

9조 기말 프로젝트 보고서 

* 호텔 리뷰 데이터를 활용한 자연어 처리 및 모델-

| **과목명** | **딥러닝 분석** |
| --- | --- |
| **교수명** | **정원일 교수님** |
| **학 과** | **정보통계 보험수리학과** |
|  | **조성찬(20110726)**  **황병호(20170845)**  **김시헌(20180728)** |
| **제출일** | **2023. 06. 11.** |

**목 차**

[**1. 개요 3**](#_heading=h.txwlcr1tho2b)

[**2. 데이터 탐색 / 데이터 전처리 4**](#_heading=h.k4yk74p83p9i)

[**2.1 데이터 확인 및 시각화 4**](#_heading=h.gjdgxs)

[**2.2 이상값 정제 및 결측값 처리 6**](#_heading=h.30j0zll)

[**3. 모델 학습/ 평가 8**](#_heading=h.1fob9te)

[**3.1 로지스틱 회귀(Logistic Regression) 8**](#_heading=h.3znysh7)

[**3.2 분류트리 (Classification Tree) 10**](#_heading=h.2et92p0)

[**3.3 랜덤 포레스트(Random Forest) 11**](#_heading=h.tyjcwt)

[**3.4 서포트 벡터 머신(Support Vector Machines) 15**](#_heading=h.3dy6vkm)

[**3.4.1 서포트 벡터 분류기(Support Vector Classifier) 15**](#_heading=h.1t3h5sf)

[**3.4.2 서포트 벡터 머신(Support Vector Machine) 16**](#_heading=h.4d34og8)

[**3.5 성능비교 18**](#_heading=h.2s8eyo1)

[**4. 호텔 리뷰 단어 분석**](#_heading=h.cybpobw7nzpx)

**5. 추가 분석**

**6. 결론** [**20**](#_heading=h.cybpobw7nzpx)

# 1. 개요

현대 사회에서는 인터넷을 통한 소비자 리뷰의 중요성이 더욱 부각되어 호텔 업계에서도 소비자 의견을 체계적으로 분석하고 경영 전략에 효과적으로 반영하는 것이 핵심 전략으로 떠오르고 있다. 소비자들의 평가와 리뷰는 호텔 서비스의 질과 만족도를 평가하는 중요한 지표로 인정되고 있다.

본 보고서는 호텔 리뷰 데이터를 활용하여 자연어 처리 및 모델 비교에 대한 연구를 다루고 있다. 약 2만 건의 테스트 데이터와 14만 건의 데이터를 사용하여 분석을 수행하였다. 주요 목표는 다양한 통계적 방법을 활용하여 텍스트 데이터를 분석하고, 모델 간의 성능 차이를 비교한다.

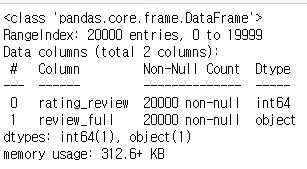
자연어 처리 과정에서는 텍스트 데이터의 전처리, 특징 추출, 모델 학습 순서로 진행한다. 1점부터 5점까지의 평가 점수를 가지고 있는 텍스트 데이터를 바탕으로 자연어 처리를 진행한다. 전처리 과정에서는 정규화, 토큰화, 불용어 처리, 표제어 추출 등을 통해 텍스트 데이터를 정제하였으며, TF-IDF와 같은 특징 추출 방법을 사용하여 데이터를 벡터 형태로 변환한다.

그 후, 딥러닝 모델인 SimpleRNN, LSTM, GRU를 활용하여 모델을 학습시킨 후 성능을 비교하며, 기본적인 통계 모델인 로지스틱 회귀, 나이브 베이즈, 의사결정 트리, 랜덤 포레스트, SVM 등을 비교 분석 대상으로 사용한다. 뿐만 아니라, 비지도 학습 방법 중 하나인 K-means 클러스터링을 이용하여 리뷰 데이터를 클러스터로 그룹화하고, 이를 시각화 하여 클러스터간 특징들을 추출한다.

# 2. 데이터 탐색 / 데이터 전처리

# 데이터 확인 및 시각화

모델을 생성하기에 앞서 EDA를 진행한다. 우선, 데이터 구조는 아래와 같으며 데이터프레임은 raitng값을 갖고 있는 rating\_review 열과 텍스트 데이터로 이루어진 review\_full 열로 구성되어 있다.

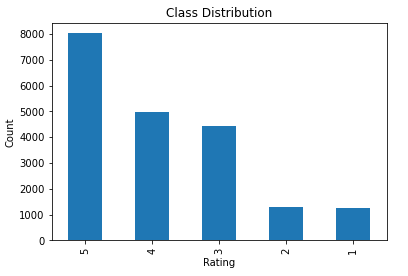


**<그림 1 데이터프레임 구조 확인>**



**<그림 2 데이터프레임 형태 확인>**

그 중 rating\_review 열은 1점부터 5점까지의 평가 점수를 가지고 있고, 점수 분포를 시각화하여 확인하기 전에 중복된 행이 4개 존재하여 제거 후에 19,996개의 데이터 셋에 대하여 rating값의 분포를 확인한다. 그 결과, raitng값은 비교적 1,2점보다 3~5점에 해당하는 데이터가 많음을 확인할 수 있다.



**<그림 3 rating값 분포 확인>**

# 결측치 제거 및 텍스트 데이터 전처리

결측치는 존재하지 않아 텍스트 전처리를 진행한다. 전처리 과정에서는 정규화, 토큰화, 불용어 처리, 표제어 추출 등을 통해 텍스트 데이터를 정제 후, TF-IDF와 같은 특징 추출 방법을 사용하여 벡터 형태로 변환한다.



**<그림 4 텍스트 데이터 전처리>**

**<그림 6 연속형 데이터의 히스토그램 >**

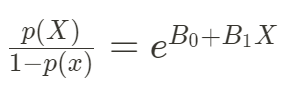
# 3. 모델 학습/평가

모델 학습을 위해 다음과 같은 모델을 적용하였다.

1. **SimpleRNN**
2. **GRU**
3. **LSTM**

# SimpleRNN

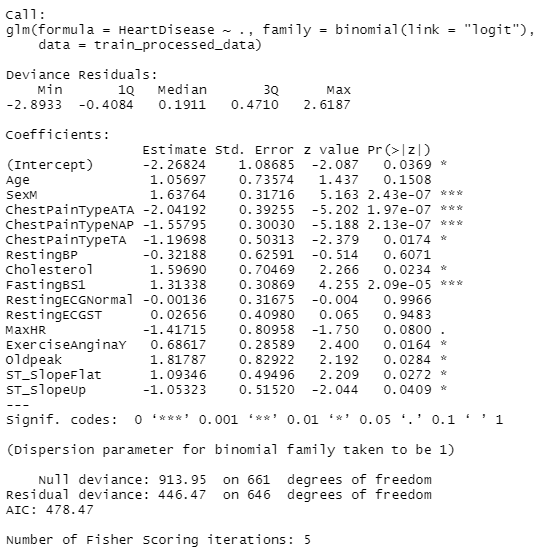
RNN 구조에서 가장 간단한 형태의 RNN 레이어인 simpleRNN을 먼저 적용해 보았다.



**<그림 8 로지스틱 회귀 함수>**

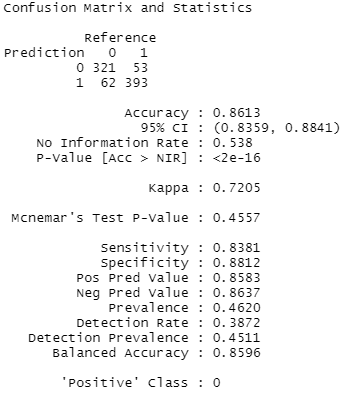
위 식에 로그를 취한 형태인 로짓(logit)을 만들고, 이를 통해 단조 증가 함수를 얻을 수 있다. 로지스틱 회귀 모델에서 설명변수의 한 유닛 증가는 로짓을 계수만큼 변화시킨다.

이후 로짓을 사용해서 가능도함수를 만들고, 가능도함수를 최대화하는 최대가능도추정량(MLE)을 사용해서 계수를 추정할 수 있다.



**<그림 9 다중 로지스틱 회귀 모델 추정량>**

전처리 된 데이터 셋을 사용해서 만든 로지스틱 회귀 모델이다. RestingECG의 경우 p값이 1에 가까운 것을 확인할 수 있다. EDA 과정에서도 해당 변수에 따라 심장병 발병 유무의 차이가 크지 않은 것 역시 확인할 수 있다.

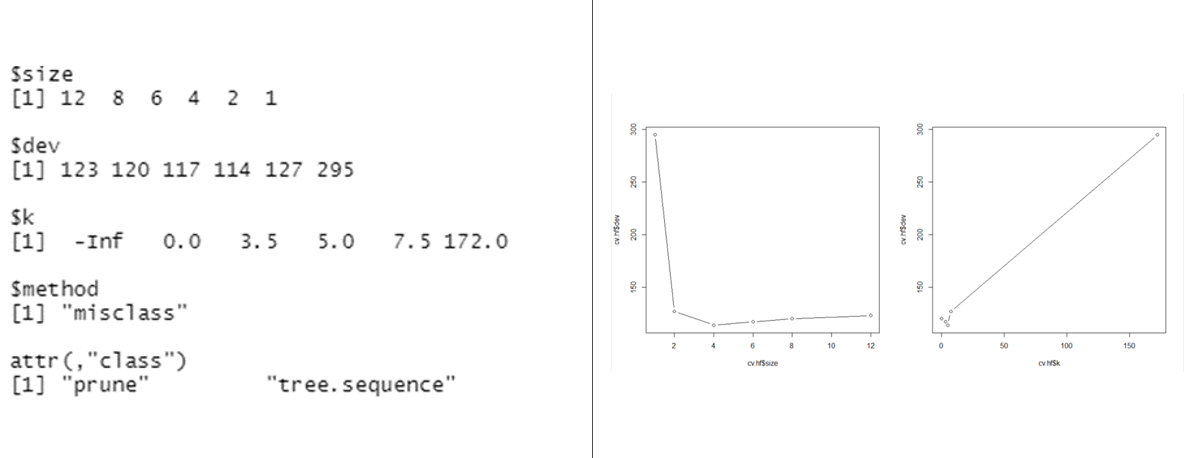


**<그림 10 로지스틱 회귀 분석에 대해 10-fold CV 실행 결과>**

이후 전체 데이터에 대해서 10-fold CV를 실행했다. 10-fold CV를 실행한 로지스틱 회귀 모델의 정확도는 약 86.1%이다.

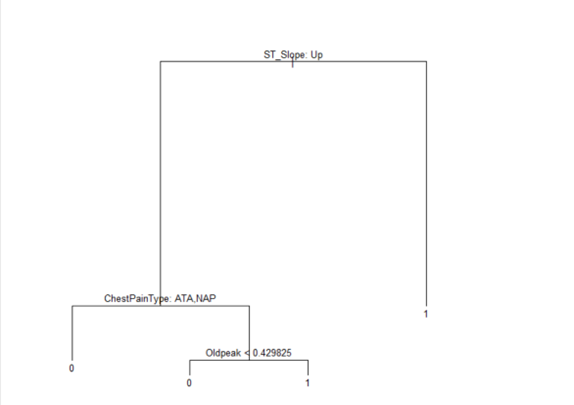
# 분류트리 (Classification Tree)

분류트리는 질적 반응변수를 예측하는 데 사용된다. 각 관측치가 그것이 속하는 영역 내 훈련 관측치들이 가장 많이 포함된 클래스에 속하는지를 예측한다. 결과를 해석하는 데 있어서 특정 터미널 노드 영역에 대응하는 클래스 예측 뿐만 아니라 그 영역에 속하는 훈련 관측치들 사이의 클래스 비율에 관심이 있다.



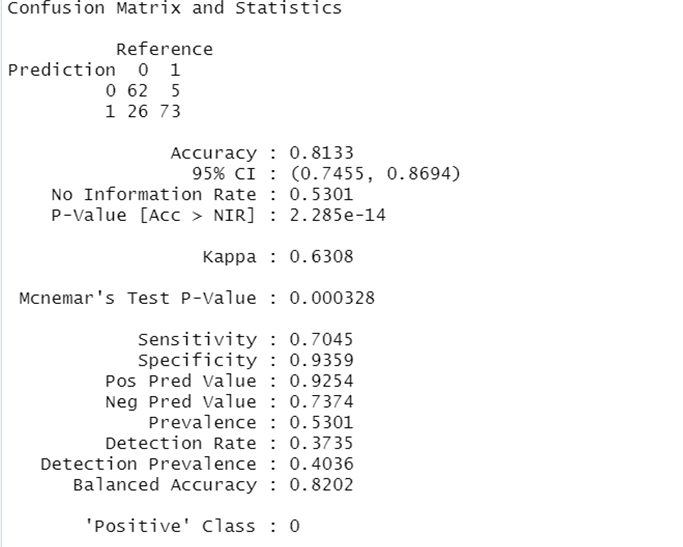
**<그림 11 10-fold CV를 이용한 분류트리 모형 적합 결과 및 시각화>**

size는 terminal node의 수를 의미하는데 이때 CV error rate를 의미하는 dev값이 가장 작을 때의 size를 선택하여 가지치기를 수행한다.



**<그림 12 분류트리 모형에 대해 가지치기 수행 결과>**

다음은 terminal node 값을 4로 지정하여 가지치기를 수행한 결과이다.



**<그림 13 분류트리 모델에 대한 confusionMatrix 수행 결과>**

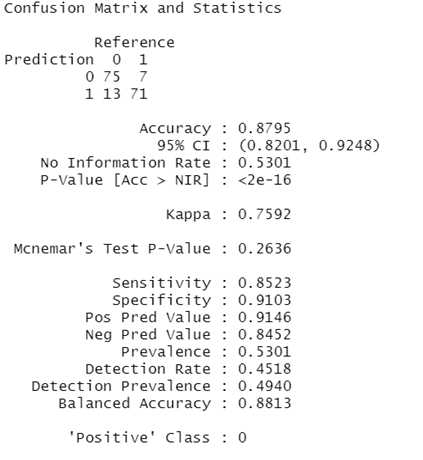
적합 결과, 모델의 정확도(Accuracy)는 약 81.33%이다.

# 랜덤 포레스트(Random Forest)

RandomForest 모델은 bagging 과정을 거쳐 여러 개의 decision tree를 생성하지만, tree에서 분할이 고려될 때 모든 predictors를 사용하지 않고, predictors 중에서 M개를 random sampling 하여 사용한다. 보통 M은 predictors 개수의 제곱근이다. 이를 통해 bagging 방법에서 분산을 줄임으로써 test error를 줄일 수 있다.

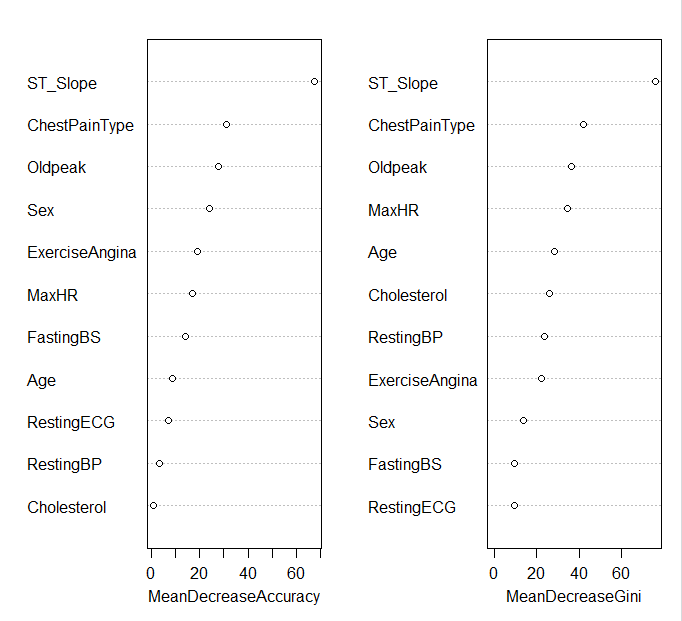
Bagging 방법을 사용할 경우 하나의 강력한 predictor가 있다면 이 predictor로 인해 대부분의 tree의 top split에 강력한 predictor가 위치할 가능성이 높다. 그로 인해 tree들의 모양이 비슷해지기 때문에 예측값들의 correlation이 커져 분산을 줄이지 못하게 되며, 이는 test error의 증가로 이어진다. 이를 방지하기 위해 tree의 모양을 크게 바꿔주는 randomForest 모델을 사용한다.

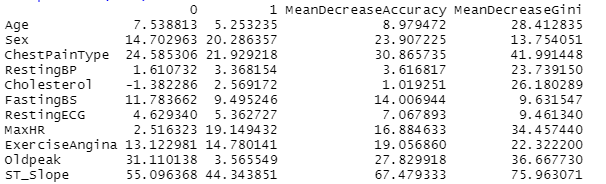
RandomForest 모델의 파라미터 중 하나인 mtry = M을 뜻하며, 모든 설명변수를 사용하는 초기 모델의 경우 11개의 predictors를 사용하므로 차후 모든 모델의 mtry = 3으로 설정한다. 11개의 설명변수를 사용한 결과는 아래와 같다.



**<그림 14 randomForest 초기 모델에 대한 confusionMatrix 수행 결과>**

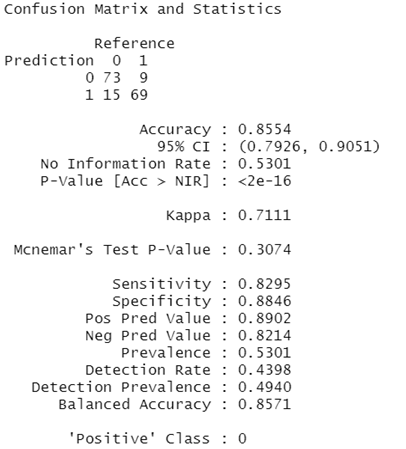
적합결과 accuracy는 약 0.8795로 꽤 높은 수치로 판단된다. 이 모델의 predictor별 중요도는 아래와 같다.





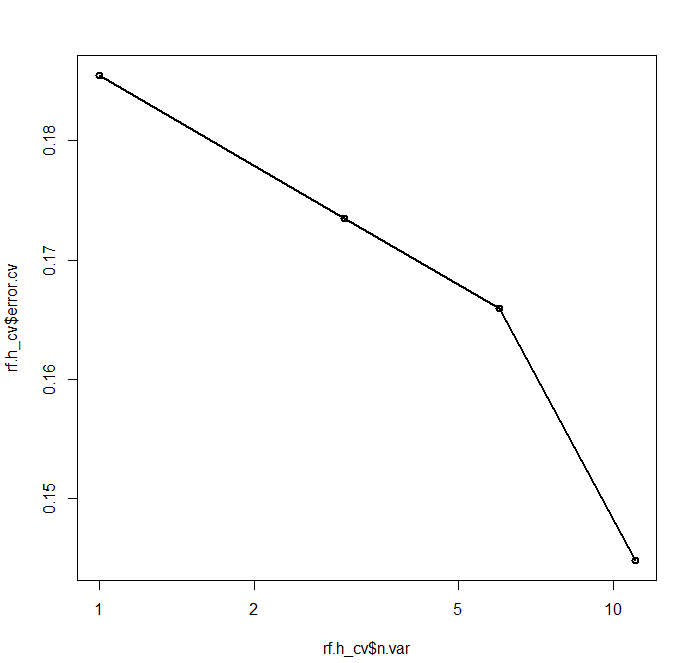
**<그림 15 randomForest 변수별 중요도>**

좌측의 MeanDecreaseAccuracy와 우측의 MeanDecreaseGini에서 중요도가 높은 변수 5개는 ST\_Slope, ChestPainType, Oldpeak, ExerciseArgina, MaxHR, Age이다. 이 다섯개의 변수로 다시 모델을 적합하면 결과는 아래와 같다.



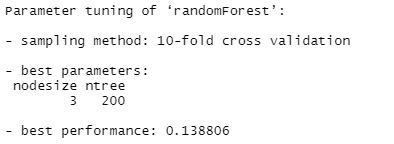
**<그림 16 randomForest 변수선택 모델에 대한 confusionMatrix 수행 결과>**

그 결과 accuracy가 약 0.02하락하였으며, 이는 11개의 설명변수 모두 사용하여 적합한 모델의 성능이 더 높다는 것을 알 수 있다. rfcv() 함수를 통해 선택 변수의 개수별 error rate을 구하여 최종모델을 선정하겠다. K = 5인 K-fold CV를 실행한 결과는 아래와 같다.



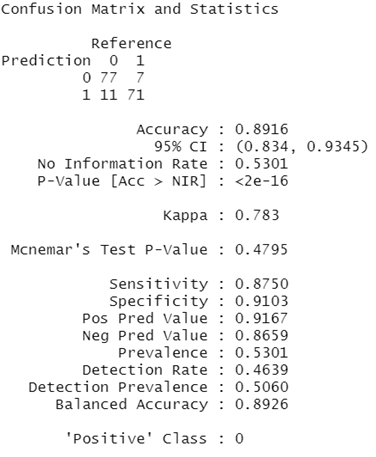
**<그림 17 rfcv함수 실행 결과 변수개수별 error>**

변수가 11개일 때 error rate이 가장 작은 것을 확인할 수 있었고, 이를 통해 초기에 설정한 모델이 가장 성능이 좋은 함수라고 판단할 수 있다. 변수 선택 후 randomForest 모델의 parameter 튜닝을 위해 e1071 패키지의 tune.randomForest를 사용하였다. Parameter 튜닝 결과는 아래와 같다.



**<그림 18 randomForest 모델에 대한 best parameters>**

이에 따라 randomForest의 nodesize를 3으로, ntree는 200으로 설정하고 재적합한 결과 최종 accuracy는 약 0.8916로 초기 모델보다 0.012 상승하였다.

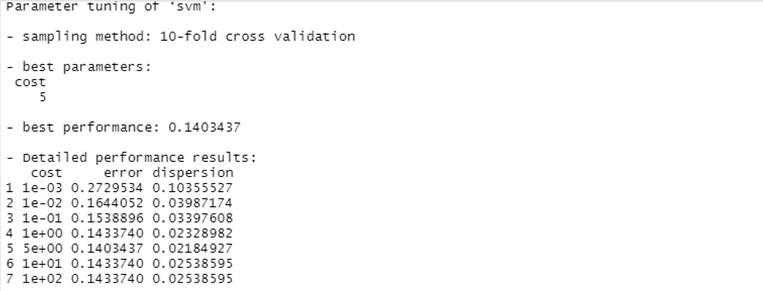


**<그림 19 randomForest 최종모델에 대한 confusionMatrix 수행 결과>**

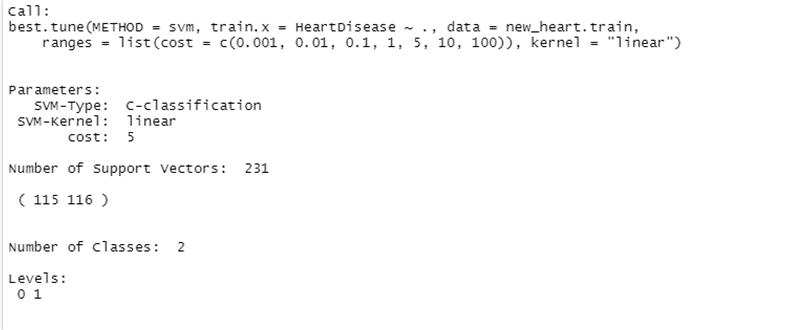
# 서포트 벡터 머신(Support Vector Machines)

# 3.4.1 서포트 벡터 분류기(Support Vector Classifier)

서포트 벡터 분류기는 모든 관측치가 초평면뿐만 아니라 마진의 올바른 쪽에 있도록 가능한 가장 큰 마진을 찾는 대신에, 일부 관측치들은 마진의 옳은 쪽에 있거나 심지어는 초평면의 옳지 않은 쪽에 있을 수 있도록 허용된다. 초평면의 어느 쪽에 검정 관측치가 있느냐에 따라 그 관측치를 분류한다. 초평면은 대부분의 훈련 관측치들을 두 개의 클래스로 정확하게 분류하도록 선택되지만 일부 소수의 관측치들은 잘못 분류될 수도 있다.



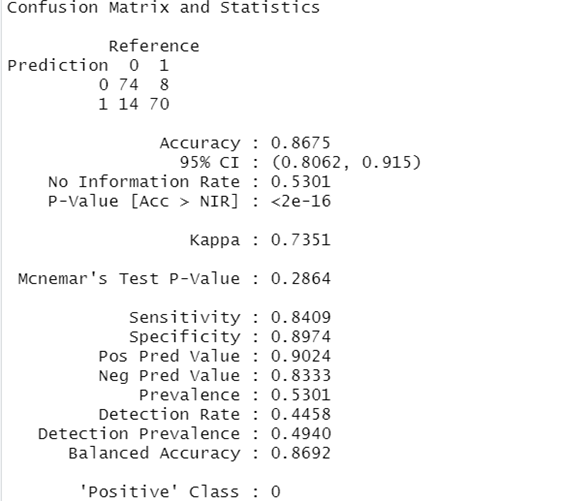
**<그림 20 Support Vector Classifier parmeter tuning >**



**<그림 21 최적의 parameter로 수행된 train model>**

다음은 서포트 벡터 분류기의 조율 파라미터인 C값을 10-fold CV를 이용하여 튜닝한 결과이다.

결과적으로 C=5 일 때 error rate가 가장 낮으므로 이를 파라미터 값으로 지정한다.

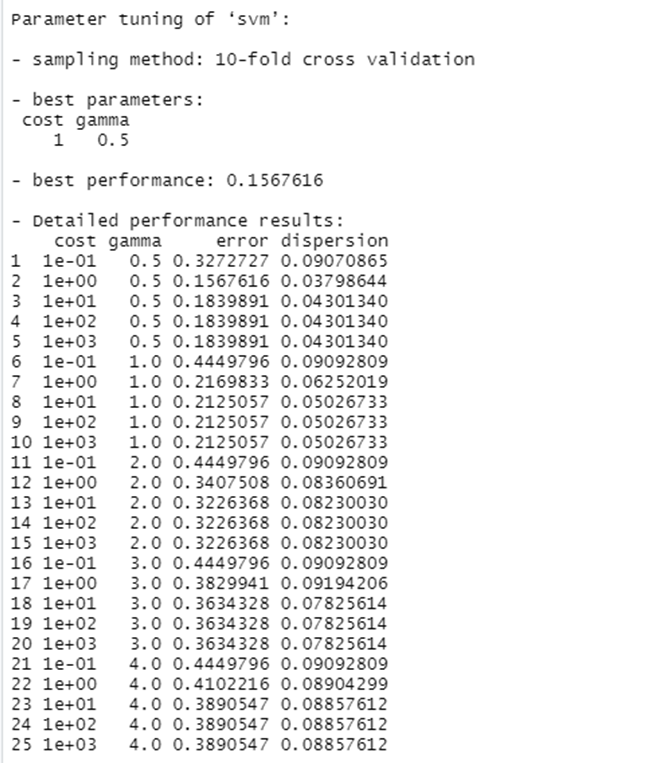


**<그림 22 Support Vector Classifier 모델에 대한 confusionMatrix 수행 결과 >**

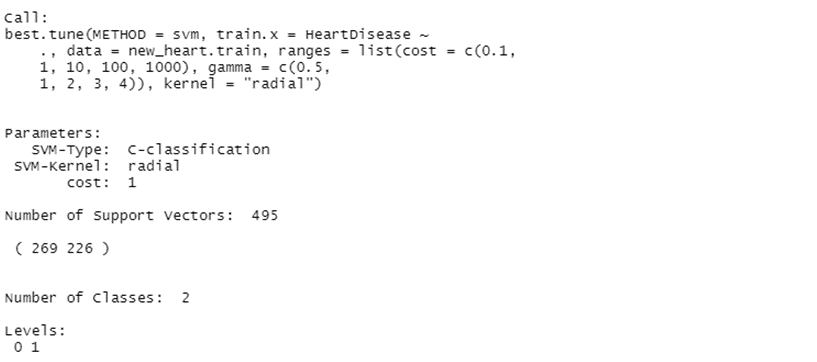
적합 결과, 모델의 정확도(Accuracy)는 약 86.75%이다.

# 3.4.2 서포트 벡터 머신(Support Vector Machine)

서포트 벡터 머신(SVM)은 서포트 벡터 분류기의 확장으로 kernel을 사용하여 특정한 방식으로 변수공간을 확장한 결과이다. 서포트 벡터 분류기와 달리 방사커널(radial kernel)을 사용하며, 이는 비선형 데이터에서 두 클래스를 더 잘 분류할 수 있다.



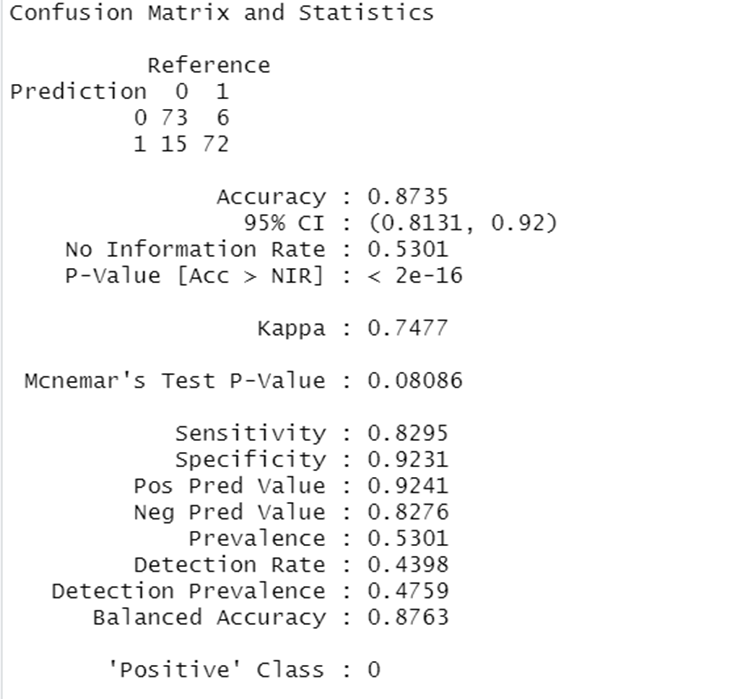
**<그림 23 SVM모델에 대한 parameter tuning>**



**<그림 24 최적의 parameter로 수행된 train model>**

다음은 방사커널(radial kernel)을 사용하여 적합한 SVM모델의 조율 파라미터인 cost와 gamma값을 10-fold CV를 이용하여 튜닝한 결과이다.

결과적으로 C=1, gamma=0.5 일 때 error rate가 가장 낮으므로 이를 파라미터 값으로 지정한다.



**<그림 25 SVM모델에 대한 confusionMatrix 수행 결과>**

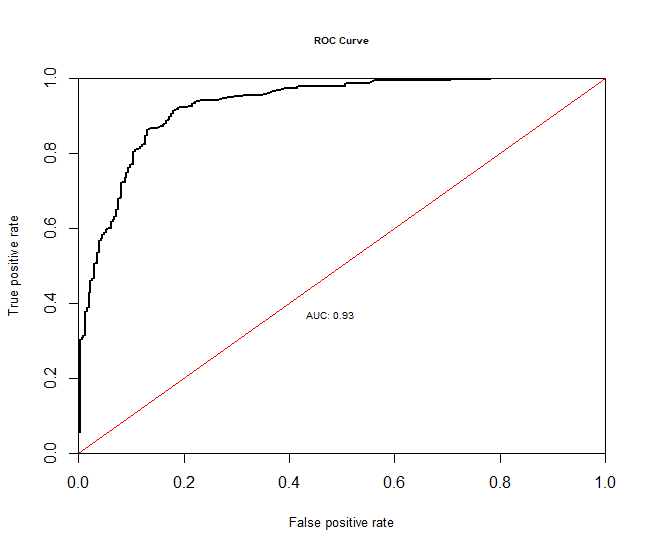
적합 결과, 모델의 정확도(Accuracy)는 약 87.35%이다.

# 성능비교

[로지스틱 회귀]

로지스틱 회귀를 적합하였을 때의 ROC curve 그림이다.

AUC값이 약 0.93이다.

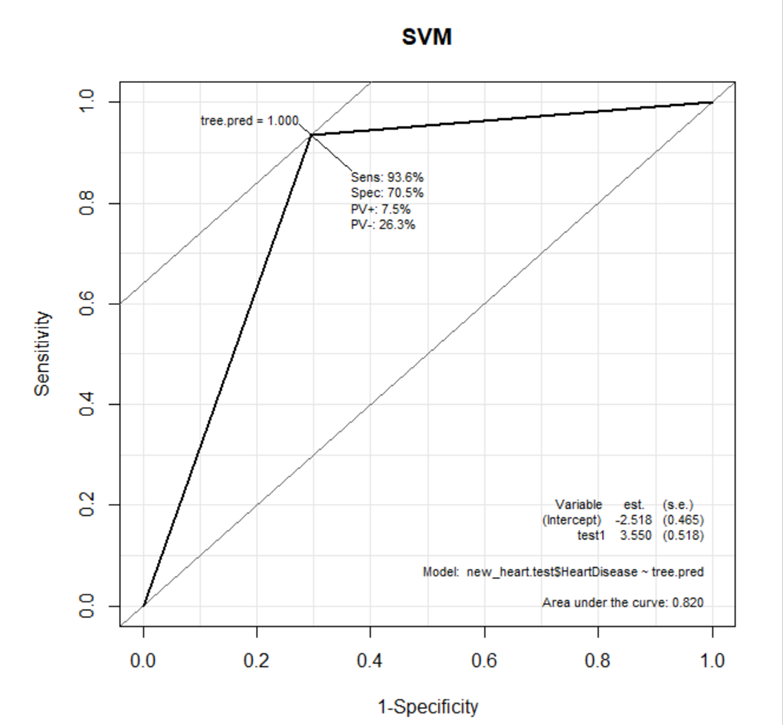


**<그림 26 Logistic Regression ROC curve>**

[분류트리]

Classification Tree를 적합하였을 때의 ROC curve 그림이다.

AUC값이 약 0.82이다.

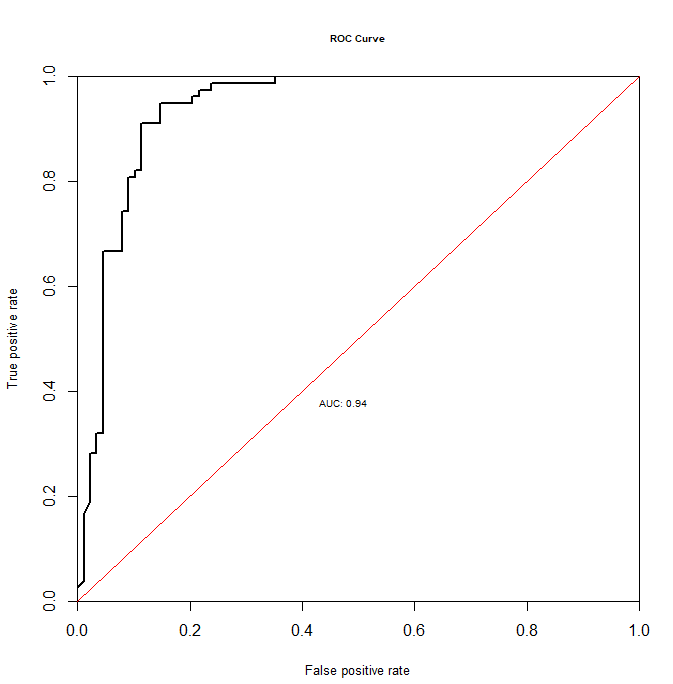


**<그림 27 Classification Tree ROC curve>**

[Random Forest]

Random Forest 적합하였을 때의 ROC Curve 그림이다.

AUC값은 약 0.94이다.

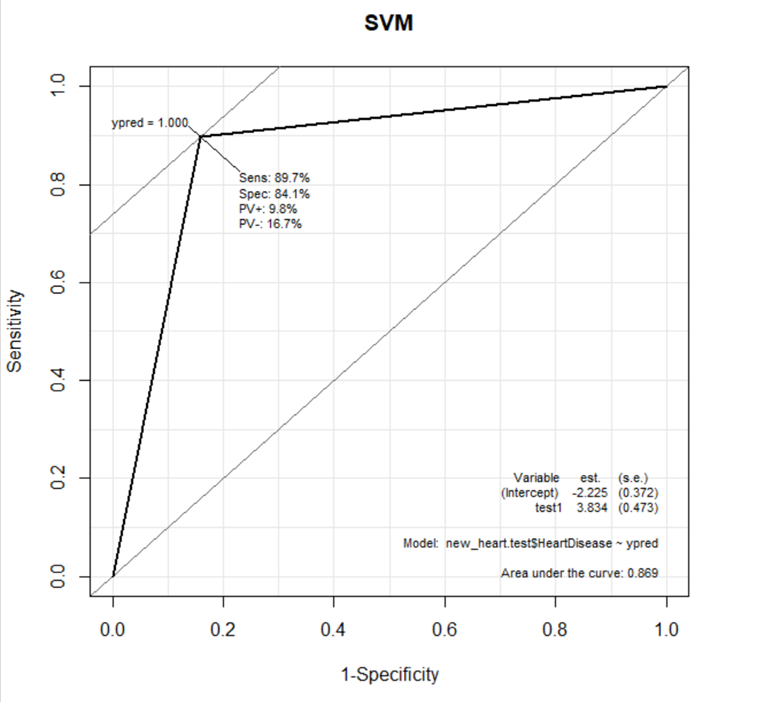


**<그림 28 Random Forest ROC curve>**

[서포트 벡터 분류기]

Support Vector Classifier를 적합하였을 때의 ROC curve 그림이다.

AUC값이 약 0.869이다.

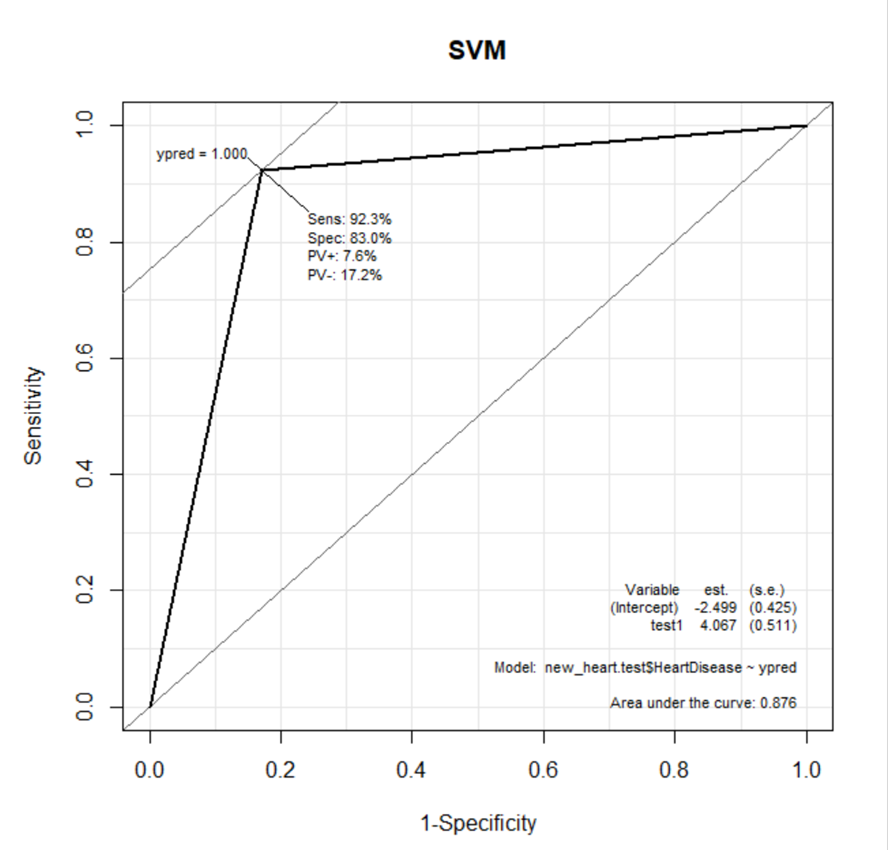


**<그림 29 Support Vector Classifier ROC curve>**

[서포트 벡터 머신]

SVM을 적합하였을 때의 ROC curve 그림이다.

AUC값이 약 0.876이다.



**<그림 30 SVM ROC curve>**

# 4. [호텔 리뷰 단어 분석](#_heading=h.cybpobw7nzpx)

주어진 분석을 수행하기 위해 호텔의 리뷰인 텍스트 데이터를 이용하여 분석하는 방법과 rating값을 이용하여 분석하는 방법 2가지를 제시한다.

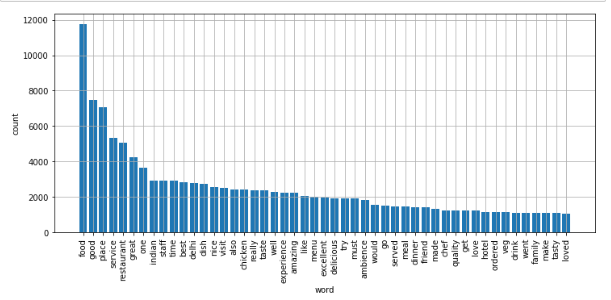
# 4.1 텍스트 데이터 자체를 이용한 분석

어떤 단어가 좋은 호텔 리뷰에 많이 쓰이고 어떤 단어가 나쁜 호텔 리뷰에 많이 쓰이는지 분석하기 위해 rating값이 1,2점은 부정적인 평가로, 4,5점은 긍정적인 평가로 기준을 세운다. rating값이 3점인 리뷰는 중립적인 평가로 간주하고 제외한 후 분석을 수행한다.

4.1.1 단어 빈도 분석 및 Word Cloud

TF-IDF를 활용하여 좋은 리뷰와 나쁜 리뷰 각각에서 단어의 중요도를 계산하고 TF-IDF점수가 높은 단어를 확인하여 차이를 분석하고자 한다. 또한 각 리뷰에서 워드 클라우드를 생성하여 차이를 시각적으로 확인한다.

-좋은 리뷰에서 단어 빈도 수 상위 50개의 단어 시각화



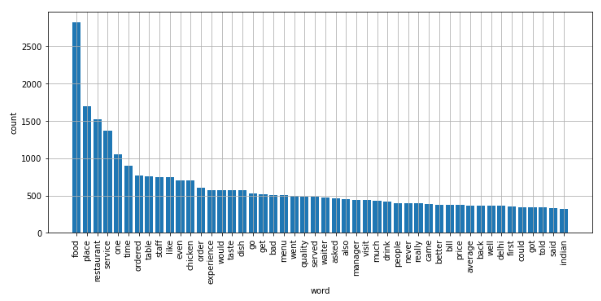
<그림 좋은 리뷰에서의 상위 50개 단어 히스토 그램>

좋은 호텔 리뷰에서 주로 "food", "good", "place", "service", "great" 등과 같은 단어들이 등장함을 알 수 있다. 구체적으로는, 각각 11771, 7488, 7063, 5354, 4235 횟수만큼 나타났다. 이를 통해 음식, 좋은 서비스, 멋진 장소 등이 좋은 리뷰에서 강조되는 것을 알 수 있다.



<그림 좋은 리뷰에서의 상위 50개 단어 워드 클라우드>

-나쁜 리뷰에서 단어 빈도 수 상위 50개의 단어 시각화



<그림 나쁜 리뷰에서의 상위 50개 단어 히스토그램>

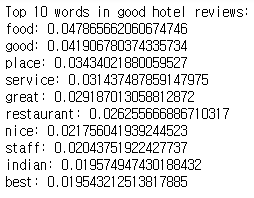
나쁜 호텔 리뷰에서 주로 "food", "bad", "place", "restaurant", "service" 등과 같은 단어들이 등장함을 알 수 있다. 구체적으로는, 각각 2825, 509, 1693, 1525, 1365 횟수만큼 나타났다. 이를 통해 음식, 나쁜 서비스, 나쁜 장소 등이 나쁜 리뷰에서 강조되는 것을 알 수 있다.



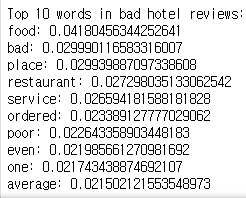
<그림 나쁜 리뷰에서의 상위 50개 단어 워드 클라우드>

4.1.2 TF-IDF를 활용한 단어 중요도 계산

단어 빈도 수와 워드 클라우드 시각화만을 통한 해석은 객관화가 부족하다고 생각하여 객관적인 지표인 TF-IDF를 활용하였다. TF-IDF는 단어의 빈도를 고려하여 계산되고 단어의 상대적인 빈도를 파악할 수 있다. 높은 점수를 가질수록 해당 문서에서 자주 나오는 것으로 식별할 수 있다. 따라서 좋은 호텔 리뷰와 나쁜 호텔 리뷰에서 각각 TF-IDF 점수 상위 10개의 단어들을 확인하였다. 결과는 4.1.1의 방법과 매우 유사하게 나옴을 확인할 수 있다.



<그림 좋은 호텔 리뷰에서 TF-IDF 점수가 높은 상위 10개 단어>



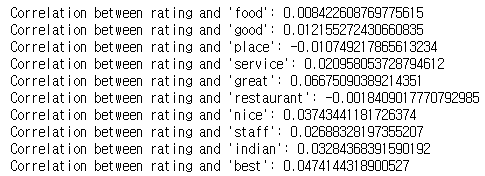
<그림 나쁜 호텔 리뷰에서 TF-IDF 점수가 높은 상위 10개 단어>

# 4.2 Rating값을 활용한 분석

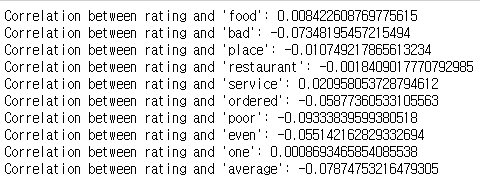
4.1은 텍스트 데이터 자체를 이용해서 분석한 반면 4.2는 rating값을 활용하여 분석하고자 한다.

호텔 리뷰의 rating값과 특정 단어들의 등장 빈도 혹은 중요도를 비교하여, 특정 단어들이 좋은 리뷰 혹은 나쁜 리뷰에서 더 자주 등장하는지 또한 rating값과 어떤 상관 관계를 가지는지 등을 분석한다.

방법은 TF-IDF를 활용하여 리뷰 텍스트를 벡터화한 후 벡터화 된 리뷰와 rating값을 결합한 데이터 프레임을 생성하여 4.1에서 도출한 각 리뷰의 상위 10개 특정 단어들과 rating값의 상관 관계를 계산한다.



<그림 좋은 호텔 리뷰 상위10 특정 단어의 등장 빈도 혹은 중요도와 rating 값의 상관 관계>



<그림 나쁜 호텔 리뷰 상위10 특정 단어의 등장 빈도 혹은 중요도와 rating 값의 상관 관계 계산>

[결론]

: 단어 빈도 수 및 WordCloud 분석뿐만 아니라 이외의 분석을 통해서도 좋은 호텔 리뷰에서는 음식, 좋은 서비스, 멋진 장소 등이 강조되는 경향을 보이는 반면, 나쁜 호텔 리뷰에서는 음식, 나쁜 서비스, 나쁜 장소 등이 강조되는 경향을 보인다. 이를 통해 음식, 서비스, 장소, 레스토랑 등과 같은 요소들이 호텔 평가에 많은 영향을 미치는 것으로 해석할 수 있다.

팀원 별 역할

공통 : 보고서 작성

조성찬 : EDA를 위한 데이터 시각화 + 로지스틱 회귀 모델 적합 및 분석 + 보고서의 전체적 프레임 구축

황병호 : 분류나무 모델, Support Vector Classifier, Support Vector Machine 모델 적합 및 분석 + 전반적인 모델 조사 및 역할 분배

김민성 : EDA 이후 전처리 및 데이터 검정 + random forest 모델 적합 및 분석 + 코드 함수화 및 통일