$를 각각 원본 그리고 변형된 데이터 집합에서 각 레코드들이 갖는 속성의 개수에 대한 열벡터라고 하자. X를 추정하기 위해서 변형 데이터 집합의 u_i의 개수에 대한 벡터 Y의 관측값 $\boldsymbol{y}=(y_1,\cdots,y_N)$을 이용하면 X에 대한 추정량(estimator) $\hat{X}$를 얻을 수 있다.

$$\hat{X}=(\hat{X_1},\cdots,\hat{X_N})^T=M^{-1}\boldsymbol{y}$$

2.2 효율적인 분포 재구축 방법

랜덤대치 기법의 효율적인 구현을 위하여 데이터 재구축 과정에서 필요로 하는 역행렬을 구하는 공식은 다음과 같이 간단히 얻을 수 있음이 알려져 있다[4]. $\gamma>1$, $N>1$인 γ-대각 행렬 M에 대해서, M의 역행렬 $M^{-1}=\left(m_{ij}^{-1}\right)_{N\times N}$은 다음과 같다.

$$m_{ij}^{-1}=\begin{cases}\dfrac{\gamma+N-2}{\gamma-1} & ,\ i=j\\[2ex] \dfrac{1}{1-\gamma} & ,\ i\neq j\end{cases}.$$

따라서 다음과 같이 분포를 재구축할 수 있다. 즉, 전체 $n=\sum_{i=1}^{N}Y_i$개의 레코드에 대한, $\hat{X_i}$는 다음과 같다.

$$\hat{X_i}=\frac{\gamma+N-1}{\gamma-1}\,Y_i-\frac{n}{\gamma-1}\quad .$$

3. 랜덤대치 기법의 취약성 분석

재구축된 데이터의 각 속성 값의 개수 $\hat{X_i}$는 변형된 레코드의 각 속성 값의 개수 Y_i에 의존하므로, 랜덤대치의 정확성을 측정하기 위해 주어진 $n=\sum_{i=1}^{N}X_i=\sum_{i=1}^{N}Y_i$개의 레코드에 대한 $Y=\left(Y_1,\cdots,Y_N\right)^T$와 $\hat{X}=\left(\hat{X}_1,\cdots,\hat{X}_N\right)^T$의 분산 및 $\|X\|$을 다음과 같이 계산할 수 있다.

$$Var(Y)=E[\|Y-E[Y]\|^2]=\frac{(N-1)(N+2\gamma-2)n}{(\gamma+N-1)^2},$$

$$Var(\hat{X})=\frac{(N-1)(N+2\gamma-2)n}{(\gamma-1)^2},$$

$$\|X\|=\left(\sum_{j=1}^{N}X_j^2\right)^{1/2}\geq\frac{n}{\sqrt{N}}\ .$$

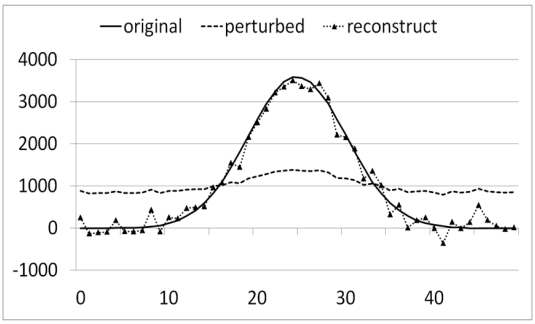
따라서 $\hat{X}$의 정확도는 다음과 같이 계산할 수 있다.

$$\frac{\sigma_{\hat{X}}}{\|X\|}\leq\frac{\sqrt{(N-1)(N+2\gamma-2)n}}{\gamma-1}\Big/\frac{n}{\sqrt{N}}$$

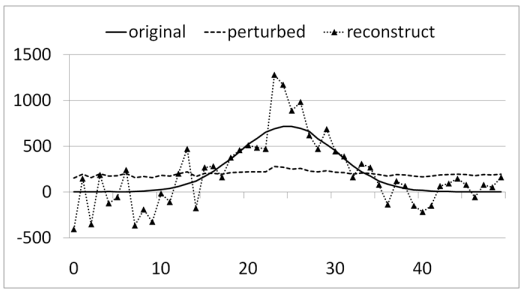
$$=\frac{\sqrt{N(N-1)(N+2\gamma-2)}}{(\gamma-1)n^{1/2}}\ .$$

이론적으로, $\hat{X}$의 정확도는 N이 커질수록 낮아지고, n과 γ가 커질수록 높아질 것이라고 예상할 수 있다. 이를 실험을 통해 확인해 보았다.

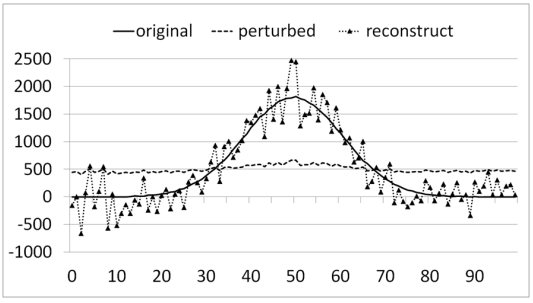
실험에서 데이터 레코드는 정규분포 또는 균등분포를 따르는 5,000과 50,000개의 레코드로 N은 25, 100으로 변화시키면서 실험하였다. 그림 1, 2, 3은 원본(original) 데이터의 속성 값이 정규분포를 따르는 경우에 변형된(perturbed) 데이터의 속성 값 분포와 재구축(reconstruct) 분포를 나타낸 것이며, 그림 4는 균등분포에 대한 각 데이터 속성값 분포를 표현한 것이다.



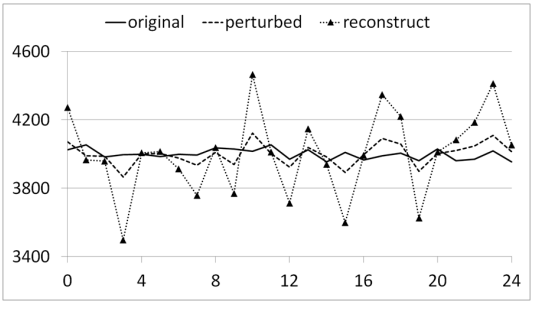
(그림 1) N=50, n=50,000일 때, 속성값 분포(정규)



(그림 2) N=50, n=10,000일 때, 속성값 분포(정규)



(그림 3) N=100, n=50,000일 때, 속성값 분포(정규)



(그림 4) N=25, n=50000일 때, 속성값 분포 (균등)

우리는 실험 결과로부터 전체 원본 데이터의 수(n)가 작을 때, 전체 데이터에 대한 속성의 도메인(N)이 클 때 랜덤대치 기법의 정확성이 현저히 떨어짐을 확인할 수 있었다. 이는 이론적인 분석을 통해 예상했던 결과로서 전체적으로 균등하게 랜덤화가 일어날 것이라는 기대와 달리 실제 랜덤화에 있어서는 약간의 차이가 발생하게 된다. 하지만 이것이 재구축 과정에서 정확도를 현저히 떨어뜨리는 원인이 되는 것이다.

4. 정확성 개선을 위한 l-확장 랜덤대치 기법

랜덤대치의 안전성은 γ-증폭과 관련한 (ρ_1, ρ_2) 프라이버시 보증에 의존한다. 그런데 l-확장 랜덤대치에서는 하나의 데이터에 대하여 l개의 랜덤화된 데이터를 생성하고자 하므로 랜덤대치와는 달리 l개의 랜덤화된 데이터에 같은 데이터가 중복되어 생성되는 경우가 발생할 수 있다. 이러한 상황을 고려하여 l-확장 랜덤대치의 γ-증폭을 다시 계산하여야 한다. l-확장 랜덤대치에서 임의의 한 원소를 l번 랜덤화하여 생성된 데이터 l개로 구성된 다중집합(multi-set)을 y^*라 하자. 다중집합 y^* 내의 l개 원소 중에서 임의의 속성 u_i가 중복되어 나타난 회수를 $s_i (0 \le s_i \le l)$라 할 때, 속성 u_i가 집합 y^*로 랜덤화 될 전이확률(transition probability)이

$$p[u_i \to y^*] = \frac{\gamma^{s_i}}{(\gamma+N-1)^l}$$

과 같이 계산된다. $\gamma \ge 1$ 이므로, 모든 u_i에 대하여 랜덤화될 확률이 가장 큰 y^*는 $s_i = l$ 일 때이고, 그 확률이 가장 작은 때는 $s_i = 1$인 y^*이다. 따라서 l-확장 랜덤화의 랜덤화된 집합 y^*에 대한 γ-증폭은 다음과 같다.

$$\forall u_1, u_2 \in U,\ \frac{p[u_1 \to y^*]}{p[u_2 \to y^*]} \le \frac{\gamma^l/(\gamma+N-1)^l}{1/(\gamma+N-1)^l} = \gamma^l.$$

따라서 정리 2.1에 의해 l-확장 랜덤대치는 다음을 만족하는 ρ_1^*, ρ_2^*에 의해 (ρ_1^*, ρ_2^*) 프라이버시 보증을 만족한다.

$$\frac{\rho_2^*}{\rho_1^*} \cdot \frac{1-\rho_1^*}{1-\rho_2^*} > \gamma^l.$$

이는 사전확률이 ρ_1^*이하인 어떤 성질도 l-확장 랜덤대치를 통해 사후확률이 ρ_2^*를 넘을 수 없다는 것을 의미한다. 그런데 $\gamma^l \ge \gamma$ 이므로, 동일한 사전 확률에 대해서 l-확장 랜덤대치로 인한 사후확률의 상계가 더 크거나 같다. 즉, 하나의 원본 데이터에 대한 l개의 랜덤화된 데이터에 대해 중복이 적을수록 기존의 랜덤대치와 비슷한 안전성을 보이고, 중복이 없다면 동일한 안전성을 보이게 된다. C를 최대 중복수라고 하면, 중복이 발생할 확률은 다음과 같다.

$$P[C \ge 2] = 1 - \frac{N-l+l\gamma}{(\gamma+N-1)^l} \prod_{i=1}^{l-1} (N-i).$$

이는 N 에 반비례하고, γ 과 l에 비례하여 증가한다.

l-확장 기법은 N의 값이 비교적 클 경우에 사용이 되므로 중복은 적을 것이다. 하지만 l의 값에 비례하여 어느 정도 증가하여 발생할 것이므로, 발생한 중복 속성 값에 대해서는 재랜덤화를 하여 중복이 일어나지 않도록 한다. 알고리즘은 다음과 같다.

알고리즘 2. l-확장 랜덤대치 기법

입력 : n개의 레코드로 이루어진 원본 데이터 집합 O,
속성 A에 대한 정의역 $U = \{u_1, \cdots, u_N\}$,
U에 대한 전환행렬 $M_{N\times N}$, $l \ge 1$.

결과 : 변환된 데이터 집합 P

수행 과정 :

모든 $o \in O$에 대해 다음 과정을 실행한다.

1. o가 가지는 속성 값의 인덱스 값 k를 구한다.
   즉, o가 가지는 속성 값은 u_k이다.
2. $1 \le j \le l$ 에 대하여 다음을 실행한다.($U_0 = \varnothing$)
   2.1. (0, 1] 상의 균등분포로부터 랜덤수 r을 선택한다.
   2.2. 다음을 만족하는 정수 $1 \leqq h \leqq N$ 를 찾는다.
   $$\sum_{i=1}^{h-1} m_{ik} < r \leqq \sum_{i=1}^{h} m_{ik}\ .$$
   2.3. if $u_h \not\in U_{j-1}$
   then o의 j번째 랜덤화 속성값 = u_h ,
   $U_j = U_{j-1} \cup \{u_h\}$.
   else go to 2.1.
3. o에 대응되는 변환된 레코드의 속성값의 집합을 U_l로 결정한다.

원본데이터 집합을 l배하여 랜덤화를 수행하였으므로, 최종적으로 재구축된 분포는 $\widehat{X^*}$를 l로 나누어 계산한다. 다음 소절에서 우리는 개선된 정확성에 대하여 논한다.

4.1 랜덤대치 기법과의 정확도 비교분석

l-확장 랜덤대치는 기존 n개의 레코드를 l배 확장 하는 것이므로, 다음과 같은 D^*에서 랜덤대치를 수행한다.

$$D^* = [d_1^*, d_2^*, \ldots, d_{ln}^*]$$
$$= [d_1, \ldots, d_1, d_2, \ldots, d_2, \ldots, d_n, \ldots, d_n]$$

또한, D^*의 속성별 빈도수를 X^*, 변형된 데이터의 속성별 빈도수를 Y^*라 했을 때, 기존의 랜덤대치와 비교하여 다음과 같은 성질을 만족한다.

$$X^* = (lX_1, \ldots, lX_N) = l\,X\ ,$$
$$E[Y^*] = X^*P = l\,XP = l\,E[Y],$$
$$Var(Y^*) = l\ Var(Y),$$

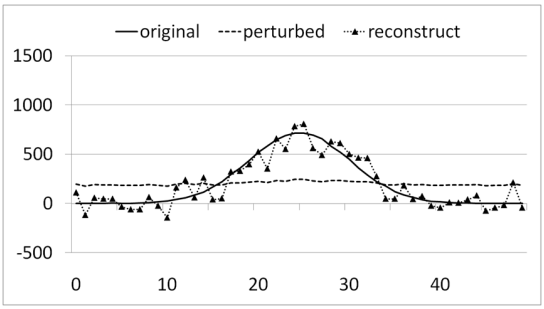
$$Var(\widehat{X^*}/l) = \frac{1}{l} Var(\hat{X}).$$

따라서 $\widehat{X^*}/l$의 X에 대한 정확도는 기존의 랜덤대치와 비교하여 다음과 같이 향상된다.

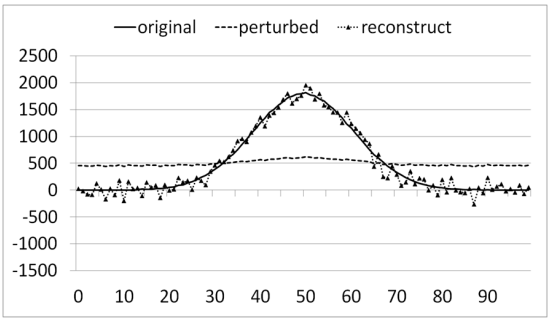
$$\frac{\sigma_{\widehat{X^*}/l}}{\|X\|} = \frac{1}{\sqrt{l}} \frac{\sigma_{\hat{X}}}{\|X\|}.$$

4.3. 실험 결과

실험은 랜덤대치에서와 동일한 환경에서 l값의 변화를 주어 관찰하였고, 다음 세 개의 그림은 l=4일 때, 각 속성값의 변화된 분포를 나타낸다.

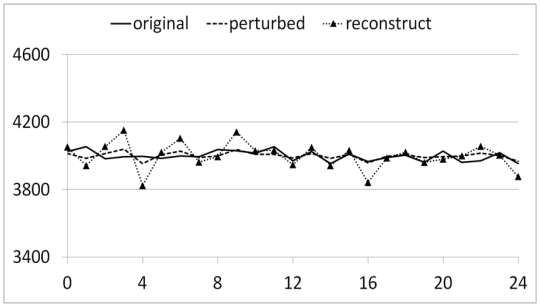


(그림 5) N=50, n=50000일 때, 속성값 분포(정규)



(그림 6) N=100, n=50000일 때, 속성값 분포(정규)

그림 5와 그림 6은 동일한 N, n값에 대하여 기존의 랜덤대치방식의 결과인 그림 2와 그림 3에 비해 정확성이 현저히 향상되었음을 보여준다.



(그림 7) N=25, n=50000일 때, 속성값 분포(균등)

그림 7에서도 균등분포상의 고정된 γ, N에 대하여 그림 3과 비교하여 정확성이 향상되었음을 확인할 수 있다. 다음 두 표는 실험 결과를 이론값과 비교하여 l값의 변화에 따른 평균오차를 나타낸 표이다.

<표 1> n=50000 일 때, 평균오차 비교

	l = 1	l = 4	l = 16
평균오차	0.341073	0.169863	0.085865
이론값	0.337025	0.168513	0.084256

<표 2> n=100000 일 때, 평균오차 비교

	l = 1	l = 4	l = 16
평균오차	0.240577	0.122234	0.061435
이론값	0.2383	0.11915	0.059575

두 결과에서 모두 $\sqrt{l}$만큼 정확성이 향상되었음을 확인할 수 있다.

5. 결론

정확성 개선을 위한 l-확장 랜덤대치는 앞서 제안된 랜덤대치의 프라이버시 수준을 유지하면서 보다 정확하게 원본 데이터의 분포를 재구축할 수가 있다. 이로 인해 연관규칙 마이닝, 의사결정나무 마이닝 등 여러 데이터 마이닝 기법에서 SMC등과 결합되어 보다 실용적으로 사용될 수 있다. 하지만 정확도가 증가하는 만큼의 계산량이 늘어나기 때문에 앞으로도 정확도와 안전성, 그리고 계산량에 대한 심도 있는 연구가 지속적으로 수행되어야 할 것으로 보인다.

참고문헌

[1] Shipra Agrawal, and Jayant R. Haritsa, "A Framework for High-Accuracy Privacy-Preserving Mining", Proc. of ICDE 2005, 2005.

[2] Jim Dowd, Shouhuai Xu, and Weining Zhang, "Privacy-Preserving Decision Tree Mining Based on Random Substitutions", ETRICS2006, LNCS 3995, Springer-Verlag, pp. 145-159, 2006.

[3] A. Evfimievski, J. Gehrke, and R. Srikant, "Limiting Privacy Breaches in Privacy Preserving Data Mining", Proc. of ACM Symp. on Principles of Database Systems (PODS), 2003.

[4] 강주성, 안아론, 홍도원, “행렬기반 랜덤화를 적용한 프라이버시 보호 기술의 안전성 및 정확성 분석”, 한국정보보호학회논문지, 제18권 4호, 2008, pp. 53-68.