베이지안통계특론 I Final Project

베이지안 회귀분석

통계학과 202STG18 이재빈

Table of Contents

I. MOTIVATION 3

II. DATA 3

1. DATA DESCRIPTION 3
2. DATA PREPROCESSING 4

III. REVIEW 5

1. 선형회귀분석을 위한 데이터 전처리 5
2. 선형회귀모형 적합 7
3. 변수선택법 7

IV. METHOD 8

1. GVS (GIBBS VARIABLE SELECTION) 8
2. BAYESIAN LINEAR REGRESSION 8

V. ANALYSIS 9

1. GVS (GIBBS VARIABLE SELECTION) 9
2. BAYESIAN LINEAR REGRESSION 10

VI. SUMMARY & CONCLUSION 12

VII. REFERENCE 12

VIII. [APPENDIX] CODE 13

1. [PYTHON] DATA PREPROCESSING 13
2. [R] BAYESIAN MODELING 14

I. Motivation

전통적인 선형회귀분석에서 가장 관심있는 문제 중 하나는 변수선택이다. 많은 설명변수 중 유의미한 설명변수를 찾아서, 모델의 성능을 높이고, 동시에 모델의 설명력을 해석하고자 한다.
선형회귀분석에서의 변수선택법이 아닌, MCMC 방법을 이용해 베이지안 변수선택을 하는 방법을 적용해 보고, 베이지안 방법으로 회귀계수를 추정해 봄으로써 기존의 전통 회귀분석 방법론과 비교해 보고, 사전 정보를 활용하는 모델링을 진행해 보고자 한다.

II. Data

Kaggle의 인도 중고차 거래 데이터를 사용하였다.
중고차 상태를 나타내는 여러 변수들을 통해, 중고차 거래 가격 형성에 영향을 끼치는 요소를 파악하고자 하였다.

1. Data Description

데이터에 대한 세부 사항은 다음과 같다.

< y >

- **Price**: The price of the used car in INR Lakhs.

< X >

- Name: The brand and model of the car
- Location: The location in which the car is being sold or is available for purchase.
- Year: The year or edition of the model.
- Kilometers_Driven: The total kilometres driven in the car by the previous owner(s) in KM.
- Fuel_Type: The type of fuel used by the car. (Petrol, Diesel, Electric, CNG, LPG)
- Transmission: The type of transmission used by the car. (Automatic / Manual)
- Owner_Type: Whether the ownership is Firsthand, Second hand or other.
- Mileage: The standard mileage offered by the car company in kmpl or km/kg
- Engine: The displacement volume of the engine in CC.
- Power: The maximum power of the engine in bhp.
- Seats: The number of seats in the car.
- New_Price: The price of a new car of the same model.

2. Data Preprocessing

총 6019 개의 데이터로 이루어져 있으며, 14개의 변수로 구성되어 있다.
데이터를 살펴보면 다음과 같다.

	Unnamed: 0	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mileage	Engine	Power	Seats	New_Price	Price
0	0	Maruti Wagon R LXI CNG	Mumbai	2010	72000	CNG	Manual	First	26.6 km/kg	998 CC	58.16 bhp	5.000000	NaN	1.750000
1	1	Hyundai Creta 1.6 CRDi SX Option	Pune	2015	41000	Diesel	Manual	First	19.67 kmpl	1582 CC	126.2 bhp	5.000000	NaN	12.500000
2	2	Honda Jazz V	Chennai	2011	46000	Petrol	Manual	First	18.2 kmpl	1199 CC	88.7 bhp	5.000000	8.61 Lakh	4.500000
3	3	Maruti Ertiga VDI	Chennai	2012	87000	Diesel	Manual	First	20.77 kmpl	1248 CC	88.76 bhp	7.000000	NaN	6.000000
4	4	Audi A4 New 2.0 TDI Multitronic	Coimbatore	2013	40670	Diesel	Automatic	Second	15.2 kmpl	1968 CC	140.8 bhp	5.000000	NaN	17.740000

- Index 를 나타내는 Unnamed: 0 칼럼을 제거하였다.
- New_Price 변수는 5000개 이상이 결측치로 이루어져 있으므로 제거하였다.
- 차 종류를 나타내는 Name 변수는 1876개, 위치를 나타내는 Location 변수는 11개로 이루어져 있어서 더미변수로 만들기에 적합하지 않다고 판단하여 제거하였다.
- Fuel_Type, Transmission, Owner_Type 은 각각 4개, 5개, 2개 값으로 이루어져 있어 더미화 하였다.
- Mileage, Engine, Power 변수는 뒤의 단위를 제거하여 수치형으로 변환하였다.
- Mileage 에서 결측값을 가지고 있는 변수는 Engine, Power, Seats 등의 값이 존재하므로 0값을 누락했다고 판단하여 0으로 결측치를 채우는 과정을 거쳤다.

	Unnamed: 0	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mileage	Engine	Power	Seats	New_Price	Price
4446	4446	Mahindra E Verito D4	Chennai	2016	50000	Electric	Automatic	First	NaN	72 CC	41 bhp	5.000000	13.58 Lakh	13.000000
4904	4904	Toyota Prius 2009-2016 Z4	Mumbai	2011	44000	Electric	Automatic	First	NaN	1798 CC	73 bhp	5.000000	NaN	12.750000

- Engine, Power 에 결측치가 있는 36 개 데이터는, 대부분의 칼럼에서 결측값을 가지고 있으므로 제거하였다.

	Unnamed: 0	Name	Location	Year	Kilometers_Driven	Fuel_Type	Transmission	Owner_Type	Mileage	Engine	Power	Seats	New_Price	Price
194	194	Honda City 1.5 GXI	Ahmedabad	2007	60006	Petrol	Manual	First	0.0 kmpl	NaN	NaN	nan	NaN	2.950000
208	208	Maruti Swift 1.3 VXi	Kolkata	2010	42001	Petrol	Manual	First	16.1 kmpl	NaN	NaN	nan	NaN	2.110000
733	733	Maruti Swift 1.3 VXi	Chennai	2006	97800	Petrol	Manual	Third	16.1 kmpl	NaN	NaN	nan	NaN	1.750000
749	749	Land Rover Range Rover 3.0 D	Mumbai	2008	55001	Diesel	Automatic	Second	0.0 kmpl	NaN	NaN	nan	NaN	26.500000
1294	1294	Honda City 1.3 DX	Delhi	2009	55005	Petrol	Manual	First	12.8 kmpl	NaN	NaN	nan	NaN	3.200000

이러한 과정을 거쳐 최종적으로 분석에 이용한 데이터의 shape는 (5983, 15) 이다.

Year	Kilometers_Driven	Mileage	Engine	Power	Seats	Price	Fuel_Diesel	Fuel_Electric	Fuel_LPG	Fuel_Petrol	Transmission_Manual	Owner_Fourth & Above	Owner_Second	Owner_Third
2010	72000	26.600000	998.000000	58.160000	5.000000	1.750000	0	0	0	0	1	0	0	0
2015	41000	19.670000	1582.000000	126.200000	5.000000	12.500000	1	0	0	0	1	0	0	0
2011	46000	18.200000	1199.000000	88.700000	5.000000	4.500000	0	0	0	1	1	0	0	0
2012	87000	20.770000	1248.000000	88.760000	7.000000	6.000000	1	0	0	0	1	0	0	0
2013	40670	15.200000	1968.000000	140.800000	5.000000	17.740000	1	0	0	0	0	0	1	0

III. Review

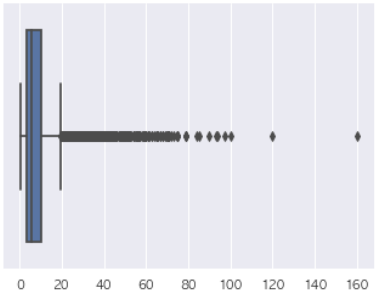
1. 선형회귀분석을 위한 데이터 전처리

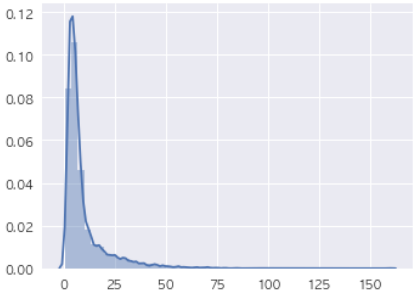
선형회귀모형을 적합시키기 위해서는, 데이터가 등분산성, 정규성, 선형성, 독립성 네 가지 가정을 만족해야 한다.

설명변수들 사이의 관계를 파악하기 위해 그래프를 그려보았다.

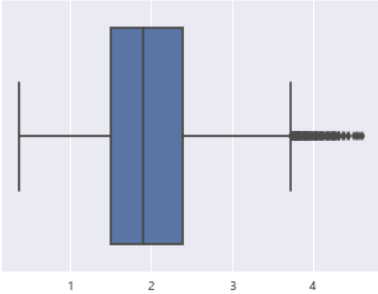
- Kilometers_Driven 변수에 이상점이 하나 존재하여, 이를 제거하였다.
- Engine 과 Power 변수는 선형 관계를 보이고 있으므로, 두 개의 변수를 함께 선형회귀분석에 이용하면 다중공선성 문제가 발생할 것 같아 Engine 변수를 제외하였다.

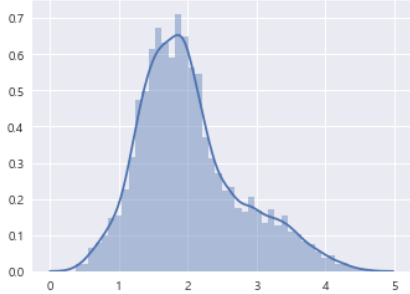
또한 이에 더하여 반응변수의 분포를 살펴 보았다.





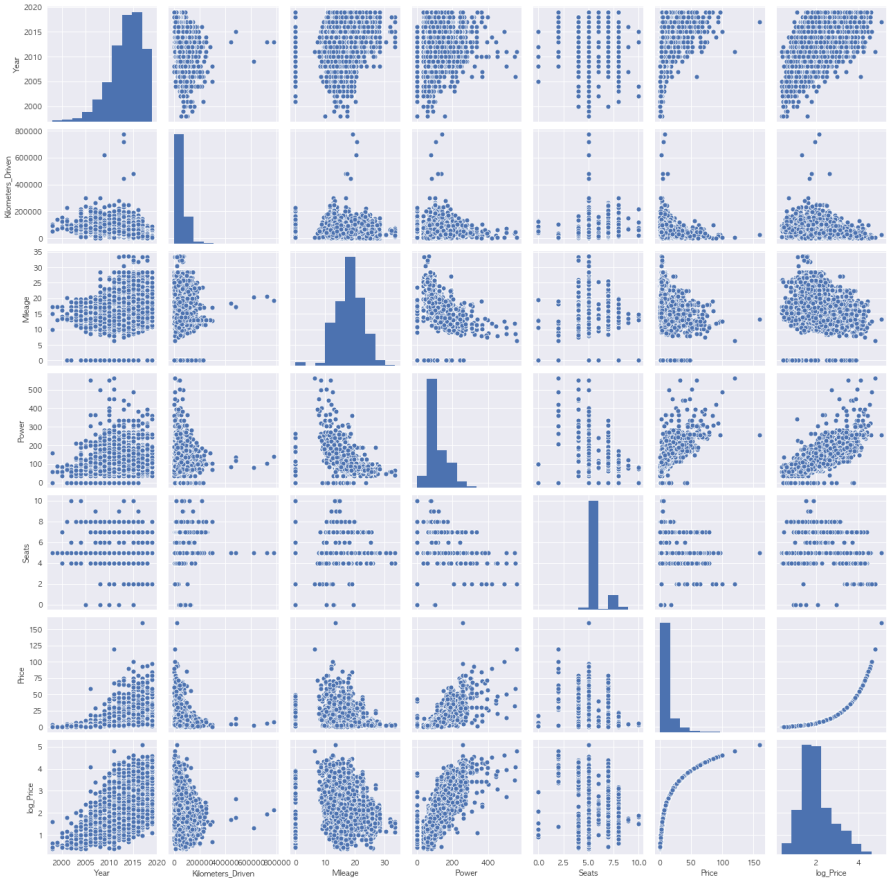
왼쪽으로 지나치게 치우쳐져 있으므로, 반응변수를 log 변환하였다.





정규분포에 가까운 형태를 보이는 것을 확인할 수 있다.

따라서 이를 기반으로 변환한 최종적인 데이터 산점도를 그려보면 다음과 같다.



마지막으로, 반응변수 값이 다른 변수들에 비해 지나치게 튀는 경향을 보이면 회귀분석 시 역행렬을 만들 때 문제가 생기므로, min max scaling 을 진행하여 0과 1 사이의 값으로 변환하였다.

$$x_i = \frac{x_i - x_{min}}{x_{max} - x_{min}}$$

위의 과정을 거쳐, 최종적으로 (5982, 14) shape 의 데이터를 분석에 이용하였다.

2. 선형회귀모형 적합

일반화 선형회귀모형을 적합한 결과는 다음과 같다.

```
Call:
lm(formula = Price ~ ., data = data_norm)

Residuals:
    Min      1Q  Median      3Q     Max 
-3.1696 -0.1683  0.0038  0.1666  2.6759 

Coefficients:
                       Estimate Std. Error t value Pr(>|t|)    
(Intercept)             0.67944    0.06121  11.100  < 2e-16 ***
Year                    2.01769    0.03044  66.294  < 2e-16 ***
Kilometers_Driven      -0.69958    0.08898  -7.862 4.45e-15 ***
Mileage                -0.99847    0.03830 -26.072  < 2e-16 ***
Power                   3.68422    0.05372  68.582  < 2e-16 ***
Seats                  -0.10711    0.05240  -2.044   0.0410 *  
Fuel_Diesel             0.17310    0.03812   4.542 5.69e-06 ***
Fuel_Electric           0.18220    0.20307   0.897   0.3696    
Fuel_LPG               -0.07682    0.09612  -0.799   0.4242    
Fuel_Petrol            -0.19595    0.03845  -5.096 3.58e-07 ***
Transmission_Manual    -0.30059    0.01079 -27.862  < 2e-16 ***
Owner_Fourth...Above    0.07488    0.09942   0.753   0.4513    
Owner_Second           -0.04528    0.01044  -4.336 1.47e-05 ***
Owner_Third            -0.06003    0.02781  -2.158   0.0309 *  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2792 on 5968 degrees of freedom
Multiple R-squared:  0.8605,	Adjusted R-squared:  0.8602 
F-statistic:  2833 on 13 and 5968 DF,  p-value: < 2.2e-16
```

p-value 값을 통해 Year, Kilometers_Driven, Mileage, Power, Seats, Fuel_Diesel, Fuel_Petrol, Transmission_Manual, Owner_Second, Onwer_Third 10개 변수들이 유의미한 설명변수라고 판단되었다.

R-squared 값이 0.86 으로 모델의 설명력이 좋은 것을 확인할 수 있다.

3. 변수선택법

$$\theta = g(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k)$$

일반화 선형회귀모형에서는 p-value 와 AIC/BIC 를 기준으로 가장 설명력이 있는 모델을 선택하고, 이 때의 모델에서 β_j가 0인지 아닌지를 파악한다.

p-value 가 작은 순서대로 변수를 축차적으로 추가하는 Forward, p-value 가 큰 순서대로 변수를 축차적으로 제거하는 Backward 방법이 대표적이며, 이 두 방법을 결합하여 변수의 추가와 제거가 모두 가능한 Stepwise 방법론을 적용해 보았다.

```
Step:  AIC=-15252.74
Price ~ Year + Kilometers_Driven + Mileage + Power + Seats + 
    Fuel_Diesel + Fuel_Petrol + Transmission_Manual + Owner_Second + 
    Owner_Third

                      Df Sum of Sq    RSS    AIC
<none>                               465.48 -15253
+ Fuel_Electric        1      0.07 465.40 -15252
+ Fuel_LPG             1      0.06 465.42 -15252
+ Owner_Fourth...Above 1      0.04 465.43 -15251
- Seats                1      0.33 465.81 -15250
- Owner_Third          1      0.37 465.84 -15250
- Owner_Second         1      1.49 466.97 -15236
- Fuel_Diesel          1      2.08 467.56 -15228
- Fuel_Petrol          1      2.34 467.81 -15225
- Kilometers_Driven    1      4.78 470.26 -15194
- Mileage              1     54.54 520.01 -14592
- Transmission_Manual  1     60.73 526.21 -14521
- Year                 1    344.17 809.65 -11943
- Power                1    368.91 834.38 -11763

Call:
lm(formula = Price ~ Year + Kilometers_Driven + Mileage + Power + 
    Seats + Fuel_Diesel + Fuel_Petrol + Transmission_Manual + 
    Owner_Second + Owner_Third, data = data_norm)

Coefficients:
        (Intercept)                Year   Kilometers_Driven             Mileage 
            0.67675             2.01824            -0.69617            -1.00228 
              Power               Seats          Fuel_Diesel          Fuel_Petrol 
            3.68010            -0.10768             0.17888            -0.19044 
Transmission_Manual         Owner_Second          Owner_Third 
           -0.30098            -0.04566            -0.06044 
```

AIC 값이 가장 작은 모델을 선택하였다. Year, Kilometers_Driven, Mileage, Power, Seats, Fuel_Diesel, Fuel_Petrol, Transmission_Manual, Owner_Second, Onwer_Third 10개 설명변수들을 선택하였다.

IV. Method

1. GVS (Gibbs Variable Selection)

Kuo-Mallick 변수선택 방법에서는, 0 또는 1의 이항값을 갖는 지시변수 γ를 회귀모형에 직접 도입하여 변수 선택을 진행한다.

$$\theta = g(\gamma_0\beta_0 + \gamma_1\beta_1 x_1 + \gamma_2\beta_2 x_2 + \cdots + \gamma_k\beta_k x_k)$$

그렇지만 이 방법론은 $\gamma_j = 0$ 에 해당하는 β_j 가 모형에서 사라져 버리기 때문에, MCMC 과정에서 사후분포를 이용하는 것이 아니라, 자신의 사전분포를 이용하게 된다는 단점이 있다.

따라서 이러한 단점을 개선한 Gibbs Variable Selection 방법론을 적용해, 변수 선택을 진행하였다.

$$\pi(\beta_j|\gamma_j)\begin{cases}\pi_1(\beta_j), & if\ \gamma_j = 1\\ \pi_0(\beta_j), & if\ \gamma_j = 0\end{cases}$$

GVS 또한 지시변수 벡터 γ를 통해 변수를 선택하는 방법이다.
사전분포로 $\pi(\beta_j, \gamma_j) = \pi(\beta_j|\gamma_j)\,\pi(\gamma_j)$ 를 가정함으로써, β_j의 사전분포가 γ_j 에 따라 달라지게끔 한다. MCMC 과정에서 β, γ 의 표본을 추출한 뒤 γ 표본을 취하고, 이에 대한 사후표본들 중에서 가장 상대도수가 높은 γ 조합을 최종 선택한다.

2. Bayesian Linear Regression

선형회귀모형에서의 모수 β, σ^2 에 대해서 사전분포를 가정하고, 결합사후밀도함수를 통해 조건부 사후분포를 계산함으로써 표본을 생성하여, 깁스 표본기법을 적용해 β, σ^2 에 대해 근사적인 사후추론을 하는 방법이다.

보통 사전분포는 선형회귀모형 추정치인 $\beta \sim N(\hat{\beta}, \Sigma_0)$ 로 설정하며, $\sigma^2 \sim IG(a, b)$ 로 설정하여 MCMC 를 통해 사후추론을 진행한다.

V. Analysis

1. GVS (Gibbs Variable Selection)

회귀계수에 대한 pseudo-prior로 정규분포를 가정하고, 평균과 분산을 회귀분석에서 얻은 최우추정치 $\hat{\beta}$ 와 분산추정치의 대각원소만 취한 $\widehat{var}(\hat{\beta})$ 을 사용하였다.
사전분포는 정규분포로 가정하고, 평균은 0, 분산은 $100 * \widehat{var}(\hat{\beta})$ 을 사용하였다.

위에서 설정한 데이터 초기치와 사전모수를 dataList 에 저장하고, γ 의 초기치를 모두 1로 설정한 후 MCMC를 수행하여 β와 γ의 표본을 취하였다.

Chain 의 개수는 3개로 설정하였고, adaptation 은 1000개로 설정해 Tuning 하였다.
burn-in 은 3000개로 설정하고, 10000개 표본까지 Iteration 을 진행하게끔 설정하였다.

Intercept를 포함한 모든 변수의 조합에 대해 계산하였고, 벡터변수 γ의 가능한 조합에 대한 사후확률을 계산해 내림차순으로 정렬한 결과는 다음과 같다.

	gamma[1]	gamma[2]	gamma[3]	gamma[4]	gamma[5]	gamma[6]	gamma[7]	gamma[8]	gamma[9]	gamma[10]	gamma[11]	gamma[12]	gamma[13]	gamma[14]	N
1:	0	1	0	1	1	1	0	0	0	1	1	1	0	0	0.5917
2:	0	1	0	1	1	1	0	1	0	1	1	1	0	0	0.1678
3:	0	1	0	1	1	1	0	0	1	1	1	1	0	0	0.1174
4:	0	1	0	1	1	1	0	0	0	1	1	0	0	0	0.0594
5:	0	1	0	1	1	1	0	1	1	1	1	1	0	0	0.0328
6:	0	1	0	1	1	1	0	1	0	1	1	0	0	0	0.0161
7:	0	1	0	1	1	1	0	0	1	1	1	0	0	0	0.0114
8:	0	1	0	1	1	1	0	1	1	1	1	0	0	0	0.0035

$\hat{\gamma}$ 을 보았을 때, M1 = (x2, x4, x5, x6, x10, x11, x12) 조합이 선정되었음을 알 수 있고,
이 때의 사후확률은 0.5917 로 매우 높은 편이다. 총 30000개의 표본 중 모형 M1 으로부터 산출된 표본은 17750개이다.

Used car data 에 대한 변수선택 결과 비교

x0: intercept, x1: Year, x2: Kilometers_Driven, x3: Milege, x4: Power, x5: Seats, x6: Fuel_Diesel, x7: Fuel_Electric, x8: Fuel_LPG, x9: Fuel_Petrol, x10: Transmission_Manual, x11: Owner_Fourth...Above, x12: Owner_Second, x13: Owner_Third

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13
GVS			●		●	●	●				●	●	●	
stepAIC	●	●	●	●	●	●	●			●	●		●	●
MLE	●	●	●	●	●	●	●			●	●		●	●

베이지안 변수선택법 적용 시 가장 적은 변수를 선택했음을 볼 수 있다.
선택한 변수는 Kilometers_Driven, Power, Seats, Fuel_Diesel, Transmission_Manual, Owner_Fourth...Above, Owner_Second 총 7개이다.

선택한 변수들의 계수 추정치와 CI는 다음과 같다.

	2.5%	50%	97.5%
Year	3.46036422	3.4968045	3.5329416
Mileage	-1.81980581	-1.7755453	-1.7314719
Power	5.94574845	6.0156561	6.0862734
Seats	-0.65675262	-0.6058200	-0.5558568
Fuel_Petrol	-0.28873763	-0.2682877	-0.2472445
Transmission_Manual	-0.45357572	-0.4393895	-0.4252544
Owner_Fourth...Above	0.07074584	0.1986250	0.3262436

2. Bayesian Linear Regression

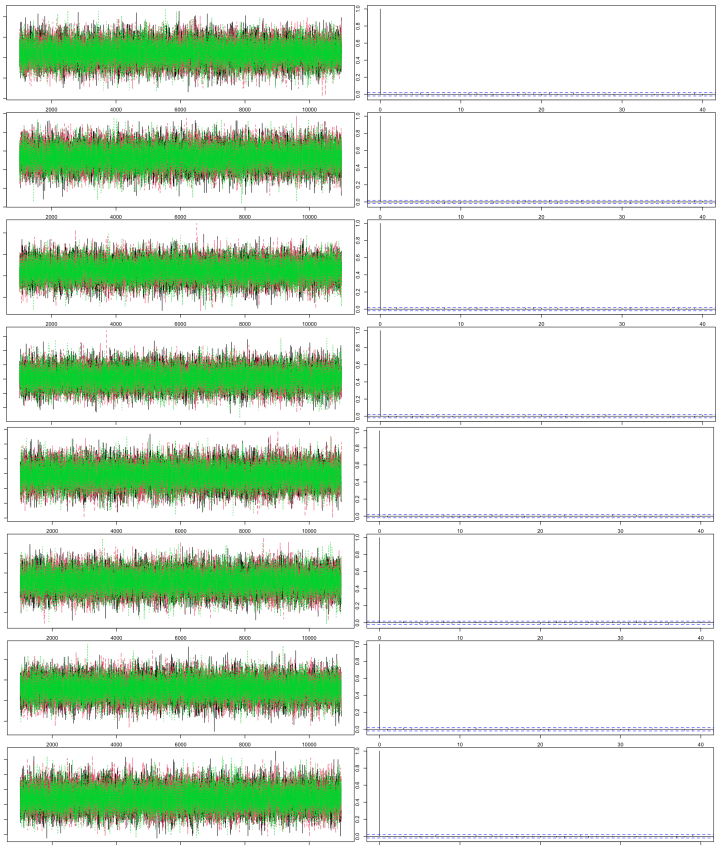
선택된 변수들만을 가지고 베이지안 회귀분석을 진행하여, 좀 더 정밀하게 계수 추정을 하였다. 절편이 없는 선형회귀분석 모형으로 진행하였으며, 위의 결과로부터 나온 7개 변수를 적합하였다.

회귀계수 β와 오차분산 σ^2의 초기치로 최소제곱추정치 (LSE) 인 $\hat{\beta}$, $\hat{\sigma}^2$를 사용하였으며,
β의 사전분포로 $N\left(\hat{\beta}, 100\hat{\sigma}^2\widehat{Var}(\hat{\beta})\right)$, $\widehat{Var}(\hat{\beta}) = Diag((X'X)^{-1})$,
σ^2의 사전분포로 $IG(1,1)$을 사용하였다.

Chain 의 개수는 3개로 설정하였고, adaptation 은 3000개로 설정해 Tuning 하였다.
burn-in 은 1000개로 설정하고, 10000개 표본까지 Iteration 을 진행하게끔 설정하였다.

MCMC를 사용하여 표본의 경로그림과 ACF를 그려본 결과는 다음과 같다.

Kilometers_Driven, Power, Seats, Fuel_Diesel, Transmission_Manual, Owner_Fourth...Above, Owner_Second, σ^2



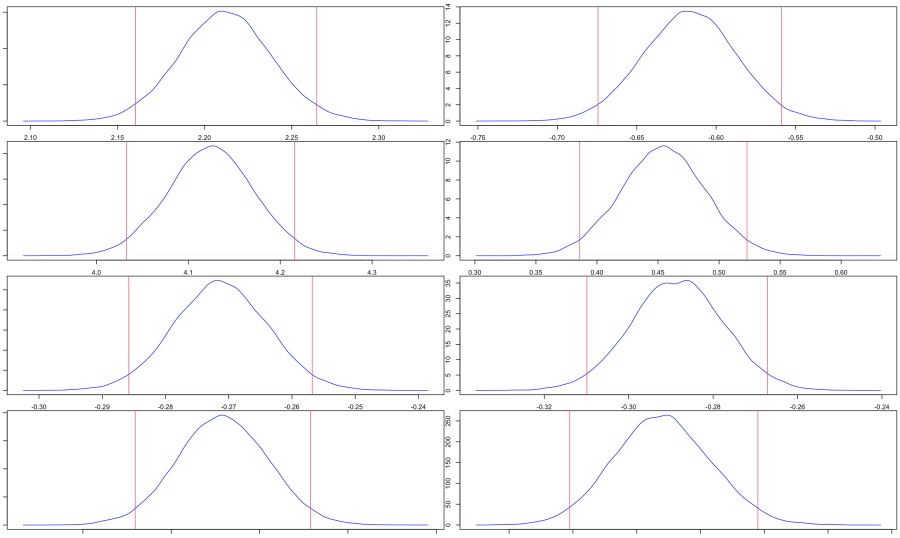
모든 모수에 대해 3개의 MCMC 체인의 경로그림이 안정적이고, 잘 겹쳐져 있음을 확인할 수 있다. 표본들이 자기상관을 보이지 않는 것 또한 확인할 수 있다.
따라서 MCMC가 수렴이 잘 되었다고 할 수 있다.

이에 더하여 Gelman 통계량을 통해 수렴을 진단하였다.

	Point est.	Upper C.I.
beta[1]	0.9999935	1.000114
beta[2]	1.0001291	1.000345
beta[3]	1.0000792	1.000269
beta[4]	1.0001319	1.000542
beta[5]	1.0000010	1.000214
beta[6]	1.0000051	1.000219
beta[7]	0.9999719	1.000095
sigsq	1.0000261	1.000293

모든 Gelman 상수들이 1에 매우 가까운 값을 가지고 있으므로, 수렴하였다고 판단하였다.

사후밀도함수와 95% 사후구간에 대한 그림은 다음과 같다.



모든 모수가 정규분포 형태를 보이고 있으므로, 추정이 잘 되었다고 할 수 있다.

또한 이에 대한 채택확률 계산 결과는 다음과 같다.

beta[1]	beta[2]	beta[3]	beta[4]	beta[5]	beta[6]	beta[7]	sigsq
1	1	1	1	1	1	1	1

모두 1 이므로 JAGS 에서 깁스 표본기법을 사용하였음을 알 수 있다.

최종적인 MCMC 계수 추정치와 CI에 대한 결과이다.

	2.5%	50%	97.5%
Year	2.16019155	2.21174694	2.26432867
Mileage	-0.67439383	-0.61641814	-0.55893255
Power	4.03256815	4.12470856	4.21576552
Seats	0.38589104	0.45424798	0.52293740
Fuel_Petrol	-0.28575699	-0.27131350	-0.25678694
Transmission_Manual	-0.30985985	-0.28844819	-0.26710457
Owner_Fourth...Above	-0.08164962	0.11607621	0.31525351
sigsq	0.07989932	0.08278372	0.08580096

VI. Summary & Conclusion

일반적인 회귀분석과 베이지안으로 변수를 선택한 후 베이지안 회귀분석을 적용하는 방법을 진행해 보았다.
일반적인 회귀분석 방법론으로 추정했을 때 보다, 베이지안 방법론으로 추정했을 때 더 간결한 모델이 선택됨을 확인할 수 있었다.
또한 베이지안 방법을 통해 변수를 선택한 후, 이를 기반으로 베이지안 회귀분석을 진행해 보았을 때 CI 폭이 더 작아지는 것으로 보아 좀 더 정밀한 추정이 가능해 짐을 확인할 수 있었다.

베이지안 모델로 추정한 회귀분석 결과, 중고차 가격에 영향을 주는 변수로는 (+)Year, (-)Mileage, (+)Power, (+)Seats, (-)Fuel_Petrol, (-)Transmission_Manual, (-)Owner_Fourth...Above 가 있다.
최신 중고차일수록, Power가 좋을 수록, 의자 개수가 많을 수록 중고차 가격이 높아지며, 많이 탑승해서 연식이 높을수록, 연료, 수동 여부, 주인이 여러 번 바뀌었을 수록 중고차 가격이 낮아지는 것을 확인할 수 있다.

MCMC 방법을 통해 계수를 추정함으로써 해당 요소들이 중고차 가격에 어느 정도 영향을 끼치는지 또한 파악할 수 있었다.

VII. Reference

- Data: https://www.kaggle.com/avikasliwal/used-cars-price-prediction
- 오만숙. (2019). JAGS를 활용한 베이지안 자료분석. 자유아카데미.

VIII. [Appendix] Code

1. [Python] Data Preprocessing

```
import pandas as pd
import numpy as np

%matplotlib inline
import matplotlib.pyplot as plt
import seaborn as sns
pd.set_option('float_format', '{:f}'.format)
pd.set_option('display.max_rows', 500)
pd.set_option('display.max_columns', 500)

plt.style.use('seaborn')
plt.rc('font', family='AppleGothic')
plt.rc('axes', unicode_minus=False)


# ============================
#      Data Preprocessing
# ============================

# Data Load
data = pd.read_csv("UsedCar.csv")

# Data Remove
data = data.drop(["Unnamed: 0", "New_Price"], axis=1)
data = data.dropna(subset = ["Engine", "Power"], axis=0)

# for Missing Value Imputation
data['Mileage'] = data['Mileage'].fillna("0 kmpl")

# Remove String
data['Mileage'] = data['Mileage'].apply(lambda x: float(x.split()[0]))
data['Engine'] = data['Engine'].apply(lambda x: float(x.split()[0]))

def to_numeric(x) :
    try :
        return float(x.split()[0])
    except :
        return 0

data['Power'] = data['Power'].apply(lambda x: to_numeric(x))
data["Seats"] = data["Seats"].fillna(0)

# Dummy Variable
data = pd.get_dummies(data, columns=['Fuel_Type'], prefix='Fuel', drop_first=True)
data = pd.get_dummies(data, columns=['Transmission'], prefix='Transmission', drop_first=True)
data = pd.get_dummies(data, columns=['Owner_Type'], prefix='Owner', drop_first=True)
```

```python
# ============================
#      Data Visualization
# ============================

# target distribution
figure, (ax1, ax2) = plt.subplots(ncols=2)
figure.set_size_inches(12,4)
sns.boxplot(data[["Price"]], ax=ax1)
sns.distplot(data[["Price"]],ax=ax2)

# log transformation
data["log_Price"] = np.log1p(data["Price"])
figure, (ax1, ax2) = plt.subplots(ncols=2)
figure.set_size_inches(12,4)
sns.boxplot(data[["log_Price"]], ax=ax1)
sns.distplot(data[["log_Price"]],ax=ax2)

# Pairplot
sns.pairplot(data[['Year', 'Kilometers_Driven', 'Mileage', 'Engine', 'Power', 'Seats', 'Price']])
plt.show()

# Remove Outlier
data = data[data["Kilometers_Driven"]<6500000]

# Remove Column
data = data.drop(['Price', 'Engine'], axis=1)

# Data Save
data.to_csv("UsedCar2.csv", index=False)
```

2. [R] Bayesian Modeling

```r
library(MASS)
library(dplyr)
library(tidyverse)
library(rjags)
library(data.table)

# Used Car Price Prediction
data <- read.csv("UsedCar2.csv")

min_max_norm <- function(x) {
  (x - min(x)) / (max(x) - min(x))
}
data_norm <- as.data.frame(lapply(data %>% select(-log_Price), min_max_norm))
data_norm$Price <- data$log_Price


# ====================================
# 1. linear regression
# ====================================
result <- lm(Price ~ . , data=data_norm)
summary(result)


# Stepwise
stepAIC(result, direction = 'both')
```

```r
# ======================================
# 2. GVS (Gibbs Variable Selection)
# pseudo-prior : Normal dist, mean/var = LSE
# ======================================
mu.beta <- result$coefficients
var.beta <- diag(vcov(result))

y <- data_norm$Price ; x <- data_norm %>% select(-Price) ; n <- nrow(data)
X <- cbind(rep(1,n), x) ; K <- ncol(X)

# pseudo prior
pseudo.beta.mean <- mu.beta
pseudo.beta.var <- var.beta

# prior
prior.beta.var <- var.beta * 100
prior.beta.mean <- rep(0, K)


# Model Setting for JAGS
modelString = "
model {
  for(j in 1:K){gbeta[j] <- gamma[j]*beta[j]}
  for(i in 1:n){
    y[i] ~ dnorm(mu[i], invsigsq)
    mu[i] <- inprod(X[i,1:K], gbeta [1:K])
  }

  for(j in 1:K){gamma[j] ~ dbern (0.5)}
  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]
  }

  invsigsq ~ dgamma (0.01, 0.01)
  }
  "
write(modelString , file="model_GVS2.txt")


# MCMC
dataList <- list(n=n, K=K, y=y, 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) # gamma init = 1
initsList=list(beta=mu.beta, gamma=gammaInit)

nChains=3; nIter=10000
jagsModel=jags.model(file="model_GVS2.txt", data=dataList, inits=initsList,
                    n.chains=nChains, n.adapt = 1000)
update(jagsModel, n.iter = 3000)
codaSamples=coda.samples(jagsModel, variable.names=c("gamma", "beta"),
                         n.chains=nChains, n.iter=nIter)
para.samples=as.matrix(codaSamples)

head(para.samples)
beta.samples = para.samples[, 1:K] # 1 ~ 14
gamma.samples = para.samples[, (K+1) : (K+K)] # 15 ~ 28
```

```r
# Variable Selection
m = gamma.samples
mm = as.data.table(m)[, .N, by=eval(paste0("gamma[", seq_len(ncol(m)), "]"))]
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]]


# Model1 Coef Estimation
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) # 17750 = 3 * 10000 * 0.5917

beta.samples.selected = beta.samples[id.selected, ]
colnames(beta.samples.selected)=c("intercept", colnames(x))
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)



# ======================================
# 3. Bayesian Linear Regression
# ======================================
y <- data_norm$Price ; x2 <- data_norm[,colnames(beta.selected.hat)] ; n <- nrow(data_norm)
K2 <- ncol(x2)

a = 1 ; b = 1 ; X2 <- as.matrix(x2)
XtX = t(X2)%*%X2 ; XtX.inv = solve(XtX) ; Xty = t(X2)%*%y
beta.hat <- as.vector(XtX.inv %*% Xty)
sigsq.hat <- sum((y-X2 %*% beta.hat)^2)/(n-K2)
beta0 <- beta.hat
Sig0 <- diag(diag(XtX.inv))*sigsq.hat*100
Sig0.inv <- solve(Sig0)

modelString2 = "
model
{
  for (i in 1:length(y)){
    y[i] ~ dnorm(inprod(X[i,], beta[]), invsigsq)
  }
  beta[1:length(beta0)] ~ dmnorm(beta0[], Sig0.inv[,])

  invsigsq ~ dgamma(a,b)
  sigsq = 1/invsigsq
}
"
writeLines(modelString2, "model_reg.txt")

dataList2 = list(X=X2, y=y, a=a, b=b, beta0=beta0, Sig0.inv=Sig0.inv)
initsList2 = list(beta=beta.hat, invsigsq = 1/sigsq.hat)
nChains=3

jagsModel2 = jags.model(file="model_reg.txt", data=dataList2, inits=initsList2,
                        n.chains = nChains, n.adapt = 3000)
update(jagsModel2, n.iter = 1000)
codaSamples2=coda.samples(jagsModel2, variable.names=c("beta", "sigsq"), n.iter=10000) # 30000
para.names2 = variable.names(codaSamples2[[1]])
```

```r
# Trace Plot & ACF
par("mar")
par(mar = c(1,1,1,1))
for(i in 1:9) {
  coda::traceplot(codaSamples2[,i], main="", ylab=para.names2[i])
  acf(codaSamples2[,i][[1]], plot=T, main=para.names2[i])
  }

MCMCSamples = as.matrix(codaSamples2)
HPD = round(apply(MCMCSamples, 2, quantile, probs=c(.025, .975)), 4)


# Posterior density & 95% HPD
par(mfrow=c(5,2))
for(i in 1:9) {
  plot(density(MCMCSamples[,i]), main="", xlab=para.names2[i], col='blue')
  abline(v=HPD[,i], col=2)
}


# Gelman Statistics
gelman = gelman.diag(codaSamples2) # beta
gelman.1 = as.matrix(gelman$psrf)
if (max(gelman.1 > 1.1)) cat("Warning : Gelman Shrink Factor > 1.1", "\n")
gelman.2 = gelman$psrf
if (max(gelman.2 > 1.1)) cat("Warning : Gelman Multivariate Shrink Factor > 1.1", "\n")
gelman.1


# Acceptance Rate
AcceptRate = 1 - rejectionRate(codaSamples2) ; AcceptRate


# beta hat
mcmc.beta.hat = apply(MCMCSamples, 2, function(x) quantile(x, c(0.025,0.5,0.975)))
colnames(mcmc.beta.hat)=c("intercept", colnames(x2), "sigsq")
t(mcmc.beta.hat)
```