The Journal of The Institute of Internet, Broadcasting and Communication (IIBC)
Vol. 19, No. 5, pp.229-238, Oct. 31, 2019. pISSN 2289-0238, eISSN 2289-0246

https://doi.org/10.7236/JIIBC.2019.19.5.229
JIIBC 2019-5-32

빅데이터 기반 대학생 학자금 대출 현황 분석

Analysis of Current Situation of University Student Loans Based on Bigdata

김정준[*], 장성준[**], 이용수[***]

Jeong-Joon Kim[*], Sung-Jun Jang[**], Yong-Soo Lee[***]

요 약 한국장학재단에서 학자금 대출제도가 시행되기 전에는 은행 등 금융회사를 통해 학자금 대출제도가 시행되고 있었지만, 한국장학재단 설립된 이후는 재단이 직접 학자금을 대출함으로써 정부의 역할은 강화되었다. 하지만, 학자금 대출의 상환실적이 저조하여 향후 학자금 대출의 상당한 부실과 정부의 재정적 부담이 우려되고 있다. 더구나 학자금 대출은 저소득층 지원을 위하여 대학을 졸업한 이후에도 상환이 이루어지기 때문에 채무자의 향후 취업률과 소득수준이 개선되지 않는 한 학자금 대출의 상환율이 개선될 가능성은 매우 희박하다. 본 논문에서는 빅데이터 기반 시스템에서 수집, 저장, 처리, 분석 단계를 거쳐 학자금 대출의 상환 금액을 최종 시각화 그래프를 표현하였다. 이는 학자금 대출에 대한 금액을 눈으로 확인하여 현재 학자금 대출제도에 대한 부담을 줄일 방안을 다양하게 생각해 낼 수 있는 근거자료가 될 수 있다.

Abstract Before the scholarship loan system was implemented at the Korea Scholarship Foundation, the government's role was strengthened by the direct lending of student funds to banks and other financial institutions. However, the low repayment performance of student loans has raised concerns over the future of student loans and the government's financial burden. Moreover, since student loans are repaid even after graduating from college to support low-income families, it is highly unlikely that the repayment rate of student loans will improve unless the employment rate and income level of the borrower improve. In this paper, the final visualization graph is presented of the repayment amount of the student loan through the collection, storage, processing and analysis phase in the Big Data-based system. This could be the basis for visually checking the amount of student loans to come up with various ways to reduce the burden on the current student loan system.

Key Words : Big Data, Hive, Scholarship Foundation, Amout Analysis

Ⅰ. 서 론

학자금 대출을 이용한 대학생들은 졸업할 경우 큰 금액의 빚을 지고 사회생활을 시작해야 하는 부담이 있다. 이에 본 논문에서는 학자금 대출 현황을 분석하여 대학생들을 대상으로 부담을 줄일 방안을 모색한다. 학자금

[*]정회원, 한국산업기술대학교 컴퓨터공학과 조교수
[**]정회원, 여주대학교 소프트웨어융합과 교수
[***]정회원, 여주대학교 소프트웨어융합과 교수
접수일자 2019년 7월 10일, 수정완료 2019년 8월 10일
게재확정일자 2019년 10월 4일

Received: 10 July, 2019 / Revised: 10 August, 2019 / Accepted: 4 October, 2019
[**]Corresponding Author: diclee@yit.ac.kr
Dept. of Computer Engineering, Korea Polytechnic University, Korea.

대출 현황 데이터는 공공 데이터 포털(www.data.go.kr)에서 수집하였다.

수집한 데이터는 일반 상환과 취업 후 상환이 있으며 세부적인 기준으로 대학소재지와 주민등록상 거주지로 나뉜다. Hive를 사용한 처리 과정을 수행하기 위해 윈도우에 있는 csv파일을 리눅스로 이동시킨 후 하둡 공간 내부로 복사하였다. 처리 과정을 통해 필요한 데이터 컬럼만 골라내 테이블을 구성, 새로운 파일로 저장하였다[1].

다음 단계는 상세한 분석 및 표현 작업을 위해 Windows 환경에서 R Studio를 사용하였다. 위에서 말한 일반 상환과 취업 후 상환으로 나뉘어 있는 두 가지 데이터를 통합하는 작업을 비롯하여 필요한 데이터로 정제하는 작업들을 다수 거친 후, 쉽게 알아볼 수 있도록 그래프를 통한 도식화를 진행하여 결과물을 도출하였다[2].

II. 관련 연구

1. 공공 데이터 포털

학자금대출 현황 데이터는 공공 데이터 포털(www.data.go.kr)에서 수집하였다. 공공 데이터 포털은 국가기관, 지방자치단체, 공공기관 등이 법령 등에서 정하는 목적을 위하여 생성 또는 취득하여 관리하고 있다.

데이터베이스, 전자화된 파일 등 광(光) 또는 전자적 방식으로 처리된 공공 데이터를 민간에 제공함으로써, 민간 활용을 통한 신규 비즈니스와 일자리 창출, 국민편익을 향상하기 위한 사이트이다[3-4].

2. Hortonwork Sandbox

빅데이터의 처리과정은 대체적으로 수집, 저장, 처리, 분석 및 시각화 5단계로 구성되며, 각 단계 마다 오픈소스로 이루어진 파일을 다운로드 받아 사용 하고자 하는 PC의 환경에 맞추어 설치가 진행된다. 하지만, 기본적인 컴파일 단계나 환경설정의 어려움을 느끼는 사용자를 위해 Hortonwork에서 제공하는 Sandbox를 사용하면 편리하게 빅데이터의 처리과정 중 수집, 저장, 처리 단계를 이용할 수 있다[5].

본 논문에서 활용한 Hortonwork SandBox의 버전은 2.1이며, 수집에 필요한 스쿱(Sqoop), 플룸(Flume), 저장에 필요한 하둡(Hadoop), 처리에 필요한 피그, 하이브(Pig, Hive) 등 다양한 오픈소스 도구를 하나로 묶어 실행만 하면 손쉽게 빅데이터 플랫폼을 사용할 수 있다[6-7].

3. R Studio

분석에 사용한 프로그램은 R Studio를 사용하였으며, R은 오픈소스로 이루어져 무료로 사용이 가능하고, 다양한 라이브러리를 지원하기 때문에 대규모 데이터의 분석 결과를 직관적으로 이해할 수 있는 시각화 기능이 뛰어나다[8].

다양한 운영체제(윈도우, 유닉스, 리눅스, 맥 등)에서 구동이 가능해 접근성이 편리하며, 빅데이터 플랫폼에서 제공하는 다양한 도구들과 연동하여 데이터를 손쉽게 불러오고, 현재 많이 사용되는 정형 데이터베이스 중 하나인 MySQL과 연동을 통한 데이터 분석 및 시각화가 가능하기 때문에 본 논문에서 분석 프로그램으로 R Studio를 사용하였다[9].

III. 대학생 학자금 대출 현황 분석

본 논문에서 제공하는 학자금 대출 현황 분석 프로세스는 다음과 진행된다.

'수집'은 공공 데이터 포털(www.data.go.kr)에서 수집되며, '저장'은 수집된 데이터를 Mount를 통해 Sandbox 내의 하둡에 저장되고, '처리'는 Sandbox 내부의 Hive를 이용하여 처리 하였다. 마지막 '분석'은 R studio를 이용하여 최종 결과물 데이터를 보다 쉽게 이해할 수 있도록 그래프로 도식화 하였다.

1. 데이터 수집

본 논문에서 분석 하고자하는 학자금 대출 현황의 데이터는 공공 데이터 포털에서 조회후 다운로드하여 수집하였다. 수집된 데이터는 2015년에 업로드된 학자금 대출 통계정보이며, URL은 다음과 같다.

https://www.data.go.kr/dataset/fileDownload.do?atchFileId=FILE_000000001389858&fileDetailSn=1

수집한 데이터는 주민등록상 거주지(일반 상환, 취업 후 상환), 대학소재지(일반 상환, 취업 후 상환) 4가지의 csv 파일을 이용하였다.

2. 데이터 저장

데이터 저장은 SandBox 내부의 HDFS를 이용하였으며, 윈도우에서 수동으로 다운로드하여 수집한 데이터를 SandBox로 연결하기 위한 작업은 그림 1과 같다.

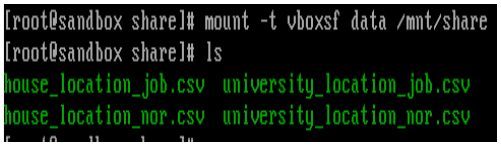

그림 1. Sandbox 공유 폴더 마운트
Fig. 1. Mount Sandbox Shared Folders

윈도우에 수집한 데이터가 있는 data 디렉토리와 Sandbox내의 /mnt/share 디렉토리를 마운트 하여 데이터를 HDFS로 전달할 작업을 진행하였다.

저장 후 ls 명령어로 확인하였으며 저장된 파일은 house location job/nor.csv(주민등록상 거주지 취업 후 상환, 일반 상환), university_location_job/nor.csv(대학소재지 취업 후 상환, 일반 상환)으로 구분된다.

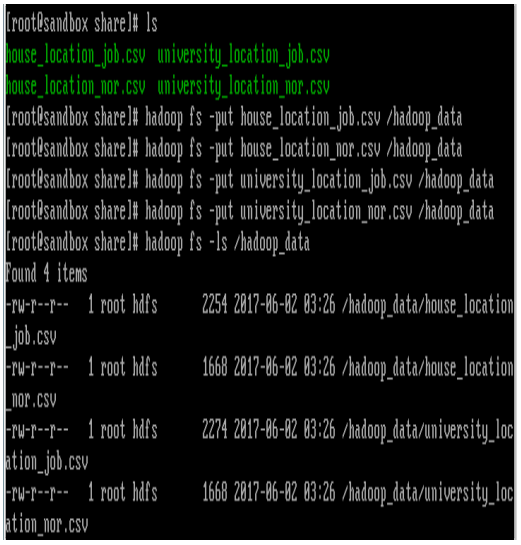

그림 2. 리눅스에서 하둡으로 파일 이동
Fig. 2. Move files from Linux to Hadoop

리눅스내에 파일 이동이 완료된 후 분산 저장 하기 위해서는 하둡으로 파일을 업로드 필요가 있다. 업로드 하는 명령어는 그림 2의 'hadoop fs -put 파일명 /업로드할 위치' 의 명령어를 통해 'hadoop fs -put house location job.csv /hadoop_data' 명령어로 HDFS내로 파일을 업로드 하였다.

업로드 후 리눅스 명령어와 유사하게 'hadoop fs -ls /hadoop_data' 명령어를 통해 확인해보면 파일이 업로드 된 것을 확인 할 수 있으며, 나머지 3개의 파일도 동일하게 HDFS내로 업로드 한다.

3. 데이터 처리

데이터 처리에는 Hive를 사용하였으며, Hive는 SQL과 유사한 문법과 형태를 가지고 있기 때문에 편리하게 사용할 수 있다.

우선 그림3과 같이 Hive를 실행하여 프롬프트가 활성화 된 것을 확인한 후 HDFS에 저장된 데이터를 로드하기 위한 테이블 생성을 그림3과 같이 한다.

그림 3. 하이브에서 테이블 생성
Fig. 3. Create Table in Hive

생성하는 테이블명은 house_nor이며, 2011년부터 2015년 까지의 정수형 데이터를 저장하기 때문에 그림 3과 같이 생성하고, csv형식의 데이터는 데이터의 구분이 ',' 콤마를 이용하여 구분하기 때문에 ROW FORMAT DELIMITED FIELDS TERMINATED BY 구문을 통해 콤마를 기준으로 데이터를 구분한다는 옵션을 준다.

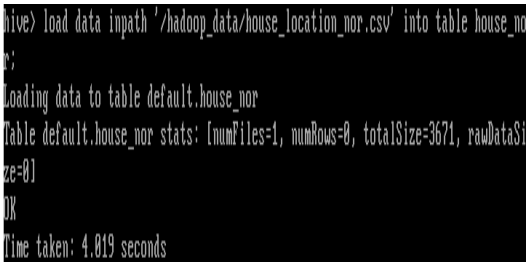

그림 4. HDFS 데이터를 Hive로 입력
Fig. 4. Enter HDFS data as Hive

그림 3을 통해 생성되는 house_nor 테이블에 데이터를 입력하는 명령어는 "load data inpath '/hadoop/data/house location nor.csv' into table house_nor;"와 같다.

데이터 저장 단계에서 진행하여 HDFS 내에 있는 /hadoop_data/house_location_nor.csv 파일을 Hive에

서 생성한 house_nor 테이블로 저장하고, 저장된 테이블을 select한 결과는 그림 5와 같다.

```
248     13886   7193    5893    11816   6891    4925
Jeonnam_Person  8026    3676    4350    4608    2460    2148    4216    2103    2
113     4360    2313    2047    3886    2229    1657
Gyeongbuk_Amount        46520   22127   24392   25620   13988   11632   24268   1
2850    11418   22948   12390   10558   22114   13315   8799
Gyeongbuk_Person        13799   6536    7263    7634    4115    3519    6943    3
573     3370    6755    3515    3240    6415    3785    2630
Gyeongnam_Amount        60868   29038   31829   30524   16893   13631   29046   1
5467    13579   29169   15963   13206   27170   16783   10387
Gyeongnam_Person        18561   8809    9752    9551    5173    4378    8634    4
410     4224    8960    4751    4209    8176    4934    3242
Gangwon_Amount  31011   14508   16504   17202   9482    7719    16481   8939    7
543     16294   8861    7433    15616   9394    6221
Gangwon_Person  9402    4386    5016    5237    2840    2397    4720    2474    2
246     4852    2521    2331    4623    2732    1891
Jeju_Amount     8815    4193    4622    5217    2993    2224    5510    3125    2
385     5330    3111    2219    5157    3010    2147
Jeju_Person     3066    1478    1588    1664    945     719     1662    923     7
39      1620    930     690     1566    905     661
Sejong_Amount   0       0       0       647     0       647     1362    715     6
47      1520    774     746     1994    1017    977
Sejong_Person   0       0       0       184     0       184     381     189     1
92      416     207     209     543     271     272
Time taken: 1.535 seconds, Fetched: 37 row(s)
hive> _
```

그림 5. 하이브에 저장된 데이터를 Select 결과
Fig. 5. The result of selecting the data stored in the Hive

그림 5는 house_nor 테이블에 저장되어있는 데이터를 Select 쿼리를 이용한 결과이다. 경북, 경남 등 대한민국 각 지역에서 학자금 대출 상환금액과 개인에게 제공해준 금액을 확인할 수 있다.

```
hive> create table house_location_nor(
    > Classification string,
    > 2011Total int,
    > 2012Total int,
    > 2013Total int,
    > 2014Total int,
    > 2015Total int)
    > ROW FORMAT DELIMITED FIELDS TERMINATED BY ',';
OK
Time taken: 0.898 seconds
hive> insert overwrite table house_location_nor
    > select Classification, 2011Total, 2012Total, 2013Total, 2014Total, 2015Tot
al from house_nor;
```

그림 6. 원하는 데이터만 다른 테이블로 저장
Fig. 6. Save only the desired data to a different table

전체 데이터가 저장되어 있는 house_nor 테이블에서 년도마다 학자금 대출 금액의 총액을 저장할 수 있는 테이블(house_location_nor)을 새로 생성한다. 처음에 생성한 house_nor 테이블과 유사하게 ',' 콤마를 기준으로 데이터를 구분하기 때문에 "ROW FORMAT DELIMITED FIELDS TERMINATED BY ',';" 옵션을 지정한다.

```
hive> insert overwrite local directory '/mnt/share/data2'
    > row format delimited fields terminated by ','
    > select * from house_location_nor;
```

그림 7. 새로 정제한 테이블 데이터를 리눅스로 이동
Fig. 7. Move newly purged table data to Linux

그림 7은 최종 정제된 데이터를 R Studio에서 분석을 하기 위해서 리눅스 로컬로 데이터를 이동하는 명령어이다. 그림 7 과정을 통해 최종 정제된 데이터를 오픈하여 확인해본 결과는 그림 8과 같으며, 처리 과정을 4번 반복하여 4개의 최종 정제된 파일을 얻을 수 있다.

	A	B	C	D	E	F
	Seoul_Am	453610	234981	226576	237164	229526
	Seoul_Per	110196	55077	49660	51328	49780
	Busan_Am	113740	48801	40961	41324	40607
	Busan_Per	33607	15056	11743	12189	12041
	Daegu_Am	83382	34982	30067	30687	30713
	Daegu_Pe	24104	10307	8440	8786	8717
0	Incheon_A	119946	54683	50041	49906	47965
1	Incheon_P	31678	14592	12422	12536	11990
2	Gwangju_	42377	22302	20591	20659	20569
3	Gwangju_	13019	6736	5847	5989	5838
4	Daejeon_A	47886	23727	22989	22654	22579
5	Daejeon_F	13988	7112	6440	6504	6452
6	Ulsan_Am	20648	10154	9865	10009	9481
7	Ulsan_Per	5880	3074	2826	2881	2689
8	Gyeonggi_	433218	229714	224523	227920	216001
9	Gyeonggi_	111425	59045	54496	55421	51909
0	Chungbuk	29991	15283	14024	14368	14117
1	Chungbuk	9001	4623	4059	4208	4103
2	Chungnan	39608	21995	20974	20394	19609
3	Chungnan	11069	6332	5743	5751	5399
4	Jeonbuk_A	42335	22110	20586	19626	19814
5	Jeonbuk_F	12921	6783	5822	5759	5705
6	Jeonnam_	24024	13563	13064	13086	11816
7	Jeonnam_	8026	4608	4216	4360	3886
8	Gyeongbu	46520	25620	24268	22948	22114
9	Gyeongbu	13799	7634	6943	6755	6415
0	Gyeongna	60868	30524	29046	29169	27170
1	Gyeongna	18561	9551	8634	8960	8176
2	Gangwon_	31011	17202	16481	16294	15616
3	Gangwon_	9402	5237	4720	4852	4623
4	Jeju_Amou	8815	5217	5510	5330	5157
5	Jeju_Perso	3066	1664	1662	1620	1566
6	Sejong_Ar	0	647	1362	1520	1994
7	Sejong_Pe	0	184	381	416	543
8						

그림 8. 최종 정제된 데이터
Fig. 8. Final Refined Data

4. 분석 및 시각화

분석 및 시각화 단계에서는 R 프로그래밍을 이용하고, 분석 전 최종 정제된 파일 4가지(주민등록상 거주지 취업 후 상환, 일반 상환, 대학소재지 취업 후 상환, 일반 상환)를 주민등록상, 대학소재지로 묶어 2개의 데이터프레임으로 만든 코드는 그림 9와 같다.

```
1  university_location_job =
   read.csv("university_location_job");
   university_location_nor =
   read.csv("university_location_nor");

2  university_location = university_location_job +
   university_location_nor;

3  university_location[,1] <- university_location_job[,1];

   write.csv(university_location, file =
4  "university_location_test.csv", row.names = F);
```

그림 9. 대학소재지 데이터프레임 코드
Fig. 9. Code of data frames at university site

1) university_location_job/nor 변수에 최종 정제된 대학소재지 취업 후 상환, 일반 상환 두 파일을 변수에 저장한다. 2) university_location 변수에 취업 후 상환, 일반 상환 두 테이블을 통합한다. 통합하면 문자열은 덧셈을 할 수 없으므로, 문자열로 구성된 1열의 값이 이상해 진다. 3) 이상해진 1열의 컬럼을 수정하기 위해 university_location의 첫 번째 열에 기존에 존재하던 데이터프레임의 1열을 다시 덮어 씌어 준다. 4) 통합된 데이터프레임을 university_location_test.csv로 다시 파일을 추출한다.

위 데이터프레임 생성 과정을 동일하게 주민등록상 거주지 취업 후 상환, 일반 상환 두 파일에도 적용하여 하나의 데이터프레임으로 만들고, 확인한 결과는 다음 그림 10 및 그림 11과 같다.

대학소재지 및 주민등록지 통합 상환 파일을 이용하여 막대그래프을 이용하여 연도별 통합 금액을 확인하였으며, 코드는 그림 12와 같다.

	Classification	X2011Total	X2012Total	X2013Total	X2014Total	X2015Total
1	Total_Amount	2685314	2326473	2552082	2421648	2125398
2	Total_Person	733534	727667	784800	783722	712679
3	Seoul_Amount	692492	554587	600509	582732	512364
4	Seoul_Person	173166	155678	166364	166275	150104
5	Busan_Amount	204013	162951	174903	160232	137891
6	Busan_Person	60183	56086	58464	57441	51572
7	Daegu_Amount	156268	125915	132138	117312	99373
8	Daegu_Person	45714	42199	43541	40711	35997
9	Incheon_Amount	199886	175377	191531	179390	153851
10	Incheon_Person	52824	53043	57219	56458	50722
11	Gwangju_Amount	73852	63682	69845	65336	58464
12	Gwangju_Person	23086	22581	24474	24174	22185
13	Daejeon_Amount	86891	70408	77121	72042	64907
14	Daejeon_Person	25405	24001	25670	25344	23542
15	Ulsan_Amount	37548	32576	36032	32995	28134
16	Ulsan_Person	10647	10739	11493	11216	9872
17	Gyeonggi_Amount	699991	641828	718137	694677	613060
18	Gyeonggi_Person	180894	188333	207839	211197	194110
19	Chungbuk_Amount	55350	48671	52254	49898	45210

그림 10. 대학소재지 통합 상환 파일
Fig. 10. University Location Integrated Payback File

	Classification	X2011Total	X2012Total	X2013Total	X2014Total	X2015Total
1	Total_Amount	2685314	2326473	2552082	2421648	2125398
2	Total_Person	733534	727667	784800	783722	712679
3	Seoul_Amount	658897	603569	664276	676499	618541
4	Seoul_Person	162331	162405	177299	183811	171389
5	Busan_Amount	208586	168456	181819	166428	140668
6	Busan_Person	62718	59989	62730	62001	55091
7	Daegu_Amount	98761	78656	85966	79029	67825
8	Daegu_Person	30306	28375	29689	28735	25830
9	Incheon_Amount	76751	69001	74826	67615	57131
10	Incheon_Person	21480	22329	23986	22956	20111
11	Gwangju_Amount	77448	70618	77687	75191	66344
12	Gwangju_Person	23969	24902	27138	27657	25078
13	Daejeon_Amount	112011	92706	102956	94953	84653
14	Daejeon_Person	32060	32096	34760	34458	31900
15	Ulsan_Amount	19421	16848	18454	17832	15721
16	Ulsan_Person	5547	5610	6106	6185	5611
17	Gyeonggi_Amount	554586	492135	537306	503138	435222
18	Gyeonggi_Person	142737	144568	156108	154517	139635
19	Chungbuk_Amount	99430	85117	97143	90938	77492

그림 11. 주민등록상 통합 상환 파일
Fig. 11. Integrated repayment file for resident registration

```
1  house_location <- read.csv("house_location.csv");

2  house_location_ver_1 = data.frame(do.call('rbind',
   strsplit(as.character(house_location$Classification),
   split='_', fixed=T)))

3  house_location_ver_2 = cbind(house_location_ver_1,
   house_location[,-1]);

4  house_location_bar = house_location[-c(1,2),]

5  house_location_bar=house_location_bar[seq(1,nrow(h
   ouse_location_bar),2),];

6  barplot(as.matrix(t(house_location_bar[,-1])),col=c("#
   CEFBC9","#FFC4EB","#FAED71","#FF4848","#368AFF"),
   names=house_location_ver_2[seq(3,nrow(house_locat
   ion_ver_2),2),1], main="주민 등록지 기준", ylab =
   "금액(단위 100만원)", beside = T,ylim = c(0,800000))

7  legend("topright",legend=c("2011","2012","2013","2014
   ","2015"),pch=c(20,20,20,20,20),col=c("#CEFBC9","#FF
   C4EB","#FAED71","#FF4848","#368AFF")
   ,bg="#F6F6F6")
```

그림 12. 주민등록지 기준 금액 코드
Fig. 12. code of the amount based on the resident registration site

1) 그림 11에서 제공되는 데이터(house_location)를 house_location 변수에 저장한다. 2) 1열에 "_"을 기준으로 컬럼을 나눈 데이터프레임을 house_location_ver_1 변수에 저장한다. 3) 2)에서 나눈 데이터프레임을 1)의 열을 제외한 학자금 통환 금액과 합쳐 하나의 데이터프레임을 house_location_ver_2로 저장한다. 4) house location bar 변수에 3)에서 작업한 최상단 1, 2행의 Total은 그래프에 표현하지 않으므로 삭제한다. 5) house_location_bar 변수에 '지역_Person'으로 된 행을 삭제한다. 6) R Studio에서 제공하는 기본 barplot 그래프를 이용하여 년도별로 색을 지정하며, 제목과 x축, y축의 이름을 정하고, y축에서 출력되는 값의 범위를 정한다. 7) 출력된 그래프의 색에 맞추어 그래프 오른쪽 상단에 범례를 표시한다.

그림 13은 주민등록지 기준으로 학자금 상환 금액을 나타냈으며, 여덟 번째 부분인 경기도에서 학자금을 모든 지역에서 최고로 많이 상환했으며, 첫 번째 부분인 서울이 두 번째로 상환 금액이 많았다. 상환 금액이 낮은 지역은 울산, 제주, 세종으로 확인할 수 있다.

그림 12의 코드를 이용하여 이제 대학소재지 기준 통합 상환 금액을 막대 그래프로 표현한다. 코드 자체는 동일하며 그림 12의 1) 설명에서 이야기했던 데이터만 대학소재지 통합 상환 파일로 대처하면 똑같이 구성되기 때문에 코드는 생략하였다. 결과는 그림 14와 같다.

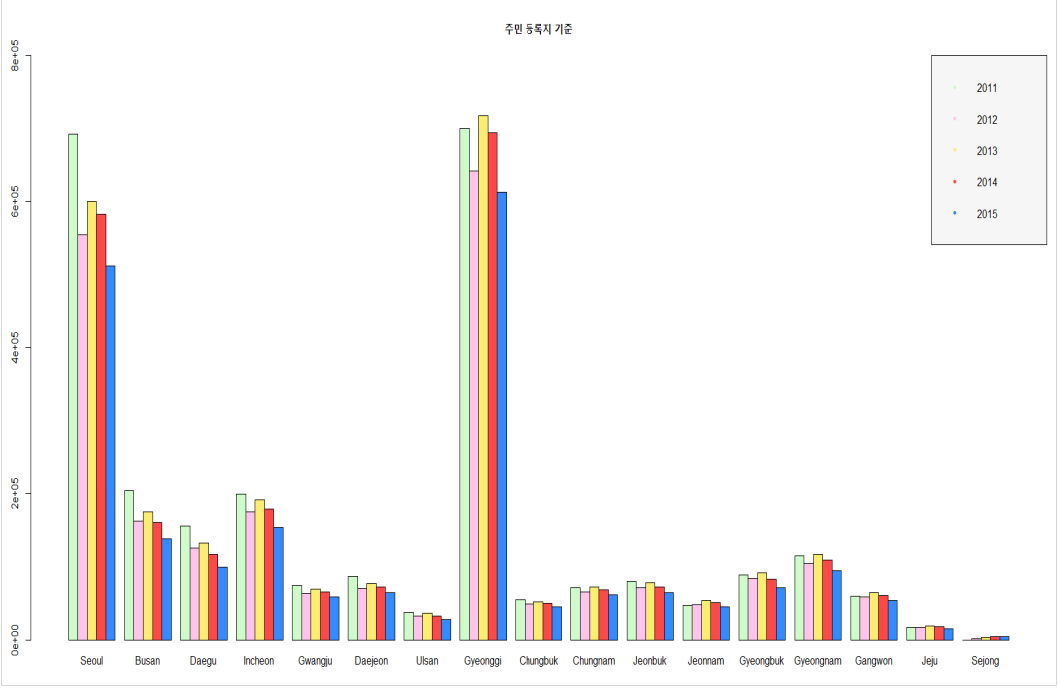

그림 13. 주민등록지 기준 금액 막대그래프 표현
Fig. 13. Bargraph of the amount based on the resident registration site

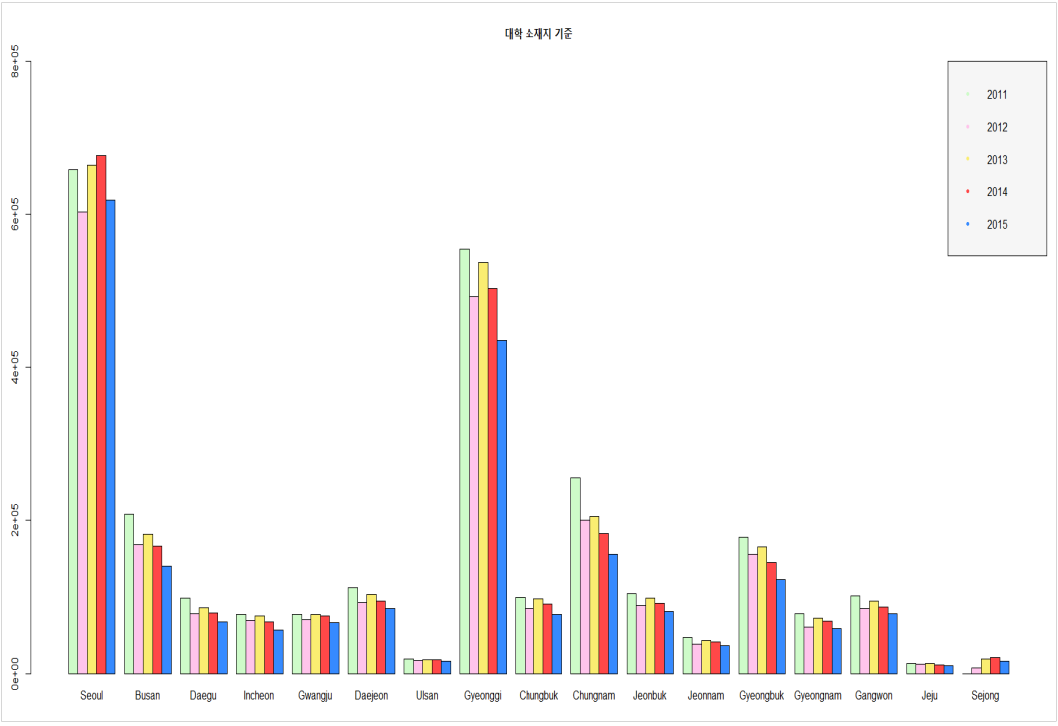

그림 14. 대학소재지 기준 금액 막대 그래프 표현
Fig. 14. Bargraph of the amount based on University Location

그림 14는 대학소재지 기준 통합 상환 금액을 막대 그래프로 표현하였으며, 주민등록지 기준과 다르게 첫 번째로 가장 많은 금액을 상환한 지역은 서울이며, 두 번째가 경기도로 나타났다. 하지만, 낮은 지역은 주민등록지 기준과 동일하게 울산, 제주, 세종이 낮게 표현되었다. 그림 15는 최근 5년간 주민등록지 기준 지역별 금액을 원형 그래프로 표현할 때 사용한 코드이다. 위 코드는 그림 12에서 진행한 코드에서 이어서 진행한다.

1) house_location에 지역별 최근 5년 평균 금액을 가지는 컬럼을 추가하여 house location pie sample 1에 저장한다. 2) house_location_pie_sample_2 변수에 최상단 1, 2행의 Total은 그래프에 표현하지 않으므로 삭제한다. 3) house_location_pie_sample_3 변수에 '지역_Person'으로 된 행을 삭제한다. 4) R Studio에 기본 내장 되어 있는 원형 그래프를 그릴 수 있는 pie를 이용하여 지역별로 색을 다르게 지정하여 표현한다. 5) 출력된 원형 그래프의 우측 상단에 범례를 지정한다.

```
1  house_location_pie_sample_1 =
   cbind(house_location,
   average=rowMeans(house_location[,-1]))
2
   house_location_pie_sample_2 =
   house_location_pie_sample_1[-c(1,2),];
3
   house_location_pie_sample_3 =
   house_location_pie_sample_2[seq(1,nrow(house_loca
   tion_pie_sample_2),2),];
4
   pie(house_location_pie_sample_3$average, col =
   c("#FFD8D8","#FAF4C0","#D4F4FA","#E8D9FF","#F6F6
   F6","#FFC19E","#CEF279","#CEFBC9","#7F7EFF","#DA
   D9FF","#E8D9FF","#FFD9FA","#FFD9EC","#A4FFFF","#F
   FDD73","#F15F5F","#8C8C8C","#1DDB16"),
   main="주민 등록지 기준", labels =
   university_location_ver_2[seq(3,nrow(university_locat
   ion_ver_2),2),1]);
5
   legend("topright",legend=c("최근 5년간 지역별 평균"),
   pch=c(),bg="#F6F6F6")
```

그림 15. 최근 5년간 주민등록지 기준 금액 코드
Fig. 15. Code for the amount based on the resident registration site for the last five years

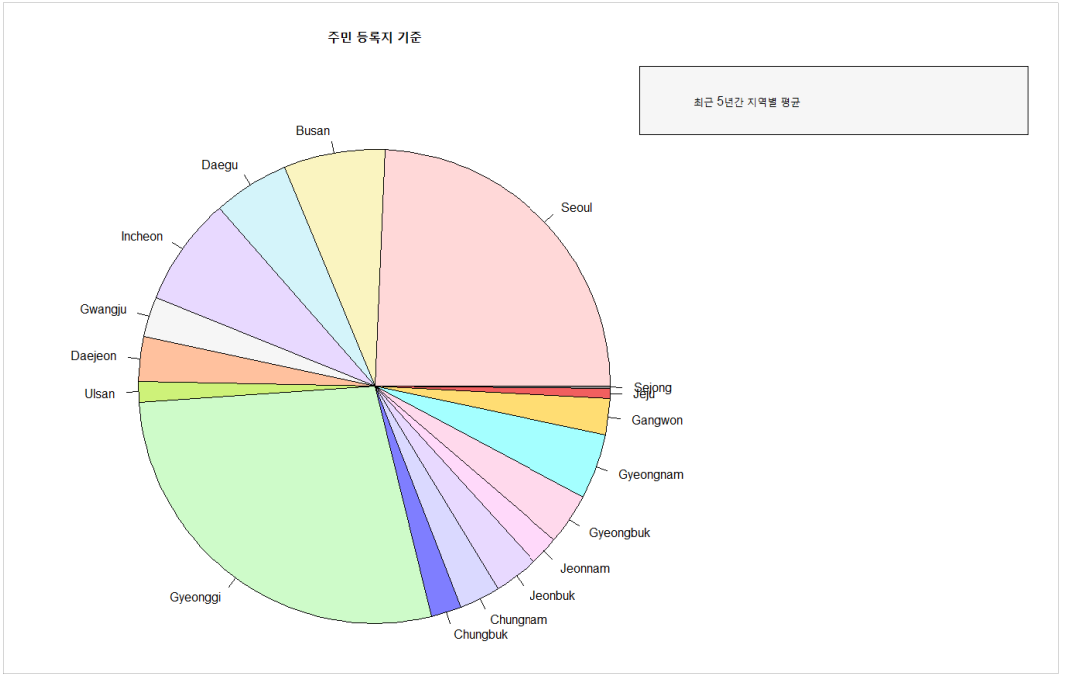

그림 16. 주민등록지 기준 평균 금액 원형 그래프 표현
Fig. 16. Average amount based on resident registration paper circular graph representation

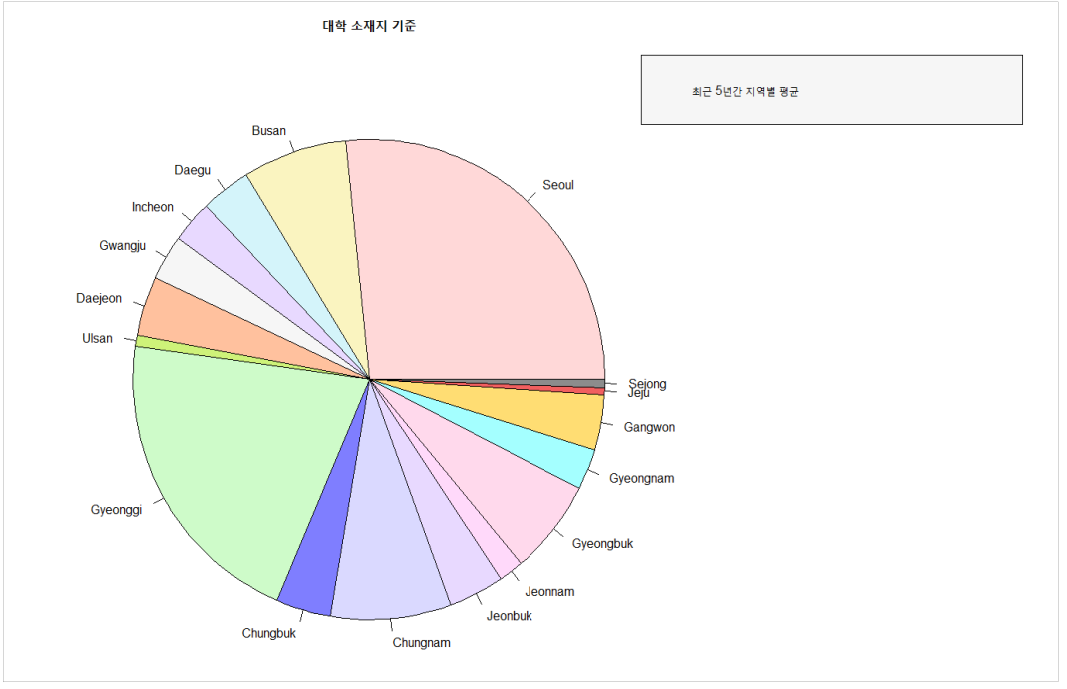

그림 17. 대학소재지 기준 평균 금액 원형 그래프 표현
Fig. 17. Average amount of money based on university site circular graph representation

그림 16은 주민등록지 기준 5년의 평균 상환 금액을 원형 그래프로 표현한 결과이다. 경기도 지역의 평균 상환 금액이 최고로 높았고, 두 번째로 서울이 높은 결과를 확인할 수 있다.

그림 15의 코드를 이용하여 이제 대학소재지 기준 통합 상환 금액을 원형 그래프로 표현한다. 코드 자체는 동일하기 때문에 생략하였다. 그림 17은 대학소재지 기준 5년치의 평균 상환 금액을 원형 그래프로 표현하였으며, 주민등록지 기준과 유사하게 서울과 경기도에서 높은 상환 금액을 확인할 수 있다.

하지만 충남 지역에서 대학소재지가 주민등록지 기준보다 더 높게 나왔으며, 인천지역은 감소하였다.

Ⅳ. 결 론

대학생의 등록금 및 생활비가 포함된 학자금 대출 현황을 볼 수 있다. 학자금 대출을 받는 유형은 일반 상환과 취업 후 상환 두 가지로 분류된다. 두 유형을 합쳐 지역별 학자금 대출 현황을 분석하였다. 지역별로 분석할 경우 주민등록상 거주지와 대학소재지 현황으로 분리하여 분석하였다.

주민등록상 거주지에서는 경기 서울 인천 부산 순으로 학자금 대출을 많이 받고 있으며, 대학소재지에서는 서울 경기 충남 경북 순으로 학자금 대출을 많이 받고 있었다. 대학생들이 졸업하게 되면 상당한 금액의 빚을 지고 사회생활을 시작해야 한다. 이러한 부담을 줄이기 위해 정부의 지원이 조금이나마 더 필요하다고 생각한다.

주민등록상 거주지에서의 지역별 학자금 대출 비율에 따라 대학생들을 대상으로 하는 지역의 동사무소, 시청, 공공기관 등에서 근로 장학생 기회 제공 등 다른 다양한 혜택을 지원의 기대효과를 낼 수 있는 근거자료가 될 수 있다.

대학소재지에서의 지역별 학자금 대출 비율에 따라 대학생들을 대상으로 학교 주변 보증금 지원, 월세 인하 등의 혜택을 줄 수 있다. 지역 순위에 따라 근로 장학생 추가 채용, 기숙사 입주 비용 인하 등의 지원 및 기대효과를 낼 수 있는 근거자료가 될 수 있다.

References

[1] Ashish Thusoo, Joydeep Sen Sarma, Namit Jain, Zheng Shao, Prasad Chakka, Suresh Anthony, Hao Liu, Pete Wyckoff, Raghotham Murthym, "Hive: a warehousing solution over a map-reduce framework", Proceedings of the VLDB Endowment, Vol. 2, No. 2, pp. 1626-1629, Aug 2009.
DOI: https://doi.org/10.14778/1687553.1687609

[2] Yeoung-Eun Yang, "The Study of repayment burdens of student loan program guaranteed by government and Income Contingent Loan by University Type and Income Level", Department of Education The Graduate School of Ewha Womans University, pp. 1-107, Jan 2011.

[3] Man-Jai Lee, "Big Data and the Utilization of Public Data", Internet and information Security, Vol. 2, No. 2, pp. 47-64, Nov 2011.

[4] Hyo-Jin Song, Sung-Soo Hwang, "Seeking Strategies for Local Goverments to Prepare for Public Data Act", The Korean Association for Regional Information Society, Vol. 17, No. 2, pp. 1-28, June 2014.

[5] Jeong-Joon Kim, Kwang-Jin Kwak, Don-Hee Lee, Yong-Soo Lee, "Study of Trust Bigdata Platform," Journal of The Institute of Internet, Broadcasting and Communication, Vol. 16, No. 6, pp. 225-230, Dec, 2016.
DOI: https://doi.org/10.7236/JIIBC.2016.16.6.225

[6] HortonWorks Data Platform (HDP) Support, https://ko.hortonworks.com/services/support/enterprise/

[7] HortonWorks HDP 2.1 Introduction Component, https://ko.hortonworks.com/licenses/

[8] Hadley Wickham, "Advanced R", http://adv-r.had.co.nz/

[9] Jeffrey S. Racine, "R Studio: A Platform-Independent IDE for R and Sweave", Journal of Applied Econometrics, Vol. 27, No. 1, pp. 167-172, Oct 2011.
DOI: https://doi.org/10.1002/jae.1278

저 자 소 개

김 정 준(정회원)

• Jeong-Joon Kim received his BS and MS in Computer Science at Konkuk University in 2003 and 2005, respectively. In 2010, he received his PhD in at Konkuk University. He is currently a professor at the department of Computer Science at Korea Polytechnic University. His research interests include Database Systems, BigData, Semantic Web, Geographic Information Systems (GIS) and Ubiquitous Sensor Network (USN), etc.

장 성 준(정회원)

• Sung-jun Jang received his BS and ME in Electronic Engineering at Incheon University in 1995 and 1997, respectively. In 2008, he received his Ph.D at Incheon University. He is currently a professor at the department of software convergence at Yeojoo Institute of Technology. His study is interested in Memory Device and Network Protocols and Algorithms, etc.

이 용 수(정회원)

• Yong-soo Lee received his MS in Computer Science at Konkuk University in 1989. In 2015, he received his PhD in Information & Control Engineering at Kwangwoon University. He is currently a professor at the Department of software convergence at Yeoju Institute of Technology. He is the Member of the Korea Institute of Internet, Broadcasting & Communication (IIBC). His research interests include Database Systems, Data Mining, BigData, Wireless Sensor Networks and Ubiquitous Sensor Network (USN), etc.