

9조 기말 프로젝트 보고서 

* 호텔 리뷰 데이터를 활용한 자연어 처리 및 모델-

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

**목 차**

[**1. 개요 3**](#_Toc121500191)

[**2. 데이터 탐색 / 데이터 전처리 4**](#_Toc121500192)

[**2.1 데이터 확인 및 시각화 4**](#_Toc121500193)

[**2.2 이상값 정제 및 결측값 처리 6**](#_Toc121500194)

[**3. 모델 학습/ 평가 8**](#_Toc121500195)

[**3.1 로지스틱 회귀(Logistic Regression) 8**](#_Toc121500196)

[**3.2 분류트리 (Classification Tree) 10**](#_Toc121500197)

[**3.3 랜덤 포레스트(Random Forest) 11**](#_Toc121500198)

[**3.4 서포트 벡터 머신(Support Vector Machines) 15**](#_Toc121500199)

[**3.4.1 서포트 벡터 분류기(Support Vector Classifier) 15**](#_Toc121500200)

[**3.4.2 서포트 벡터 머신(Support Vector Machine) 16**](#_Toc121500201)

[**3.5 성능비교 18**](#_Toc121500202)

[**4. 결론 20**](#_Toc121500203)

# 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.

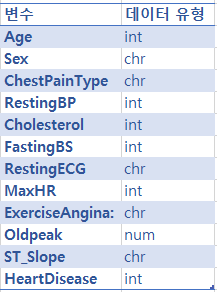
자연어 처리 과정에서는 텍스트 데이터의 전처리, 특징 추출, 모델 학습 순서로 진행한다. 1점부터 5점까지의 평가 점수를 가지고 있는 텍스트 데이터를 바탕으로 자연어 처리를 진행한다. 전처리 과정에서는 정규화, 토큰화, 불용어 처리, 표제어 추출 등을 통해 텍스트 데이터를 정제하였으며, TF-IDF와 같은 특징 추출 방법을 사용하여 데이터를 벡터 형태로 변환한다.

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

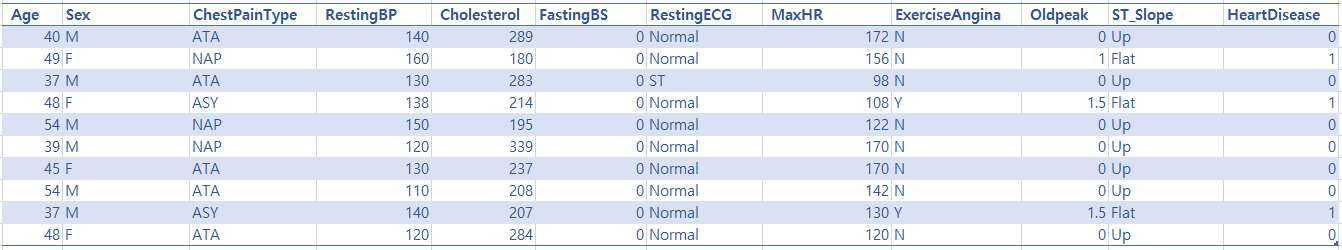
# 데이터 확인 및 시각화

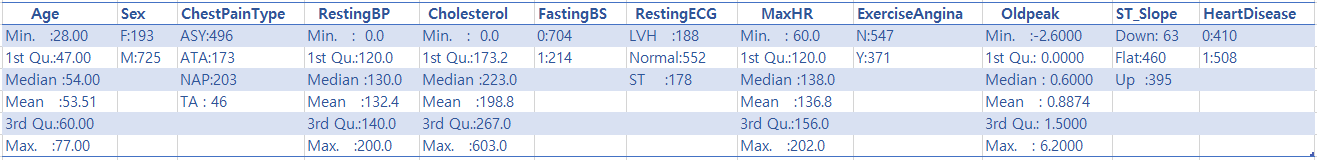
모델을 생성하기에 앞서 EDA를 진행한다. 데이터 구조를 파악하면서 변수의 데이터 타입 변경 및 표준화를 진행한다. 또한 시각화 된 결과를 바탕으로 변수 간의 상관관계를 간접적으로 확인하고, 이어서 기초 통계량을 분석한다.



<그림 1 데이터 셋 변수 확인>

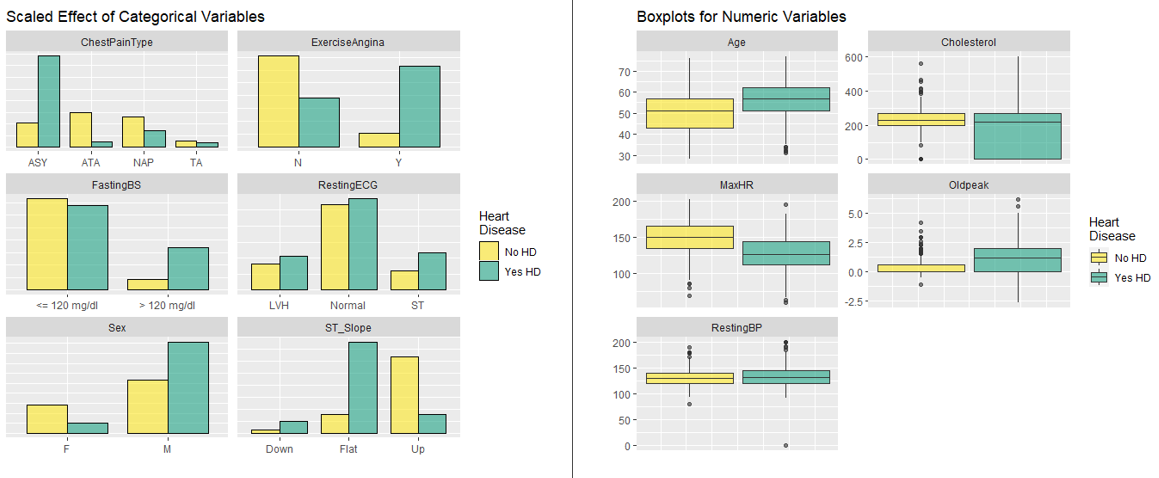
제공된 데이터는 총 12개의 변수로 이뤄져 있으며, 1개의 반응변수와 11개의 설명변수로 이뤄져 있다. 데이터 분석을 진행하기 위해서 각각의 데이터 형태를 수정한다. 범주형 데이터는 R에서 Factor를 사용하고, 연속형 데이터는 Numeric으로 변경한다. 이후 시각화를 진행하기에 앞서 기초 통계량을 확인한다.





<그림 2 데이터 요약 및 형태 확인>

각각의 설명변수에 대한 빈도, 평균, 중앙값, 그리고 최소 최대값을 확인할 수 있다. 이후 연속형 데이터의 경우, 이상값 및 결측값 처리, 변수별 정규화를 진행한다.

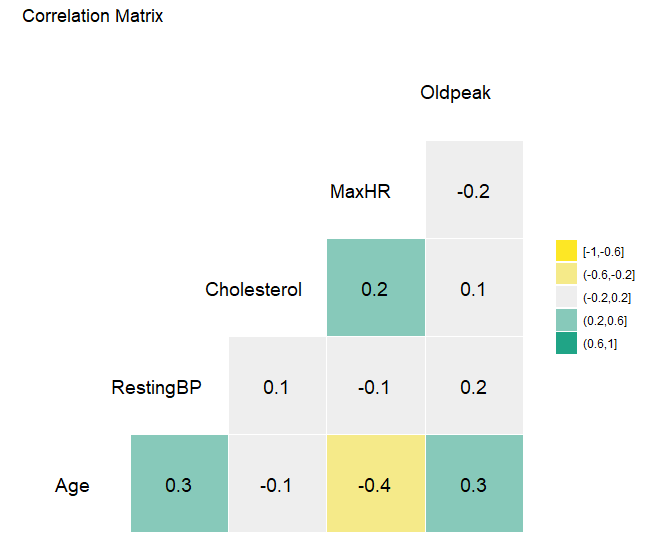


<그림 3 심장병 발병 여부에 따른 히스토그램과 상자그림>

반응변수에 대한 각 설명변수의 히스토그램 그래프 그린다. 설명변수에 따라서 심장병 발병 정도의 확인과 더불어 해당 변수의 영향력을 1차적으로 확인할 수 있다.

혈당(FastingBS)의 경우, 120mg/dl이 넘는 경우 심장병 발병이 많은 것으로 확인된다. 또한 성별 변수를 보면 남성과 여성의 심장병 발병 빈도에 대한 차이를 확인할 수 있다.

연속형 변수에 대한 시각화를 통해, 이상값 및 결측값를 확인할 수 있다. 심장병 발병 유무에 따라 나이, Oldpeak의 평균이 다소 높고, 최대 심박수(MaxHR) 평균이 낮은 것을 확인할 수 있다. 콜레스테롤(Cholesterol)은 값이 0인 결측값이 존재하는 것으로 확인된다.

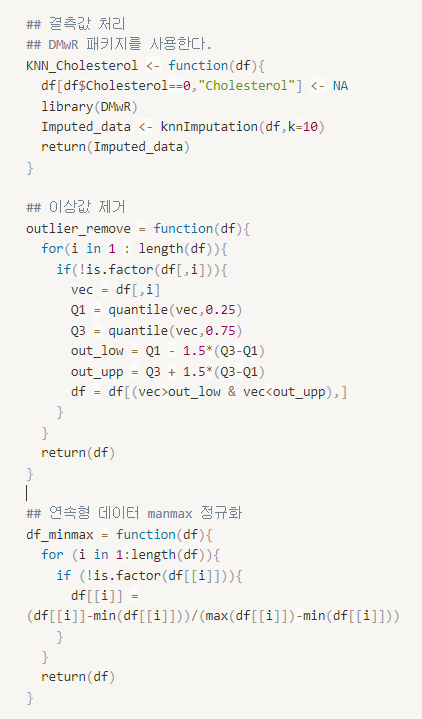


<그림 4 데이터 전처리 전 연속형 변수에 대한 상관행렬>

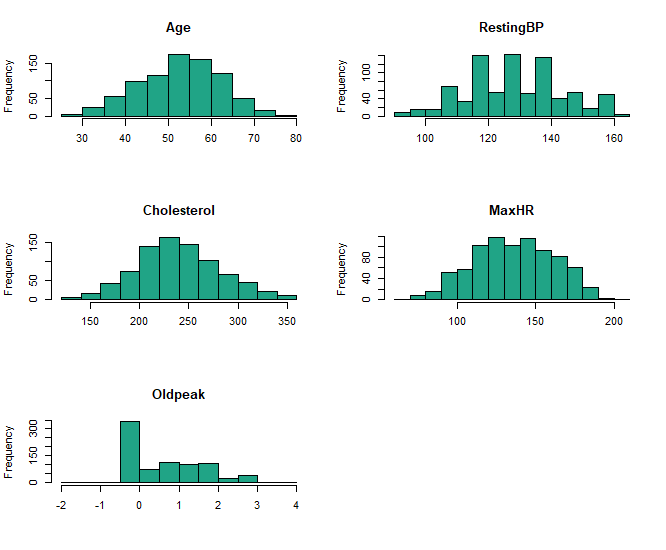
연속형 변수에 대한 상관관계를 확인하기 위한 그래프를 그린다. Age와 RestingBP, Age와 Olpeak 그리고 Cholesterol과 MaxHR 간의 양의 상관관계를 확인할 수 있었고, Age와 MaxHR 사이의 음의 상관관계를 확인할 수 있다. 나머지 관계는 다소 약한 상관관계임을 확인할 수 있다.

# 이상값 정제 및 결측값 처리

시각화와 기초 통계 분석를 통해 이상값과 결측값의 유무를 확인할 수 있다. IQR를 사용하여 일부 변수에서 발생한 이상치를 제거하고, KNN을 사용하여 결측값을 보정 처리한다.

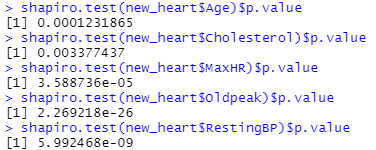


<그림 5 결측값 및 이상값 처리>



<그림 6 연속형 데이터의 히스토그램 >

연속형 변수의 표준화를 진행하기에 앞서, 각 변수가 정규분포를 따르는지 확인한다. 히스토그램을 통해 Age, Cholesterol, MaxHR의 경우 정규성을 만족하는 것으로 보이지만, 다른 변수에 대해서는 샤피로-윌크(Shapiro-wilk) 검정을 통해 정규성 확인을 수행한다.



<그림 7 샤피로-윌크 검정 결과 >

정규성 검정 결과 모든 연속형 변수들이 ‘정규성을 만족한다.’는 귀무가설을 기각하므로 표준화는 MinMax scaling를 사용한다. 데이터의 전처리 과정은 결측값 처리, 이상값 제거, minmax 정규화 순으로 진행하며, 전체 918개의 데이터 중 89개의 데이터를 제거했다.

# 3. 모델 학습/평가

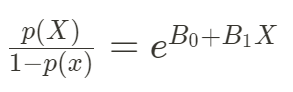
다양한 통계 기법을 사용하여, 심장병 발병 여부를 파악하는 예측 모델을 생성했다. 사용한 통계 기법은 다음과 같은 순서로 진행했다.

1. **로지스틱 회귀 분석**
2. **분류트리(Classification Tree)**
3. **랜덤 포레스트**
4. **서포트 벡터 머신(Support Vector Machine)**

가장 광범위하게 사용하는 로지스틱 회귀분석부터 트리기반의 랜덤 포레스트, 그리고 서포트 벡터 머신을 사용했다. 부족한 데이터를 보완하기 위해 재표본 추출 방법으로 K-fold 교차 검증을 사용했다.

# 로지스틱 회귀(Logistic Regression)

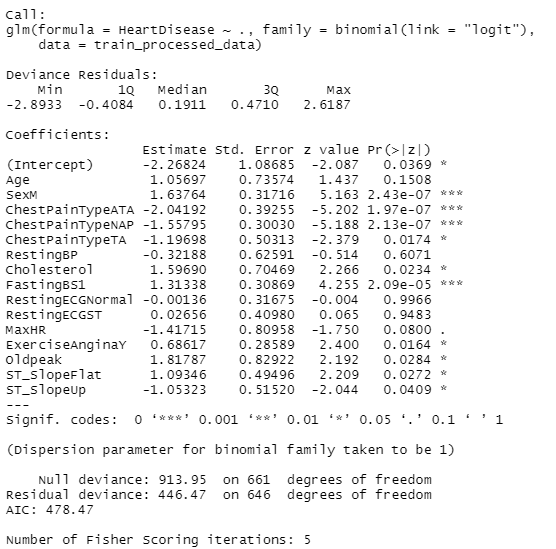
반응변수가 심장병 발병에 따른 두개의 범주로 되어 있으므로 로지스틱 회귀 분석을 사용한다. 설명변수와 반응변수의 관계를 추정하기 위해 로지스틱 함수를 사용하고, 이후 최대가능도를 사용해 계수를 추정한다. 아래의 식은 로지스틱 회귀에서 사용하는 함수이다.



<그림 8 로지스틱 회귀 함수>

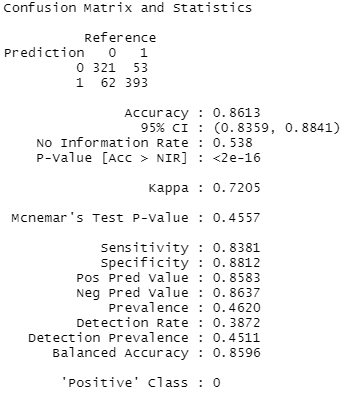
위 식에 로그를 취한 형태인 로짓(logit)을 만들고, 이를 통해 단조 증가 함수를 얻을 수 있다. 로지스틱 회귀 모델에서 설명변수의 한 유닛 증가는 로짓을 계수만큼 변화시킨다.

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.

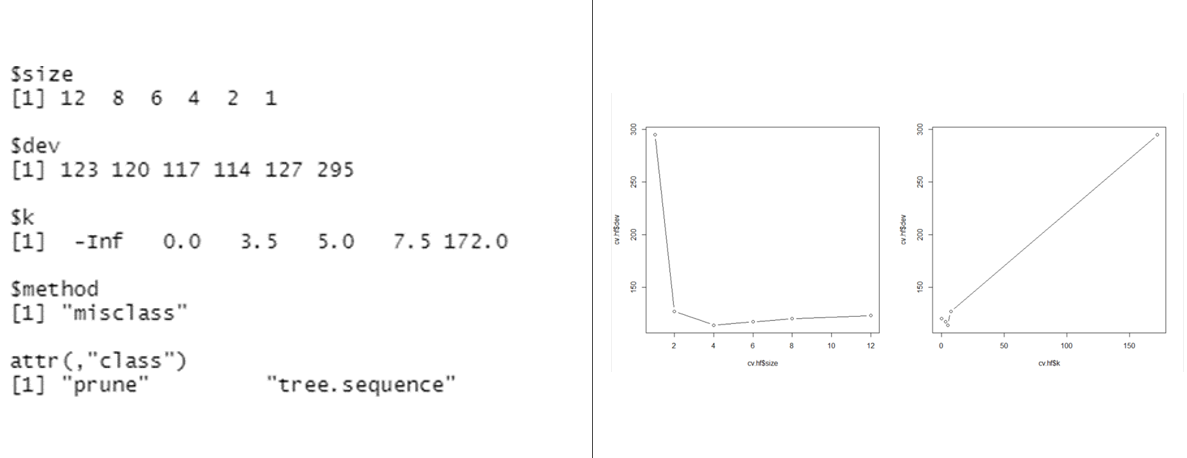


<그림 10 로지스틱 회귀 분석에 대해 10-fold CV 실행 결과>

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

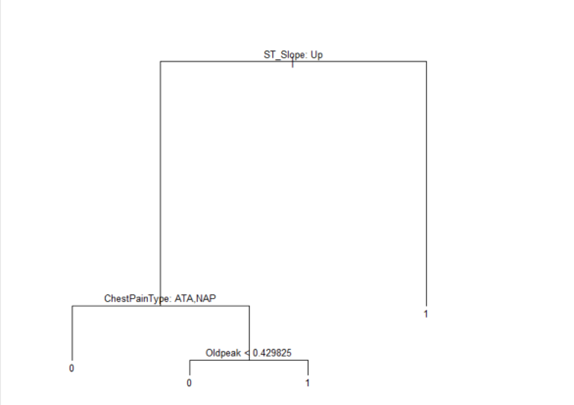
# 분류트리 (Classification Tree)

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.



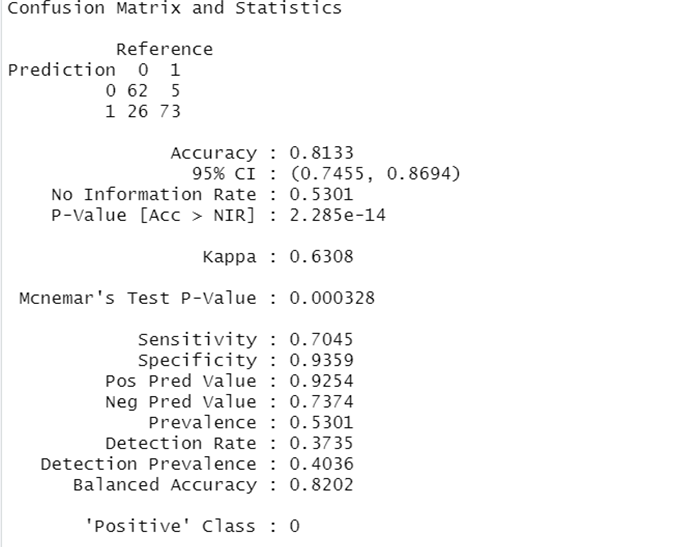
**<그림 11 10-fold CV를 이용한 분류트리 모형 적합 결과 및 시각화>**

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>**

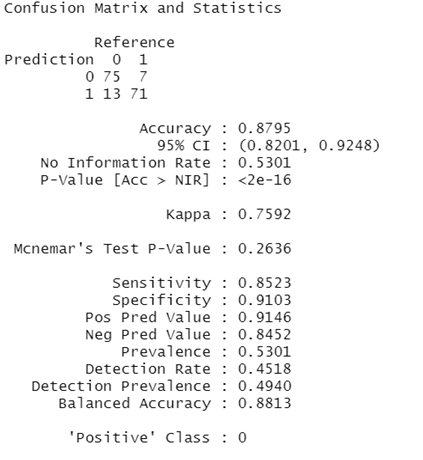
적합 결과, 모델의 정확도(Accuracy)는 약 81.33%이다.

# 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.

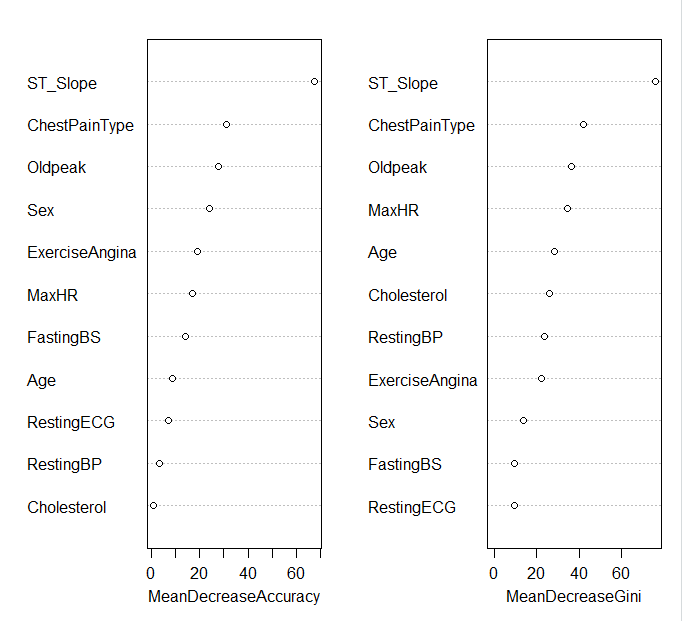
Bagging 방법을 사용할 경우 하나의 강력한 predictor가 있다면 이 predictor로 인해 대부분의 tree의 top split에 강력한 predictor가 위치할 가능성이 높다. 그로 인해 tree들의 모양이 비슷해지기 때문에 예측값들의 correlation이 커져 분산을 줄이지 못하게 되며, 이는 test error의 증가로 이어진다. 이를 방지하기 위해 tree의 모양을 크게 바꿔주는 randomForest 모델을 사용한다.

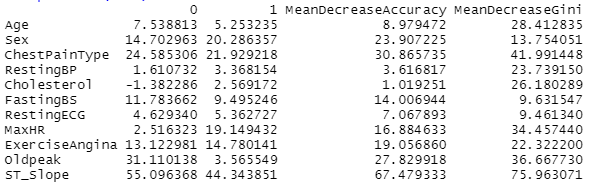
RandomForest 모델의 파라미터 중 하나인 mtry = M을 뜻하며, 모든 설명변수를 사용하는 초기 모델의 경우 11개의 predictors를 사용하므로 차후 모든 모델의 mtry = 3으로 설정한다. 11개의 설명변수를 사용한 결과는 아래와 같다.



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

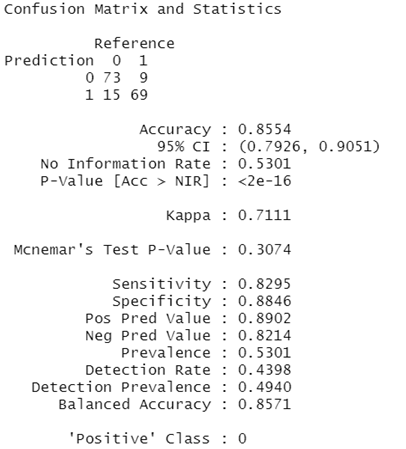
적합결과 accuracy는 약 0.8795로 꽤 높은 수치로 판단된다. 이 모델의 predictor별 중요도는 아래와 같다.





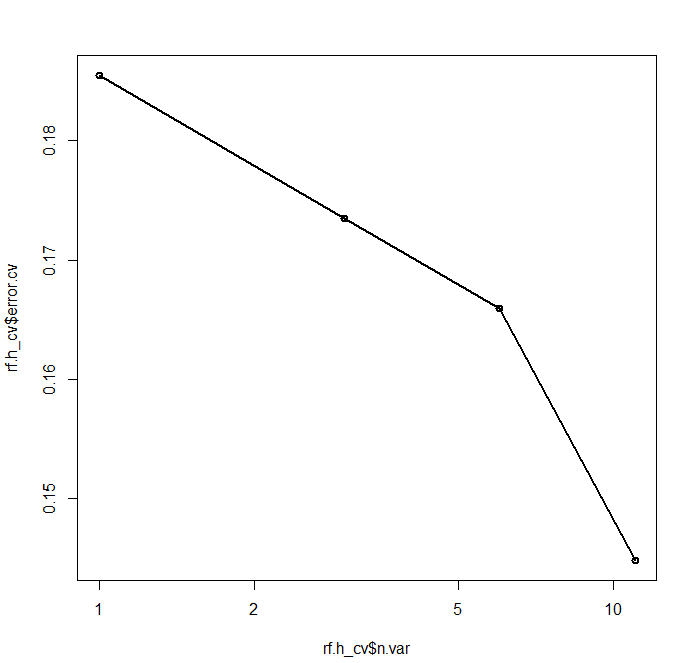
**<그림 15 randomForest 변수별 중요도>**

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



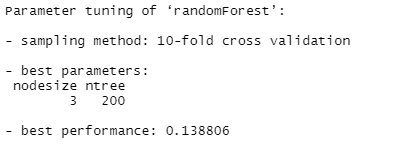
**<그림 16 randomForest 변수선택 모델에 대한 confusionMatrix 수행 결과>**

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.



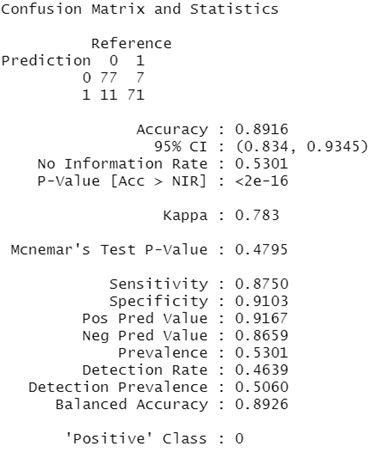
**<그림 17 rfcv함수 실행 결과 변수개수별 error>**

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.

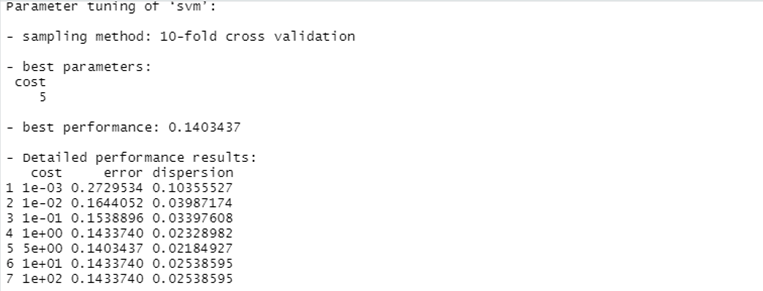


**<그림 19 randomForest 최종모델에 대한 confusionMatrix 수행 결과>**

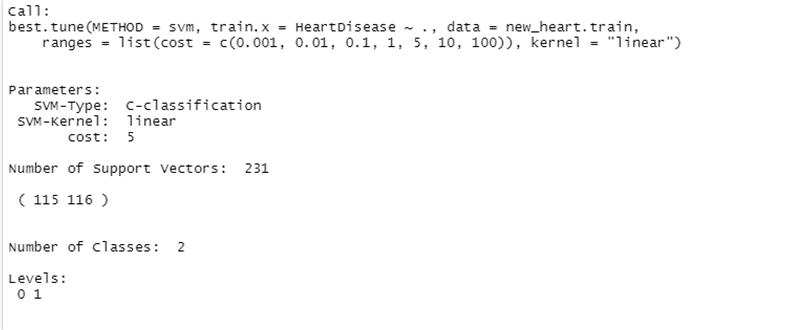
# 서포트 벡터 머신(Support Vector Machines)

# 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.



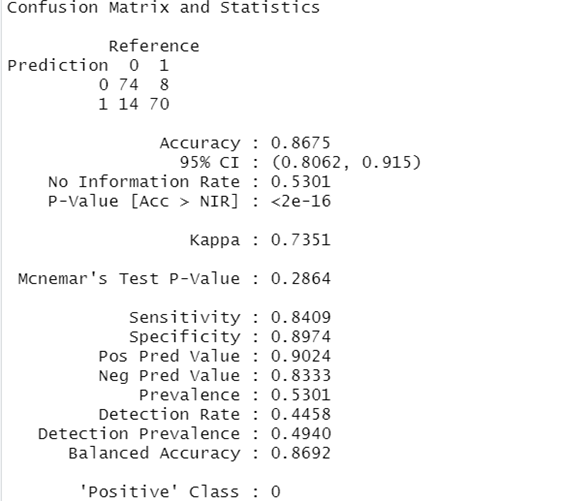
**<그림 20 Support Vector Classifier parmeter tuning >**



**<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.

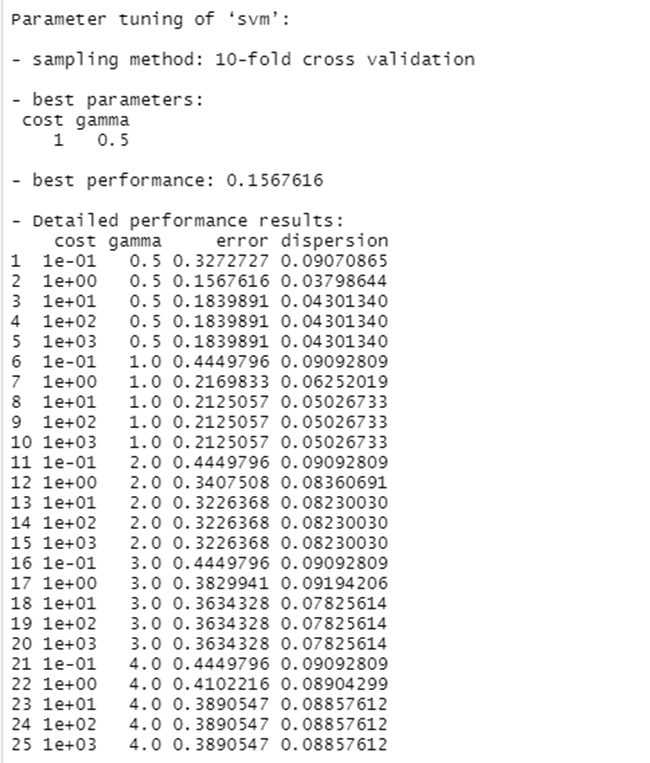


**<그림 22 Support Vector Classifier 모델에 대한 confusionMatrix 수행 결과 >**

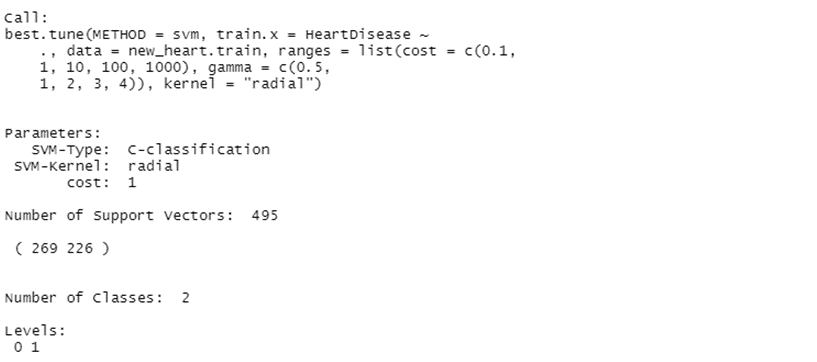
적합 결과, 모델의 정확도(Accuracy)는 약 86.75%이다.

# 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.



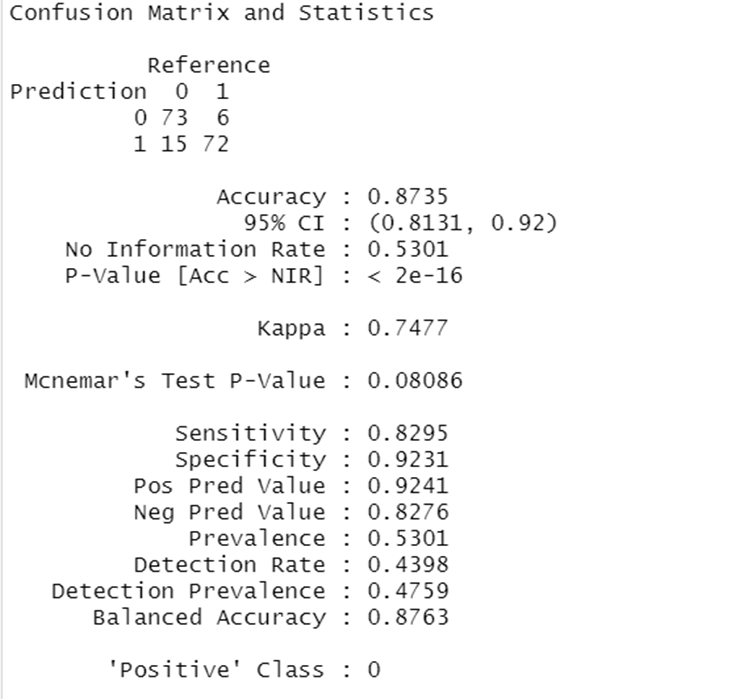
**<그림 23 SVM모델에 대한 parameter tuning>**



**<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>**

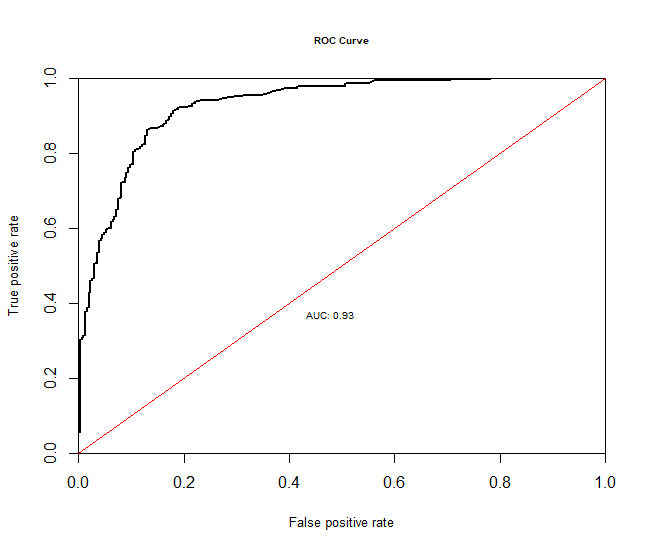
적합 결과, 모델의 정확도(Accuracy)는 약 87.35%이다.

# 성능비교

[로지스틱 회귀]

로지스틱 회귀를 적합하였을 때의 ROC curve 그림이다.

AUC값이 약 0.93이다.

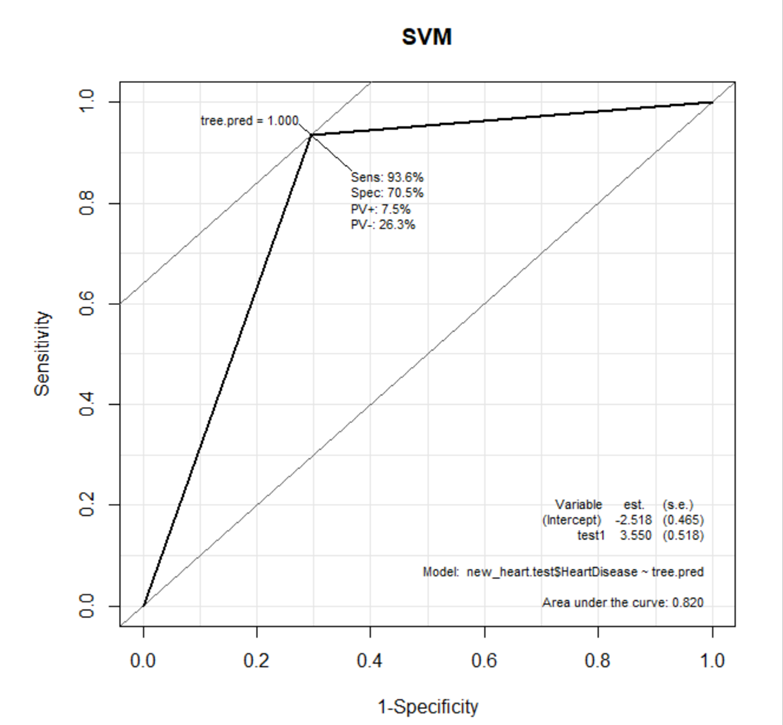


**<그림 26 Logistic Regression ROC curve>**

[분류트리]

Classification Tree를 적합하였을 때의 ROC curve 그림이다.

AUC값이 약 0.82이다.

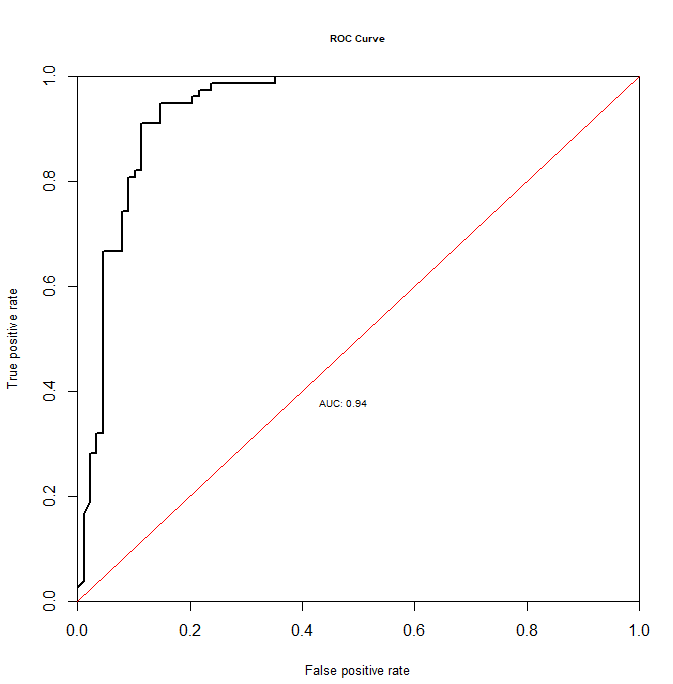


**<그림 27 Classification Tree ROC curve>**

[Random Forest]

Random Forest 적합하였을 때의 ROC Curve 그림이다.

AUC값은 약 0.94이다.

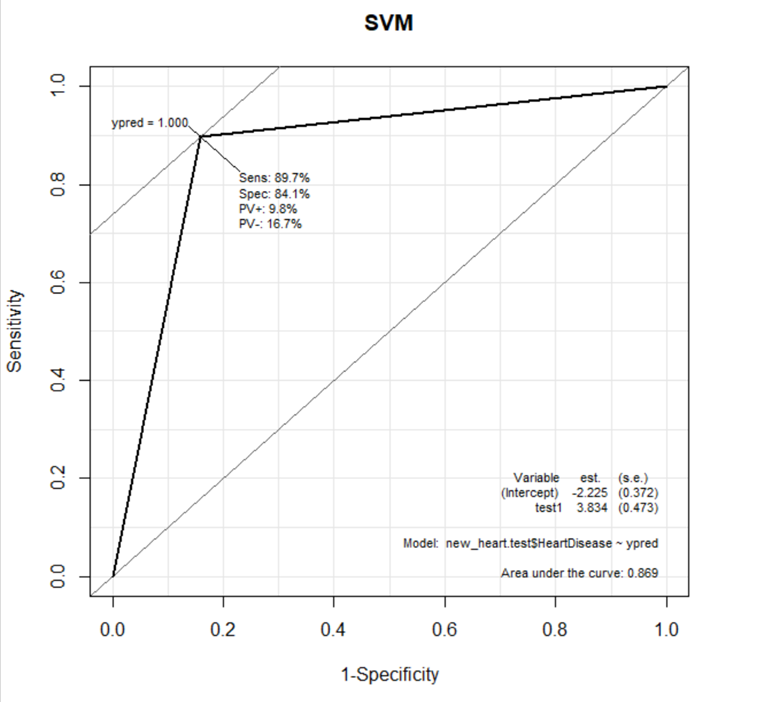


**<그림 28 Random Forest ROC curve>**

[서포트 벡터 분류기]

Support Vector Classifier를 적합하였을 때의 ROC curve 그림이다.

AUC값이 약 0.869이다.

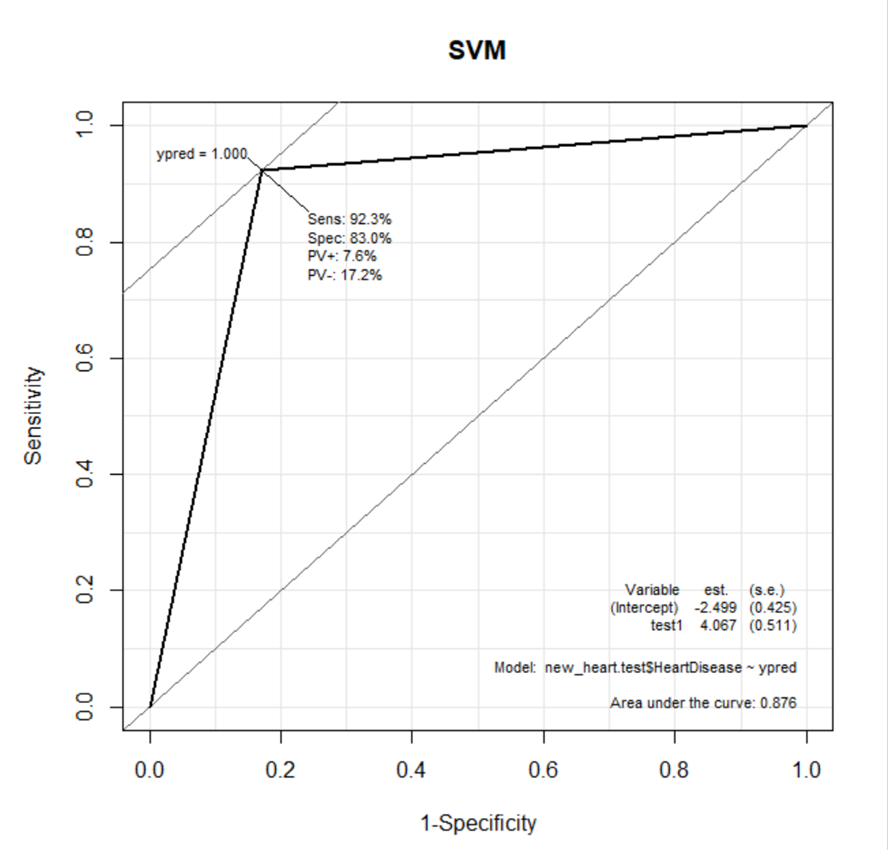


**<그림 29 Support Vector Classifier ROC curve>**

[서포트 벡터 머신]

SVM을 적합하였을 때의 ROC curve 그림이다.

AUC값이 약 0.876이다.



**<그림 30 SVM ROC curve>**

# 4. 결론

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Logistic Regression | Classification Tree | Random Forest | Support Vector Classifier | Support Vector Machine |
| AUC | 0.93 | 0.82 | **0.94** | 0.869 | 0.876 |
| Accuracy | 0.861 | 0.813 | **0.892** | 0.867 | 0.873 |

**<표 1모델 간 AUC, Accuracy 값 비교>**

ROC 커브를 통해 모델간 성능을 비교해 본 결과 Random Forest 모델의 AUC값이 0.94로 가장 성능이 좋은 모델로 판단된다. Accuracy의 경우에도 Random Forest 모델은 약 0.89로 가장 높았다. 그러나 상대적으로 튜닝의 필요가 없고 성능이 가장 낮을 것이라고 예상하던 Logistic Regression의 성능이 상당 부분 튜닝 과정을 거친 Random Forest의 성능과 비슷한 것은 의외의 결과였다.

Logistic Regression 모델이 Random Forest 모델을 제외한 모든 모델과 AUC값이 최대 0.11에서 최소 0.06이 차이나는 것은 예상치 못했던 결과이다. 모델간 성능의 차이가 크지 않다면, 복잡한 튜닝 과정을 거치는 모델들을 사용하는 것이 과연 간단한 과정을 거치는 Logistic Regression 보다 더 나은가에 대해선 더 고민해보고 사용해야 되는 부분이라고 생각한다.

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