베이지안 통계특론 I 기말 프로젝트 보고서

-데이터사이언티스트 연봉 예측

222STG14 안희성

I. 서론

1. 데이터 소개

Data scientist salary data 를 이용할 예정이다. 이 자료는 2021 년 미국의 채용공고 사이트 glassdoor (https://www.glassdoor.com)에 'data scientist'를 모집한다는 구인공고를 크롤링한 것이다. 변수는 총 42 개로, 채용 공고에 제시된 직위와 연봉, 회사에 관한 정보, 지원자에게 요구되는 데이터분석 툴 스킬, 학위 등이 있다. 관측치는 742 개이다.

[데이터 출처] https://www.kaggle.com/datasets/nikhilbhathi/data-scientist-salary-us-glassdoor

2. 선행연구

Zhen, Raheem(2022)은 tree-method 데이터 마이닝 기법과 KNN method 를 이용하여 데이터 사이언티스트에게 가장 많이 요구되는 분석 툴을 파악하고 그에 따른 연봉을 예측하였다.[1] Zhao(20220)는 BP 인공신경망 모델을 이용하여 데이터 사인언티스트 연봉을 예측하였다.[2]

이 두 연구는 데이터 사이언티스트 연봉 예측에 데이터 마이닝 기법과 머신 러닝 기법을 이용하였다. 본 보고서에서는 이에 베이지안 분석 기법을 적용할 예정이다.

3. 목적 및 기대효과

기업들은 빅데이터의 중요성을 인식하고 대규모 고용량 데이터를 수집하는 데에 큰 투자를 하고 있다. 이를 처리하고 분석하기 위해 데이터 사이언티스트에 대한 수요가 커지고 있다. 이에 반해, 전세계적으로 뛰어난 데이터 사이언티스트의 공급은 부족한 상황이다. 데이터 사이언티스트는 연봉, 경력, 사업장의 위치 등 일반적인 정보 뿐만 아니라 특정 분석 툴을 사용할 수 있는지 여부가 중요하다는 점에서 타직군과 차이를 갖는다. 이 때문에 데이터 사이언티스트 채용에서는 일반적으로 고안되는 연봉 책정 모델을 적용하기 어렵다.

본 보고서에서는 산업 군 별 데이터 사이언티스트의 연봉을 예측할 것이다. 또한, 산업군별로 데이터 사이언티스트로서 습득해야 할 분석 툴의 우선순위를 탐색할 것이다. 이와 같은 분석은 학생들이 데이터 사이언티스트로서의 역량을 키우는 데에 도움을 줄 것이다. 또한, 인사차원에서 기업이 데이터 사이언티스트의 연봉을 책정하는 데에 근거를 제공할 것으로 기대된다.

II. 데이터 전처리 및 EDA

번호	변수명	변수 설명	Type
1	index	인덱스	
2	Job Title	직무명	I
3	Salary Estimate	연봉 범주	
4	Job Description	채용 정보	
5	Rating	기업 평가 점수	numerical

[1] Zhen, Raheem(2022), Salary Prediction in Data Science Field Using Specialized Skills and Job Benefits-A Literature Review, *Journal of Applied Technology and Innovation*

[2] Zhao(2022), Predicting the salary in data science through BP neural network, International Conference on Applied Mathematics, *Modelling, and Intelligent Computing(CAMMIC)*

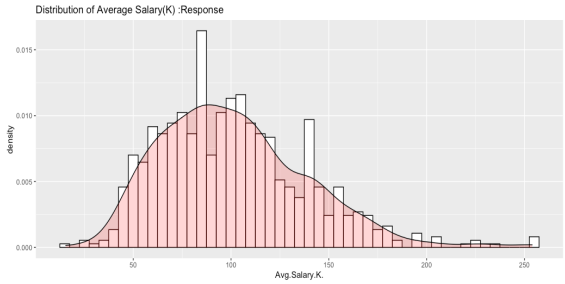
6	Company Name	기업명+평점	
7	Location	도시+state	
8	Headquarters	본사 도시+state	
9	Size	사원 수	categorical
10	Founded	설립 연도	
11	Type of ownership	사기업, 공기업, 정부기관 등	
12	Industry	산업군	
13	Sector	산업군과 유사	categorical
14	Revenue	회사의 매출	categorical
15	Competitors	경쟁사	
16	Hourly	시급 or not(1/0)	categorical
17	Employer provided	고용주가 임금 지급	
18	Lower Salary	채용정보 추출 연봉 하한($)	
19	Upper Salary	채용정보 추출 연봉 상한($)	
20	Avg Salary(K)	채용정보 추출 연봉 평균($)	numerical
21	Company txt	기업명 추출	
22	Job location	State	categorical
23	Age	회사 설립 이후 연 수	numerical
24-39	python, spark, excel, sas 등	요구 skill(1/0)	categorical
40	Job title sim	직무명 단순화	
41	Seniority by title	직급 (senior/junior)	categorical
42	Degree	요구 학위 (학사/석사/박사)	categorical

반응변수는 Avg.Salary.K 로 채용정보에서 추출한 연봉의 평균이다. 설명변수로는 중복정보를 제외하고 유의미한 변수를 선택하였다. 기업 평가 점수, 사원 수, 기관의 형태, 산업 군, 매출 등 회사 관련 정보들과 학위, 각종 분석 툴 스킬 요구 여부 등 직원 정보를 포함하였다.

Hourly==1 인 경우가 24 개 존재하였는데, 시급으로 임금이 제시된 경우 연봉 책정이 어려우므로 제외하였다.

1. 반응변수

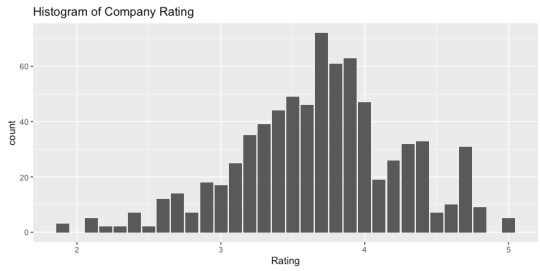
채용정보에서 추출한 연봉 범위에서 하한과 상한의 평균(Avg.salary.k)을 반응변수로 설정하였다. 수치형 변수로, 오른쪽으로 꼬리가 긴 분포를 보인다. 직급과 연봉이 높을수록 수가 적은 것이 일반적인 연봉 분포인데, 이와 상응하는 형태이다.



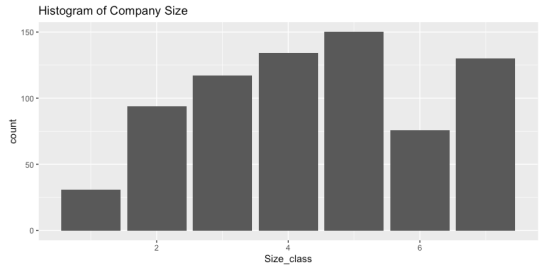
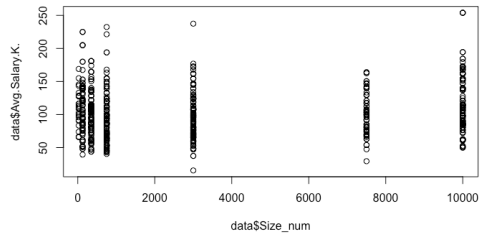
2. 설명변수-회사 정보

회사 관련 변수에는 Rating, Size, Type of ownership, Sector, Revenue, Age 가 있다.

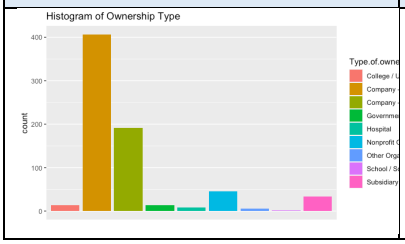
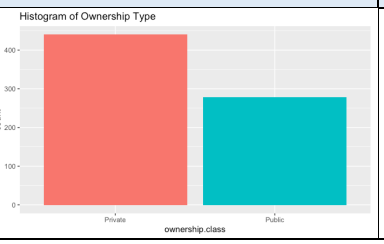
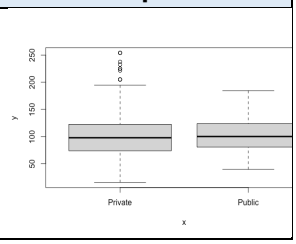
Rating 은 기업 평가 점수를 0~5 점으로 책정한 수치형 변수이다. 평균은 3.69 이고, 중앙값은 3.7 로 거의 유사하였다. NA 가 11 개 존재하였는데 중앙값인 3.7 로 대체하였다.



Size 는 회사의 사원수로, 1~50 / 51~200 / 201~500 / 501~1,000 / 1,001~5,000 / 5,001~10,000 / 10,000+으로 나누어진 범주형 변수이다. 범주 별 평균으로 대체하여 Size_num 이라는 수치형 변수를 생성하였다. Size_num 의 plot 을 보면, 사원 수에 관계없이 연봉 분포가 비슷한 것을 알 수 있다.

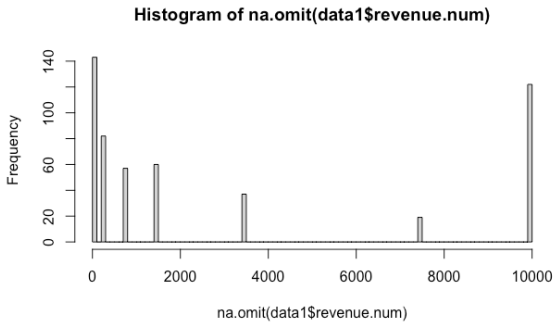
Size(before)	Size_num(after)
	

Type of ownership 은 회사의 형태 정보를 담고 있는 범주형 변수이다. 사기업, 공기업, 정부기관, 병원, 자회사, NGO, 대학, 학교, 기타 기관으로 분류되어 있다. 범주의 개수가 많아 기관의 특성을 기준으로 private(사기업+자회사)과 public(그 외)로 범주를 통합하였다. 기타 기관을 NA 로 볼 수 있는데, 총 5 개로 그 외로 간주하였다.

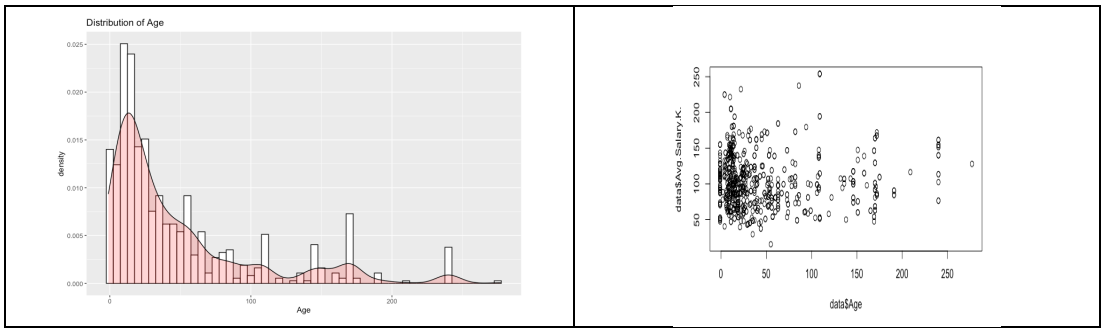
범주 통합 이전	범주 통합 이후	Boxplot
		

Sector 은 산업군 25 가지로 범주형 변수이다. NA 는 10 개인데, 기업명을 검색하거나 job description 을 보고 산업군을 분류하였다.

Revenue 는 회사의 매출 변수로, 1m 이하 / 1m~5m / … / 10b 이상으로 분류된 범주형 변수이다. 범주의 개수는 12 개이다. 평균은 $3182 million 이고, 중앙값은 $750 million 이다. Revenue 의 분포를 보면 일정하지 않고 평균, 중앙값이 대표성을 갖는다고 보기 어렵다. NA 가 204 개인데, 대표성을 갖는 값을 설정하기 어렵기 때문에 변수를 사용하지 않기로 결정하였다.

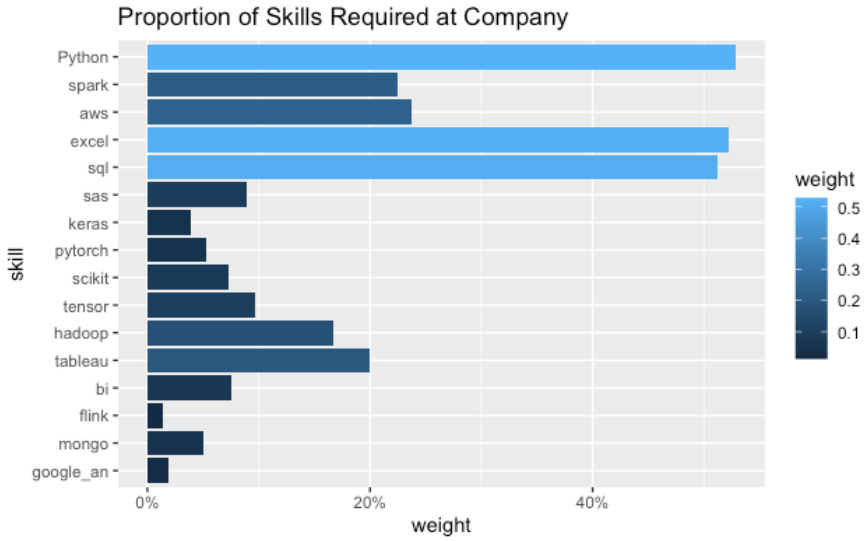


Age 는 회사 설립 이후 연 수로 수치형 변수이다. NA 는 없고, 평균은 47.5, 중앙값은 25 이다. 분포를 보면, 오른쪽으로 꼬리가 긴 형태를 띄고 있다.

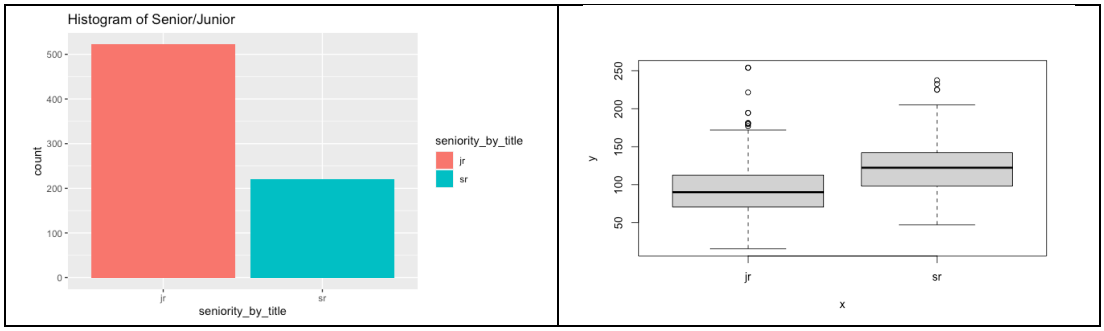


3. 설명변수-직원 정보

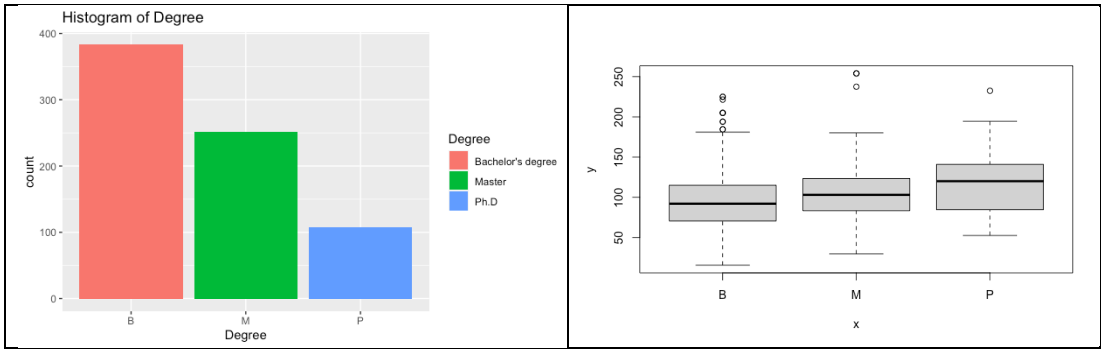
변수 번호 24~39 번은 데이터분석 tool 이름으로, 채용공고에서 해당 tool skill 을 요구하는지 여부에 따라 1/0 으로 분류된 범주형 변수이다. Tool 에는 python, spark, aws, excel, sql, sas, keras, pytorch, scikit, tensor, Hadoop, tableau, bi, flink, mongo, google analysis 가 있다. 아래 그래프는 각 skill 을 요구하는 비율이다. Python, excel, sql 의 비율이 눈에 띄게 높은 것을 확인할 수 있다.



Seniority by title 은 직무명에서 senior/junior 여부를 추출한 것으로, 범주형 변수이다. NA 는 519 개인데, junior 로 간주하였다.



Degree 는 채용공고에서 추출한 요구 학위이다. 석사(M)/박사(P)로 분류된 범주형 변수이다. NA 는 383 개인데, 학사(B) 변수를 추가하여 이로 대체하였다.



III. 데이터 분석

1. 베이지안 변수 선택

GVS(Gibbs Variable Selection) 이용하여 데이터 사이언티스트 연봉 예측 모델을 도출할 것이다. GVS 는 사전분포로 $\pi(\beta_j, \gamma_j) = \pi(\beta_j|\gamma_j)\pi(\gamma_j)$를 가정한다. $\gamma_j = 1$이면 모형에 변수가 선택되는 것이다. $\gamma_j = 1$인 경우 $\pi(\beta_j|\gamma_j) = \pi_1(\beta_j)$로, 이는 β_j가 모형에 선택될 때의 사전분포이다. $\gamma_j = 0$인 경우 $\pi(\beta_j|\gamma_j) = \pi_0(\beta_j)$로, 이는 β_j가 모형에 선택되지 않을 때의 사전분포이다. $\pi_1(\beta_j)$~$N(0, 100*s_j^2)$, $\pi_0(\beta_j)$~$N(\mu_j, s_j^2)$ 로 설정하였는데, β_j 사전분포의 평균과 분산은 다중회귀분석의 coefficient 와 공분산행렬로 지정하였다. γ_j 의 사전분포는 Bernoulli(0.5)로 설정하였다.

다음은 사후확률이 높은 상위 10 개의 변수 선택 결과이다.

	g0	g1	g2	g3	g4	g5	g6	g7	g8	g9	g10	g11	g12	g13	g14	g15	g16	g17	g18	g19	g20	g21	N
1:	1	1	0	0	1	0	1	0	1	1	0	1	0	1	0	0	0	0	0	1	1	0	0.0117
2:	1	0	0	0	1	0	1	0	1	1	0	1	0	1	0	0	0	0	0	1	1	0	0.0104
3:	1	1	0	1	1	0	1	0	1	1	0	1	0	1	0	0	0	0	0	1	1	0	0.0092
4:	1	0	0	0	1	0	1	0	1	1	0	1	1	1	0	0	0	0	0	1	1	0	0.0073
5:	1	0	0	0	1	0	1	0	1	1	0	1	0	1	1	0	0	0	0	1	1	0	0.0072
6:	1	0	0	0	1	0	1	0	1	1	0	1	0	1	0	0	0	0	1	1	1	0	0.0068
7:	1	1	0	0	1	0	1	0	1	1	0	1	0	1	0	0	0	0	1	1	1	0	0.0065
8:	1	0	0	1	1	0	1	0	1	1	0	1	0	1	0	0	0	0	0	1	1	0	0.0065
9:	1	0	0	0	1	0	1	0	1	1	0	1	0	1	0	0	1	0	0	1	1	0	0.0060
10:	1	1	0	1	1	0	1	0	1	1	0	1	1	1	0	0	0	0	0	1	1	0	0.0056

GVS 결과 x1, x4, x6, x8, x9, x11, x13, x19, x20 총 9 개의 변수가 선택되었다. 즉, Rating, python, aws, sql, sas, pytorch, tensor, google_an, seniority_by_title 가 선택되었다. 반면 Stepwise 의 경우, Rating, Size_num, Python, aws, sql, sas, pytorch, scikit, tensor, bi, mongo, google_an, seniority_by_title, ownership.class 총 14 개의 변수가 선택되었다. 베이지안 변수 선택이 좀 더 보수적으로 변수를 선택한 것을 확인할 수 있다.

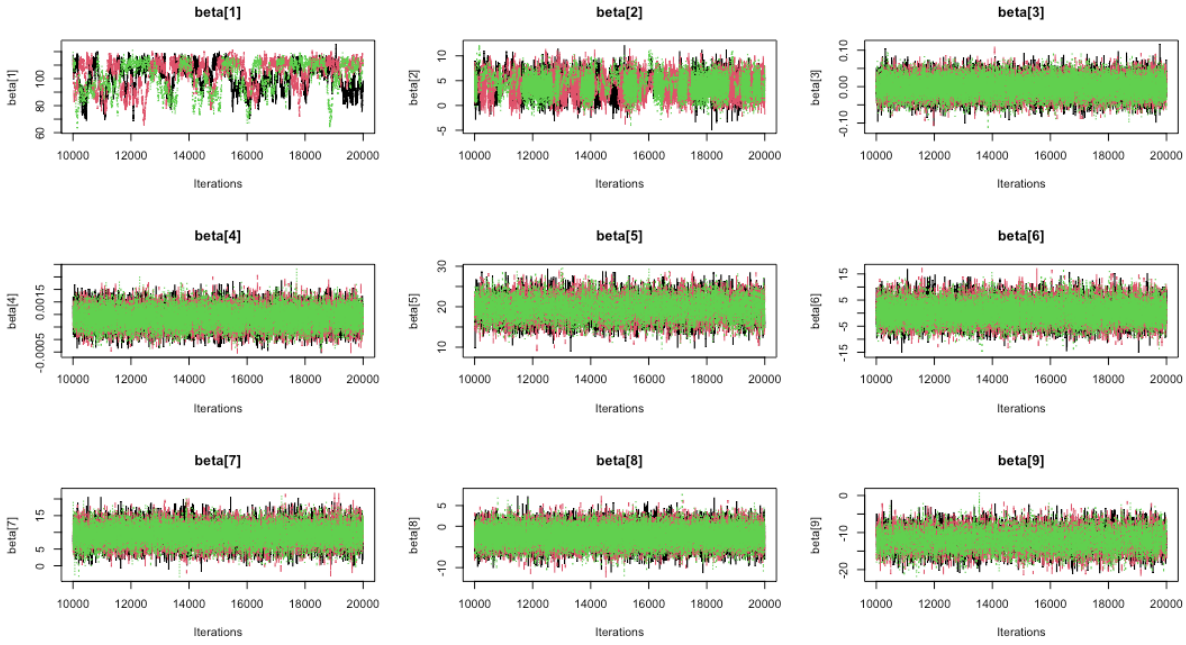
GVS 결과 회귀계수 추정치는 다음과 같다.

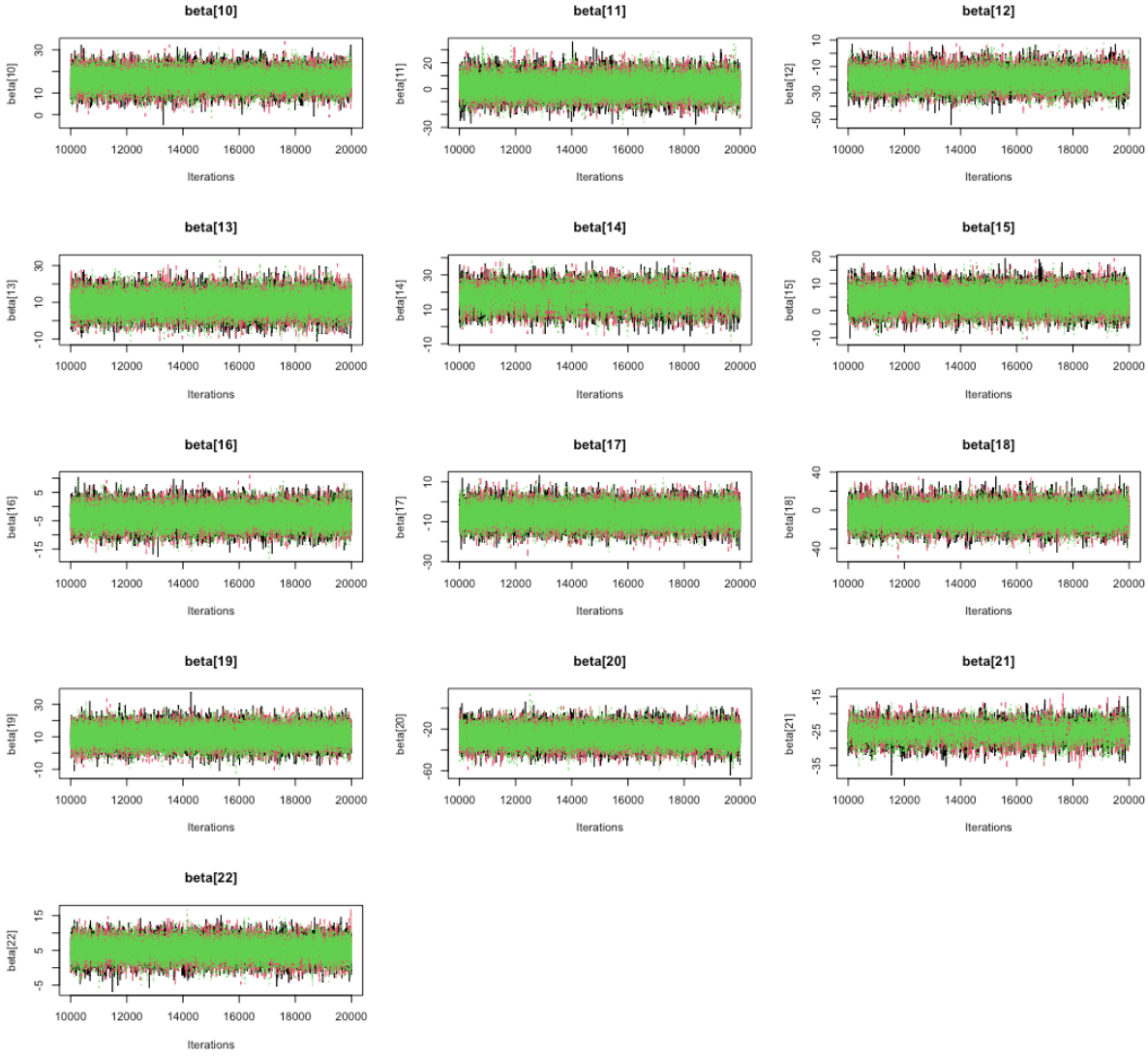
	2.5%	50%	97.5%
β_0	81.2373860	95.755409	112.369329
β_1	0.4055918	4.455359	8.479690
β_4	13.9160289	19.481864	23.981770
β_6	4.3432288	9.860225	15.552224
β_8	-16.9721850	-12.257347	-7.293768
β_9	9.6438761	17.195897	24.928569
β_{11}	-32.4533340	-19.232551	-8.458683
β_{13}	9.4905031	20.140898	29.798065
β_{19}	-42.8462372	-27.124665	-9.802897
β_{20}	-30.8256250	-25.511070	-20.253982

2. 수렴 진단

(1) Trace plot

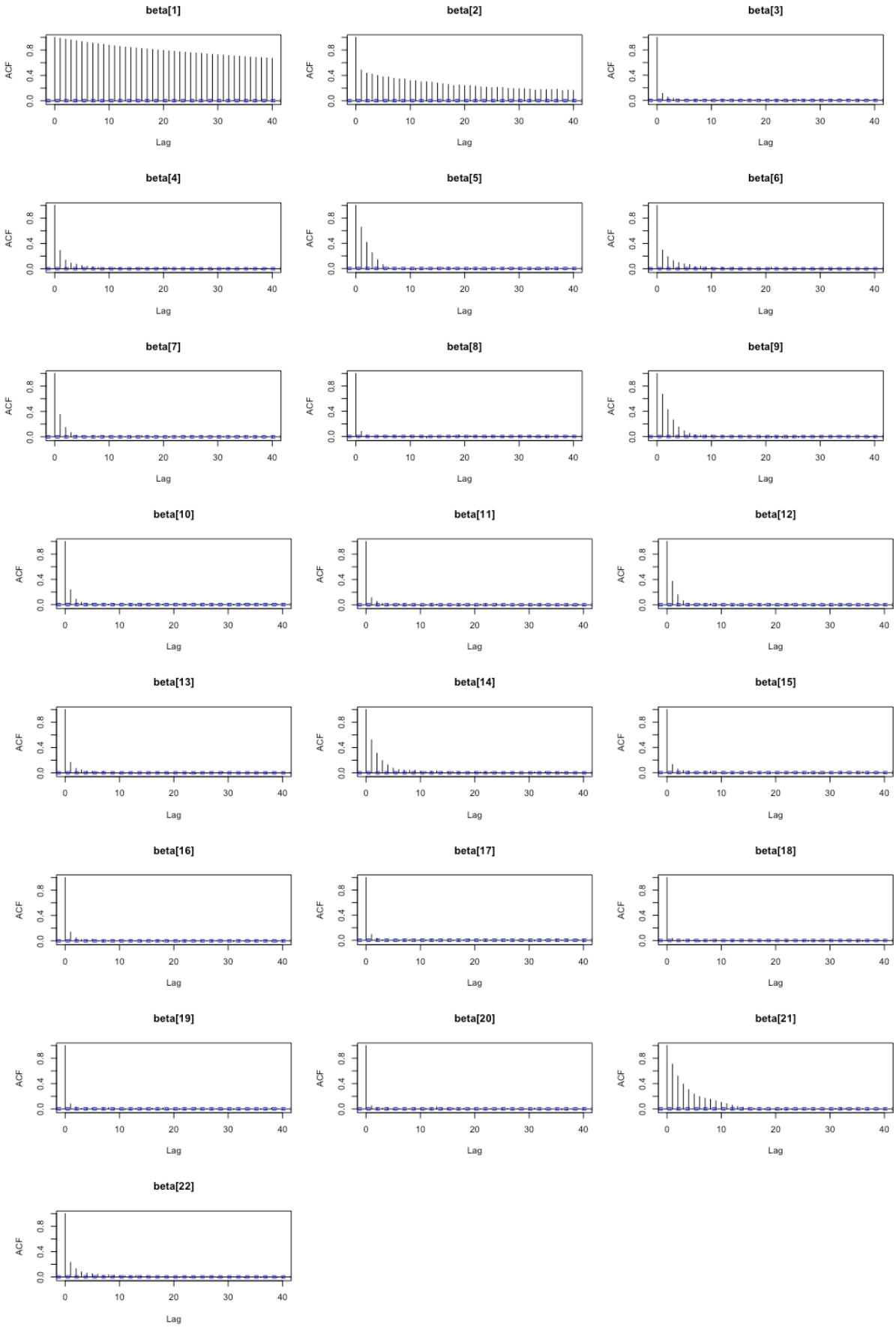
다음은 β의 trace plot 이다. β_1을 제외하고 Trace plot 이 겹쳐지는 것으로 보아 시각적으로 잘 수렴되었음을 알 수 있다. β_1이 다른 coefficient 에 비해 잘 수렴되지 않은 것은 자기상관 때문인 것으로 보인다. 이는 (2)의 ACF plot 에서 확인할 예정이다.





(2) ACF

다음은 β의 ACF plot 이다. β_1, β_2를 제외하면 다른 coefficient 들은 무자기상관인 것을 확인할 수 있다.


beta[1]
beta[2]
beta[3]
beta[4]
beta[5]
beta[6]
beta[7]
beta[8]
beta[9]
beta[10]
beta[11]
beta[12]
beta[13]
beta[14]
beta[15]
beta[16]
beta[17]
beta[18]
beta[19]
beta[20]
beta[21]
beta[22]
ACF
Lag
0.0
0.4
0.8
0
10
20
30
40

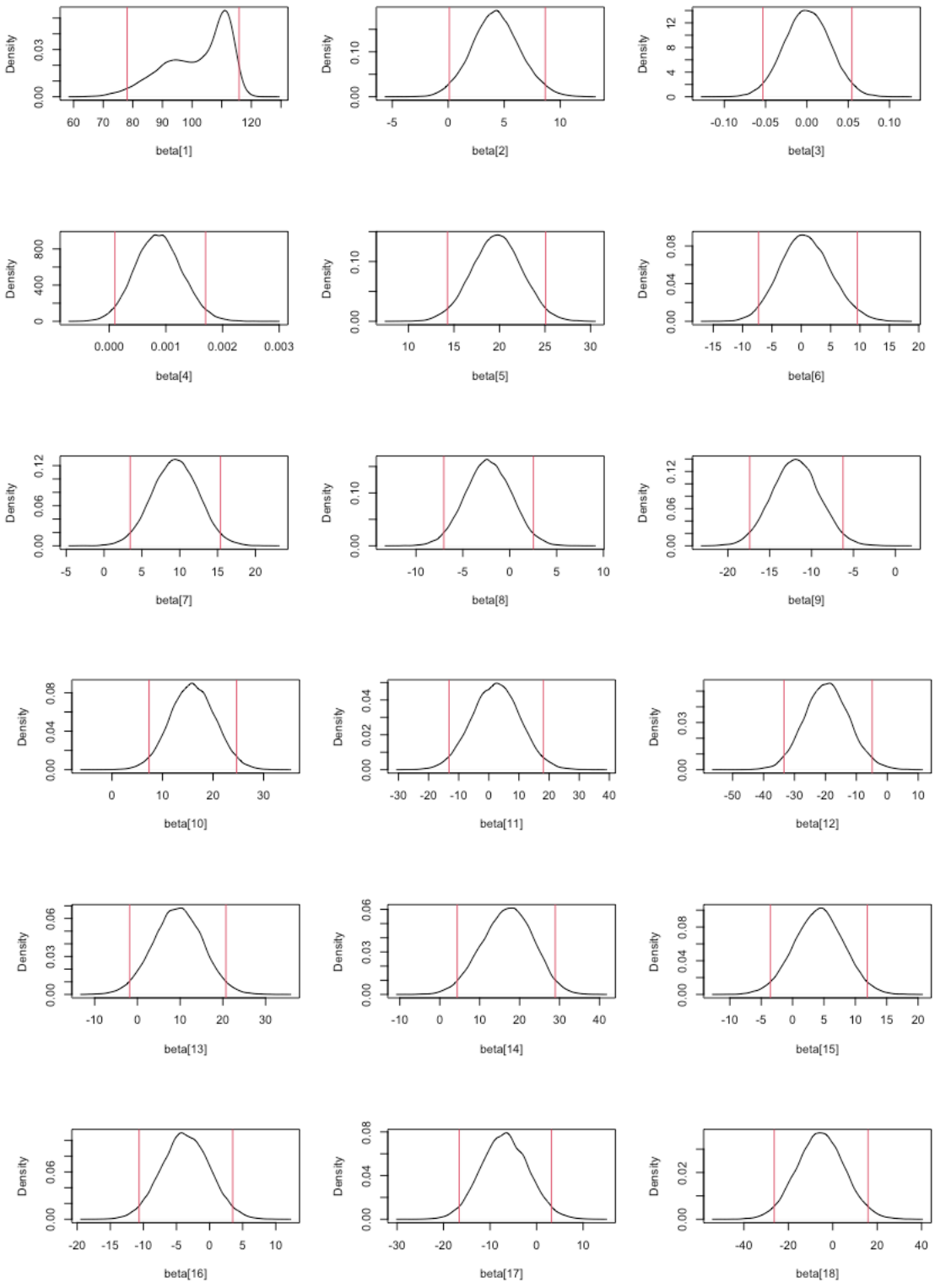
(3) Gelman 상수 및 ESS

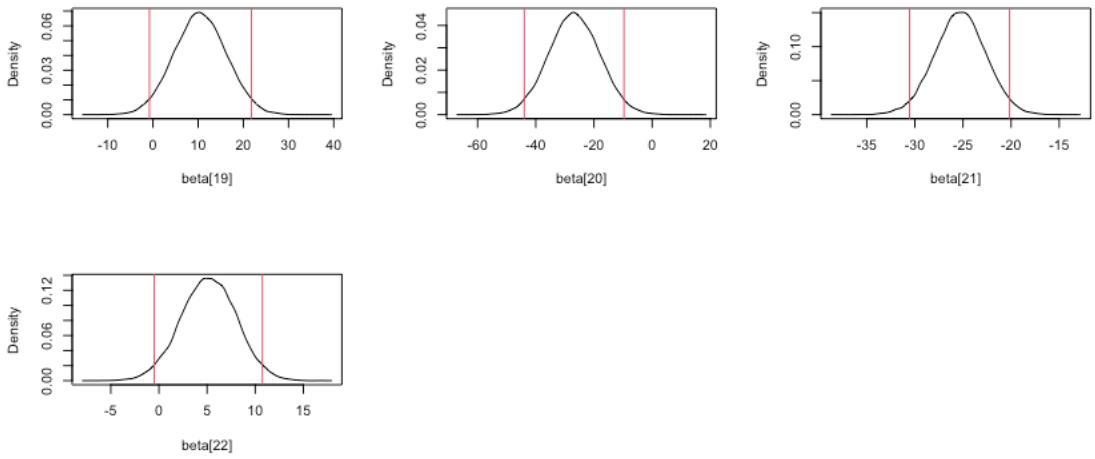
다음은 Gelman 수렴 진단 결과와 ESS 이다. Gelman 상수가 모든 coefficient 가 1.1 보다 작으므로 MCMC 가 잘 수렴되었다고 할 수 있다.

	Point estimate	95% upper	ESS
β_1	1.02	1.07	131.6560
β_2	1.01	1.02	928.6711
β_3	1.00	1.00	20171.6941
β_4	1.00	1.00	12414.1043
β_5	1.00	1.00	7212.7270
β_6	1.00	1.00	10254.1558
β_7	1.00	1.00	13881.3840
β_8	1.00	1.00	24715.4666
β_9	1.00	1.00	6762.6870
β_{10}	1.00	1.00	17754.9112
β_{11}	1.00	1.00	21048.5095
β_{12}	1.00	1.00	12626.5369
β_{13}	1.00	1.00	18031.0104
β_{14}	1.00	1.00	7257.6148
β_{15}	1.00	1.00	20564.0866
β_{16}	1.00	1.00	20832.5018
β_{17}	1.00	1.00	24214.7767
β_{18}	1.00	1.00	28445.1077
β_{19}	1.00	1.00	25160.7232
β_{20}	1.00	1.00	26885.5463
β_{21}	1.00	1.00	4075.5392
β_{22}	1.00	1.00	14554.8600

3. 사후분포

다음은 β 의 주변 사후 밀도 분포이다. β_1 를 제외하고는 모두 정규분포 형태를 띄고 있다. JAGS 모델을 설정할 때, β의 사전분포로 정규분포를 설정하였다. 위에서 β_1의 trace plot 이 다른 coefficient 에 비해 잘 수렴되지 않은 이유로 해석할 수 있다.


Density
beta[1]
beta[2]
beta[3]
beta[4]
beta[5]
beta[6]
beta[7]
beta[8]
beta[9]
beta[10]
beta[11]
beta[12]
beta[13]
beta[14]
beta[15]
beta[16]
beta[17]
beta[18]



IV. 결론

본 보고서에서는 베이지안 변수선택을 활용하여 데이터 사이언티스트 연봉 예측을 위한 모델을 구축하였다. MCMC 를 이용하여 GVS 를 수행하여 Avg.Salary.K 에 유의한 영향을 미치는 변수들을 살펴보았다. 최종모형으로는 Rating, python, aws, sql, sas, pytorch, tensor, google_an, seniority_by_title 변수가 선택되었다. Stepwise 변수 선택 결과와 비교했을 때, 더 적은 변수를 선택하였다. Trace plot 을 보면 대체적으로 체인이 잘 수렴되었다. 또한 모든 coefficient 에 대해 Gelman 상수가 1.1 보다 작아 수렴이 잘 되었다고 진단할 수 있다. 다만, 사후분포에서 β_1의 분포가 사전분포로 지정한 정규분포와 차이를 보였기 대문에, 해석 상의 주의가 필요할 것을 보인다. 향후 연구에서 β_1에 대한 사전분포와 모델을 보완할 것이다.

[code]

```
data0<-read.csv("/Users/heesung/Documents/23-1/Bayesian/final project/data_salary.csv")
library(tidyverse)
library(corrplot)
attach(data)
str(data0)
#hourly==1 24 개 제외
data<-data0%>%select(Avg.Salary.K., Rating, Size, Type.of.ownership, Industry, Sector, Revenue,
                     Hourly, Job.Location, Age, Python, spark, aws, excel, sql, sas, keras,
                     pytorch, scikit, tensor, hadoop, tableau, bi, flink, mongo, google_an,
                     seniority_by_title, Degree)%>%filter(Hourly==0)

#####Response: avg.salary#####
sum(is.na(data$Avg.Salary.K.))
quantile(data$Avg.Salary.K.)
mean(data$Avg.Salary.K.)
hist(data$Avg.Salary.K.)
ggplot(data, aes(x=Avg.Salary.K.))+
  geom_histogram(aes(y=..density..), binwidth=5,
                 color="black", fill="white")+
  geom_density(alpha=0.2, fill="red")+
  labs(title="Distribution of Average Salary(K) :Response")

#####Rating#####
hist(data$Rating)
sum(data$Rating==-1) #na 11 개
hist(data$Rating[data$Rating!=-1])
mean(data$Rating[data$Rating!=-1])
med_rating<-median(data$Rating[data$Rating!=-1])
#na median=3.7 로 대체
data$Rating[data$Rating==-1]<-med_rating
#hist(data$Rating, main="Histogram of Company Rating")
ggplot(data, aes(x=Rating))+
  geom_bar()+
  labs(title="Histogram of Company Rating")
plot(data$Rating,    data$Avg.Salary.K.)

#####Size#####
unique(Size)
table(Size)
#size categorical var
data<-data%>%mutate(Size_class=case_when(Size=="1 - 50 "~1,
                                         Size=="51 - 200 "~2,
                                         Size=="201 - 500 "~3,
                                         Size=="501 - 1000 "~4,
                                         Size=="1001 - 5000 "~5,
                                         Size=="5001 - 10000 "~6,
                                         Size=="10000+ "~7))
ggplot(data, aes(x=Size_class, fill=Size_class))+
  geom_bar()+
  labs(title="Histogram of Company Size")+
  scale_fill_discrete(labels=c("1-50", "51-200", "201-500", "501-1000", "1001-5000", "5001-10000", "10000+"))
plot(data$Size_class, data$Avg.Salary.K.)

#size categorical(s/m/l)
data<-data%>%mutate(size.type=case_when(Size_class<=3~"S",
                                        Size_class==4~"M",
                                        Size_class==5~"M",
                                        Size_class>=6~"L"))
data$size.type<-factor(data$size.type, levels=c("S", "M", "L"))
table(data$size.type)
```

```
sum(is.na(data$size.type)) #na 10 개
data$size.type[is.na(data$size.type)]<-"M"
sum(is.na(data$size.type))
ggplot(data, aes(x=size.type, fill=size.type))+
  geom_bar()+
  labs(title="Histogram of Company Size")+
  scale_fill_discrete(labels=c("S 1-500", "M 501-5000", "L 5001+"))
plot(data$size.type, data$Avg.Salary.K.)

#size numerical
data<-data%>%mutate(Size_num=case_when(Size=="1 - 50 "~25.5,
                                       Size=="51 - 200 "~125.5,
                                       Size=="201 - 500 "~350.5,
                                       Size=="501 - 1000 "~750.5,
                                       Size=="1001 - 5000 "~3000.5,
                                       Size=="5001 - 10000 "~7500.5,
                                       Size=="10000+ "~10000))
table(data$Size_num)
mean(na.omit(data$Size_num))
med<-median(na.omit(data$Size_num))
data$Size_num[is.na(data$Size_num)]<-med
sum(is.na(data$Size_num))
plot(data$Size_num, data$Avg.Salary.K.)

#####Type of Ownership#####
unique(data$Type.of.ownership)
table(data$Type.of.ownership)
data$Type.of.ownership<-as.factor(data$Type.of.ownership)
ggplot(data, aes(x=Type.of.ownership, fill=Type.of.ownership))+
  geom_bar()+
  labs(title="Histogram of Ownership Type", x="")+
  scale_x_discrete(labels=NULL, breaks=NULL)
data<-data%>%mutate(ownership.class=
                      ifelse(Type.of.ownership=="Company - Private"|Type.of.ownership=="Subsidiary or Business Segment","Private", "Public"))
data$ownership.class<-as.factor(data$ownership.class)
table(data$ownership.class)
ggplot(data, aes(x=ownership.class, fill=ownership.class))+
  geom_bar()+
  labs(title="Histogram of Ownership Type")
plot(data$ownership.class, data$Avg.Salary.K.)

#####Industry/Sector#####
unique(data$Sector)
sum(data$Sector==-1)
data[data$Sector==-1,]
#na 채우기
data[data$Sector==-1,]$Sector<-c("Health Care", "Information Technology", "Biotech & Pharmaceuticals", "Biotech & Pharmaceuticals",
                                 "Information Technology", "Biotech & Pharmaceuticals", "Biotech & Pharmaceuticals", "Biotech & Pharmaceuticals",
                                 "Biotech & Pharmaceuticals", "Biotech & Pharmaceuticals")
sum(data$Sector==-1)
data$Sector<-as.factor(data$Sector)
data1<-data%>%group_by(Sector)%>%summarise(n=n())%>%mutate(p=n/sum(n)*100)
ggplot(data1, aes(x="", y=p, fill=Sector))+
  geom_bar(stat='identity')+
  coord_polar('y', start=0)+
  theme_void()+
  geom_text(aes(label=paste0(round(p,0), '%')),
            position=position_stack(vjust=0.5))
```

```
plot(data$Sector, data$Avg.Salary.K.)
#####Revenue#####
unique(data$Revenue)
sum(data$Revenue=="Unknown / Non-Applicable" )
table(data$Revenue)
#numerical var
data<-data%>%mutate(revenue.num=case_when(Revenue=="Less than $1 million (USD)"~1,
                                          Revenue=="$1 to $5 million (USD)"~3,
                                          Revenue=="$5 to $10 million (USD)"~7.5,
                                          Revenue=="$10 to $25 million (USD)"~17.5,
                                          Revenue=="$25 to $50 million (USD)"~37.5,
                                          Revenue=="$50 to $100 million (USD)"~75,
                                          Revenue=="$100 to $500 million (USD)"~300,
                                          Revenue=="$500 million to $1 billion (USD)"~750,
                                          Revenue=="$1 to $2 billion (USD)"~1500,
                                          Revenue=="$2 to $5 billion (USD)"~3500,
                                          Revenue=="$5 to $10 billion (USD)"~7500,
                                          Revenue=="$10+ billion (USD)"~10000))
hist(na.omit(data$revenue.num), breaks=80)
mean(na.omit(data$revenue.num))
median(na.omit(data$revenue.num))

table(data$revenue.num)#대표성  없음
sum(is.na(data$revenue.num))
plot(data$revenue.num, data$Avg.Salary.K.)

#####job location#####
sum(is.na(data$Job.Location))

#####Age#####
hist(data$Age)
sum(is.na(data$Age))
mean(data$Age)
median(data$Age)
ggplot(data, aes(x=Age))+
  geom_histogram(aes(y=..density..), binwidth=5,
                 color="black", fill="white")+
  geom_density(alpha=0.2, fill="red")+
  labs(title="Distribution of Age")
plot(data$Age, data$Avg.Salary.K.)

#####skills#####
skill.p<-colSums(data[,11:26])/nrow(data)
pdata<-data.frame(skill=c("Python", "spark", "aws", "excel", "sql", "sas", "keras", "pytorch", "scikit",
                          "tensor", "hadoop", "tableau", "bi", "flink", "mongo", "google_an"),
                  weight=skill.p)
pdata
ggplot(pdata, aes(y=skill, x=weight, fill=weight))+
  geom_col()+
  scale_y_discrete(limits=rev(c("Python", "spark", "aws", "excel", "sql", "sas", "keras", "pytorch", "scikit",
                                "tensor", "hadoop", "tableau", "bi", "flink", "mongo", "google_an")))+
  scale_x_continuous(labels=scales::percent)+
  labs(title="Proportion of Skills Required at Company")

#correlation matrix
df<-data%>%select(Python, spark, aws, excel, sql, sas, keras, pytorch,
                  scikit, tensor, hadoop, tableau, bi, flink, mongo, google_an)
corrplot(cor(df))

#####seniority by title#####
table(data$seniority_by_title)
data$seniority_by_title[data$seniority_by_title=="na"]<-"jr"
data$seniority_by_title<-as.factor(data$seniority_by_title)
```

```
ggplot(data, aes(x=seniority_by_title, fill=seniority_by_title))+
  geom_bar()+
  labs(title="Histogram of Senior/Junior")
plot(data$seniority_by_title, data$Avg.Salary.K.)

#####Degree#####
table(data$Degree)
data$Degree[data$Degree=="na"]<-"B"
data$Degree<-as.factor(data$Degree)
ggplot(data, aes(x=Degree, fill=Degree))+
  geom_bar()+
  labs(title="Histogram of Degree")+
  scale_fill_discrete(labels=c("Bachelor's degree", "Master", "Ph.D"))
plot(data$Degree, data$Avg.Salary.K.)

library(MASS)
library(tidyverse)
library(rjags)
library(data.table)
library(coda)
data2<-data%>%select(Avg.Salary.K., Rating, Age, Size_num, Python, spark,
                     aws, excel, sql, sas, keras, pytorch,scikit, tensor, hadoop,
                     tableau, bi, flink, mongo, google_an,seniority_by_title,
                     ownership.class)

dummy.senior<-model.matrix(~seniority_by_title -1, data=data2)
dummy.senior<-dummy.senior[,1]
dummy.owner<-model.matrix(~ownership.class-1, data=data2)
dummy.owner<-dummy.owner[,1]
data3<-data2[,-c(21,22)]
data3<-cbind(data3, dummy.senior, dummy.owner)
str(data3)

lm.step<-lm(Avg.Salary.K.~., data=data2)
a<-stepAIC(lm.step, direction="both")
summary(a) #14 개 선택

################################################
#Beysian variable selection
n<-nrow(data3)
x<-data3[,-1]
X<-cbind(rep(1,n), x)
k<-ncol(X)

lm.out<-lm(Avg.Salary.K.~., data=data3)
summary(lm.out)

mu.beta<-lm.out$coefficients[1:k]
var.beta<-diag(vcov(lm.out))[1:k]

##pseudo prior:gamma==0 인 경우(변수선택 X) 선택됨
pseudo.beta.mean<-mu.beta
pseudo.beta.var<-var.beta

##prior: gamma==1 인 경우(변수선택 O) 선택됨
prior.beta.var<-var.beta*100
prior.beta.mean<-rep(0, k)

modelString ="model {
#response
for(i in 1:n){
  y[i] ~ dnorm(mu[i], invsigsq)
```

```
   mu[i] <- inprod(X[i,1:k], gbeta[1:k])
}

#gibbs prior
for(j in 1:k){
   gbeta[j]<- gamma[j]*beta[j] }
for(j in 1:k){
   gamma[j] ~ dbern(0.5) }

#lm coef
for(j in 1:k){
   beta[j]~ dnorm( m.b[j], tau.b[j])
   m.b[j] <- gamma[j]*prior.beta.mean[j]+(1- gamma[j])* pseudo.beta.mean[j]
   tau.b[j]<- gamma[j]/prior.beta.var[j]+(1-gamma[j])/pseudo.beta.var[j]
}

#variance
invsigsq ~ dgamma(0.01, 0.01)
}
"

dataList=list(n=n,k=k, y=data3$Avg.Salary.K., X=X, pseudo.beta.mean=pseudo.beta.mean,
                        pseudo.beta.var=pseudo.beta.var, prior.beta.var=prior.beta.var,
                        prior.beta.mean= prior.beta.mean )
gammaInit=rep(1,k)
initsList=list(beta=mu.beta, gamma=gammaInit)
nChains=3
jagsModel=jags.model( textConnection(modelString), data=dataList, inits=initsList,
                                  n.chains=nChains, n.adapt=3000)
update(jagsModel, n.iter=10000)
codaSamples=coda.samples(jagsModel, variable.names=c("gamma", "beta"),
                                     n.chains=nChains,n.iter=10000)
para.samples=as.matrix(codaSamples)
head(para.samples)
beta.samples= para.samples[, 1:k]
gamma.samples=para.samples[, (k+1):(k+k)]


m=gamma.samples
mm=as.data.table(m)[, .N, by = eval(paste0("gamma[", seq_len(ncol(m)), "]"))]
colnames(mm)=c("g0","g1","g2","g3","g4","g5","g6","g7","g8","g9","g10",
                         "g11","g12","g13","g14","g15","g16","g17","g18","g19",
                         "g20","g21","N")
mm.order=order(mm$N, decreasing=T)
mm$N=round( mm$N/(nIter*nChains),4)
gamma.hat=as.numeric(mm[which.max(mm$N)])
gamma.hat=gamma.hat[1:k]
gamma.hat
mm[mm.order[1:10]]

gamma.samples.collapsed<-apply(gamma.samples, 1,
                                              function(x) paste(x, collapse=" "))
gamma.hat.collapsed<-paste(gamma.hat, collapse=" ")
id.selected=which(gamma.samples.collapsed==gamma.hat.collapsed)
length(id.selected)

beta.samples.selected=beta.samples[id.selected,]
colnames(beta.samples.selected)=c("b0","b1","b2","b3","b4","b5","b6","b7",
                                                "b8","b9","b10","b11","b12","b13","b14",
                                                "b15","b16","b17","b18","b19","b20","b21")
beta.samples.selected2=beta.samples.selected[,gamma.hat==1]
head(beta.samples.selected2)
beta.selected.hat=apply(beta.samples.selected2,2,
```

```
                                    function(x)quantile(x, c(0.025,0.5, 0.975)))
t(beta.selected.hat)

##convergence diagnosis
#trace plot
para.names<-variable.names(codaSamples[[1]])
par(mfrow=c(3,3))
for(i in 1:k){
   traceplot( codaSamples[,i] , main=para.names[i] , ylab=para.names[i] )
}

#acf
par(mfrow=c(3,3))
for(i in 1:k){
   acf(codaSamples[,i][[1]],plot=T, main=para.names[i])
}

#gelman
ESS<-effectiveSize(codaSamples); ESS
gelman<-gelman.diag(codaSamples[,-c(k+1, k+5, k+21)]); gelman
gelman.plot(codaSamples[,-c(k+1, k+5, k+21)])


#density plot
MCMCSamples=as.matrix(codaSamples)
HPD= round(apply(MCMCSamples, 2, quantile, probs = c(0.025, 0.975)),4)
par(mfrow=c(3,3))
for(i in 1:k) {
   plot(density(MCMCSamples[,i]), main="",xlab=para.names[i])
   abline( v=HPD[,i],col=2)
}
```