

1번

```
hw1 = read.csv('stock.csv',sep="," ,header=TRUE)
attach(hw1)
head(hw1)
```

(a)

#Dummy Matrix를 만듭니다. 예를 들어, Dec 벡터는 Month가 12월인 경우 1을 부여하고, 그렇지 않은 경우 0을 부여하는 방법으로 더미 행렬을 만드는 것입니다.

```
Dec <- ifelse(Month == '12', 1, 0)
Nov <- ifelse(Month == '11', 1, 0)
Oct <- ifelse(Month == '10', 1, 0)
Sep <- ifelse(Month == '9', 1, 0)
Aug <- ifelse(Month == '8', 1, 0)
July <- ifelse(Month == '7', 1, 0)
June <- ifelse(Month == '6', 1, 0)
May <- ifelse(Month == '5', 1, 0)
April <- ifelse(Month == '4', 1, 0)
March <- ifelse(Month == '3', 1, 0)
Feb <- ifelse(Month == '2', 1, 0)
z <- cbind(Feb, March, April, May, June, July, Aug, Sep, Nov, Dec)
#cbind를 사용하여 더미 matrix z를 만들었습니다.
```

	Feb	March	April	May	June	July	Aug	Sep	Nov	Dec
[1,]	0	0	0	0	0	0	0	0	0	1
[2,]	0	0	0	0	0	0	0	0	1	0
[3,]	0	0	0	0	0	0	0	0	0	0
[4,]	0	0	0	0	0	0	0	1	0	0
[5,]	0	0	0	0	0	0	1	0	0	0
[6,]	0	0	0	0	0	1	0	0	0	0
[7,]	0	0	0	0	1	0	0	0	0	0
[8,]	0	0	0	1	0	0	0	0	0	0
[9,]	0	0	1	0	0	0	0	0	0	0
[10,]	0	1	0	0	0	0	0	0	0	0
[11,]	1	0	0	0	0	0	0	0	0	0
[12,]	0	0	0	0	0	0	0	0	0	0
[13,]	0	0	0	0	0	0	0	0	0	1
[14,]	0	0	0	0	0	0	0	0	1	0
[15,]	0	0	0	0	0	0	0	0	0	0
[16,]	0	0	0	0	0	0	0	1	0	0

[illegible]

(c)

```
beta.hat <- solve(t(X)%*%as.matrix(X))%*%t(X)%*%y
```

# $(X'X)^{-1}X'Y$  공식을 적용했습니다. 그 결과는 아래와 같습니다.

[,1]

intercept	3227.37920
Interest	146.55506
Unemployment	-432.37910
Feb	-12.51427
March	25.04739
April	-38.42843
May	17.80948
June	27.99010
July	92.10905
Aug	82.49010
Sep	51.25218
Nov	62.01427
Dec	122.99446

(d)

```
beta.hat2 <- lm(y~.,data = X)
```

#X에 있는 데이터를 이용하여 y에 대한 회귀분석을 진행했습니다. y~. 은 X에 있는 모든 변수를 사용한다는 것을 의미합니다. 그 결과는 아래와 같습니다.

# (c)에서의 결과와 동일합니다. 다만 lm 코드의 경우 intercept를 자동으로 도출해줘서 (c)에서 부여한 intercept 부분이 중복해서 계산되었습니다. 값 도출에는 문제가 없었습니다.

```
summary(beta.hat2)
```

Call:

```
lm(formula = y ~ ., data = X)
```

Residuals:

	Min	1Q	Median	3Q	Max
	-127.34	-21.25	0.00	21.25	105.55

Coefficients: (1 not defined because of singularities)

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	3227.38	1525.35	2.116	0.058 .
intercept	NA	NA	NA	NA
Interest	146.56	191.77	0.764	0.461
Unemployment	-432.38	199.44	-2.168	0.053 .
Feb	-12.51	71.00	-0.176	0.863
March	25.05	74.04	0.338	0.742
April	-38.43	71.31	-0.539	0.601
May	17.81	69.90	0.255	0.804

June	27.99	75.44	0.371	0.718
July	92.11	79.69	1.156	0.272
Aug	82.49	75.44	1.093	0.298
Sep	51.25	70.49	0.727	0.482
Nov	62.01	71.00	0.873	0.401
Dec	122.99	84.85	1.450	0.175

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 80.51 on 11 degrees of freedom

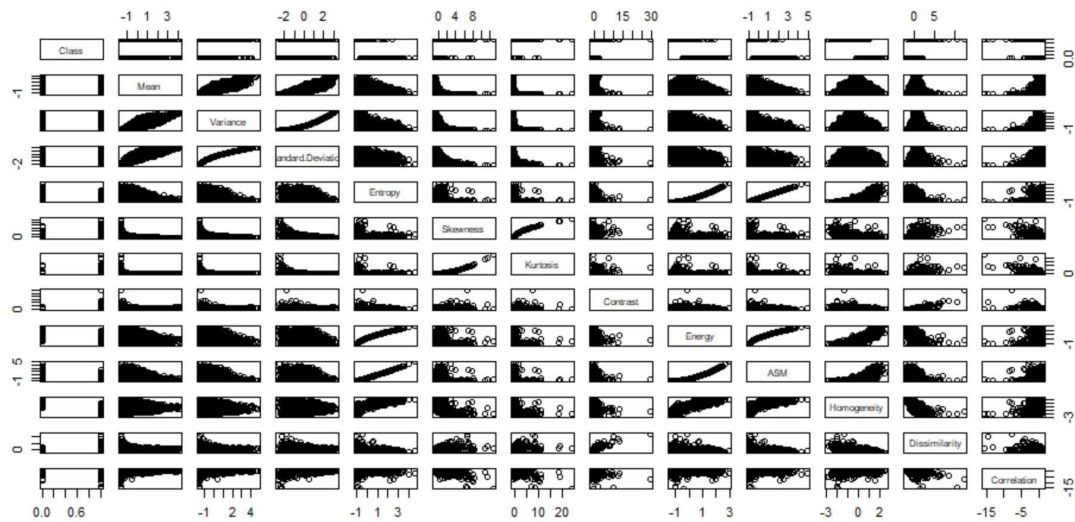
Multiple R-squared: 0.9302, Adjusted R-squared: 0.8541

F-statistic: 12.22 on 12 and 11 DF, p-value: 0.0001113

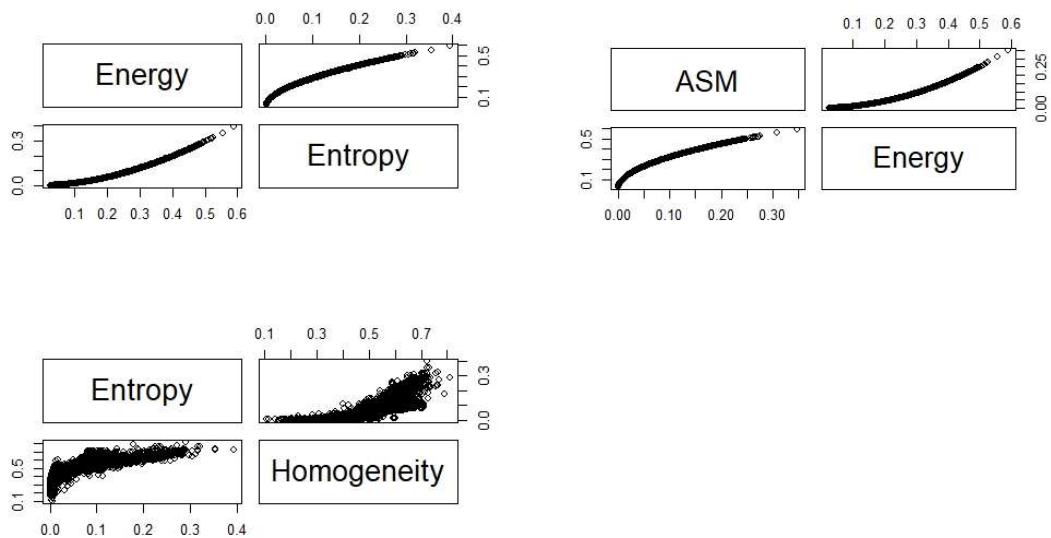
## 2번

(a)

```
hw2 = read.csv('BrainTumor.csv',sep=',',header=TRUE)
attach(hw2)
#scale을 사용해 regularization을 하여 이를 다시 hw2에 넣었습니다.
hw2$Mean <- scale(Mean,center=TRUE,scale=TRUE)
hw2$Variance <- scale(Variance,center=TRUE,scale=TRUE)
hw2$Standard.Deviation <- scale(Standard.Deviation,center=TRUE,scale=TRUE)
hw2$Entropy <- scale(Entropy,center=TRUE,scale=TRUE)
hw2$Skewness <- scale(Skewness,center=TRUE,scale=TRUE)
hw2$Kurtosis <- scale(Kurtosis,center=TRUE,scale=TRUE)
hw2$Contrast <- scale(Contrast,center=TRUE,scale=TRUE)
hw2$Energy <- scale(Energy,center=TRUE,scale=TRUE)
hw2$ASM <- scale(ASM,center=TRUE,scale=TRUE)
hw2$Homogeneity <- scale(Homogeneity,center=TRUE,scale=TRUE)
hw2$Dissimilarity <- scale(Dissimilarity,center=TRUE,scale=TRUE)
hw2$Correlation <- scale(Correlation,center=TRUE,scale=TRUE)
pairs(hw2)
```



# Energy와 Entropy, ASM과 Energy,  
 pairs(~Energy+Entropy) #강한 양의 상관관계  
 pairs(~Entropy+Homogeneity) #양의 상관관계  
 pairs(~ASM+Energy) #강한 양의 상관관계



(b)

```
#train and test data
i = sample(1:nrow(hw2), round(nrow(hw2)*0.7))
train = hw2[i,]
test = hw2[-i,]
fit2 = glm(Class ~ ., family = "binomial", data = train)
summary(fit2)
Call:
```

```
glm(formula = Class ~ ., family = "binomial", data = train)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-1.97944	-0.02235	-0.00196	0.00007	3.15129

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	3.0573	1.0833	2.822	0.004770 **
Mean	1.0443	0.7249	1.441	0.149691
Variance	-4.2462	1.4356	-2.958	0.003098 **
Standard.Deviation	6.4147	1.6560	3.874	0.000107 ***
Entropy	-45.6067	40.1401	-1.136	0.255878
Skewness	13.1185	2.2341	5.872	4.31e-09 ***
Kurtosis	-6.8743	1.4847	-4.630	3.66e-06 ***
Contrast	4.4646	0.9508	4.696	2.66e-06 ***
Energy	-7.1519	8.5419	-0.837	0.402444
ASM	47.3582	32.5196	1.456	0.145311
Homogeneity	-8.6751	1.8967	-4.574	4.79e-06 ***
Dissimilarity	-8.3560	1.5915	-5.250	1.52e-07 ***
Correlation	2.2130	0.5574	3.970	7.19e-05 ***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 3612.82 on 2632 degrees of freedom  
Residual deviance: 185.82 on 2620 degrees of freedom  
AIC: 211.82

Number of Fisher Scoring iterations: 11

유의한 변수는 아래와 같습니다.

[Intercept, Variance, Standard.Deviation, Skewness, Kurtosis, Contrast, Homogeneity, Dissimilarity, Correlation]

유의한 변수에 대한 베타의 의미는 다음과 같습니다.

Intercept: 1단위 증가함에 따라 odds비율이  $\exp(3.0573)$ 만큼 증가합니다.

Variance: 1단위 증가함에 따라 odds비율이  $\exp(-4.2462)$ 만큼 증가합니다.

Standard.Deviation: 1단위 증가함에 따라 odds비율이  $\exp(6.4147)$ 만큼 증가합니다.

Skewness: 1단위 증가함에 따라 odds비율이  $\exp(13.1185)$ 만큼 증가합니다.

Kurtosis: 1단위 증가함에 따라 odds비율이  $\exp(-6.8743)$ 만큼 증가합니다.  
Contrast: 1단위 증가함에 따라 odds비율이  $\exp(4.4646)$ 만큼 증가합니다.  
Homogeneity: 1단위 증가함에 따라 odds비율이  $\exp(-8.6751)$ 만큼 증가합니다.  
Dissimilarity: 1단위 증가함에 따라 odds비율이  $\exp(-8.3560)$ 만큼 증가합니다.  
Correlation: 1단위 증가함에 따라 odds비율이  $\exp(2.2130)$ 만큼 증가합니다.

(c)

```
glm.prob <- predict(fit2, newdata = test, type = "response")
```

#train 데이터로 수행한 fit2 모델에 test data를 적용하여 그 예측값을 구하고자 합니다. 그 값은 glm.prob라는 새 값에 할당하였습니다. 여기서 response는 정확도를 확률로 구하기 위한 인자입니다.

```
glm.prob <- ifelse(glm.prob>=0.5, 1, 0)
```

```
GLM.PROB <- as.factor(glm.prob)
```

```
table(GLM.PROB, test$Class)
```

#table 함수를 이용하여 정확도를 구하고자 합니다.

```
GLM.PROB  0  1
```

```
0 603 16
```

```
1   3 507
```

```
> (603+507)/(603+507+16+3) #정확도는 전체에서 실제값과 예측값이 일치하는 경우가 차지하는 비율입니다.
```

```
[1] 0.9831709
```

따라서 굉장히 정확도가 높다고 볼 수 있습니다.