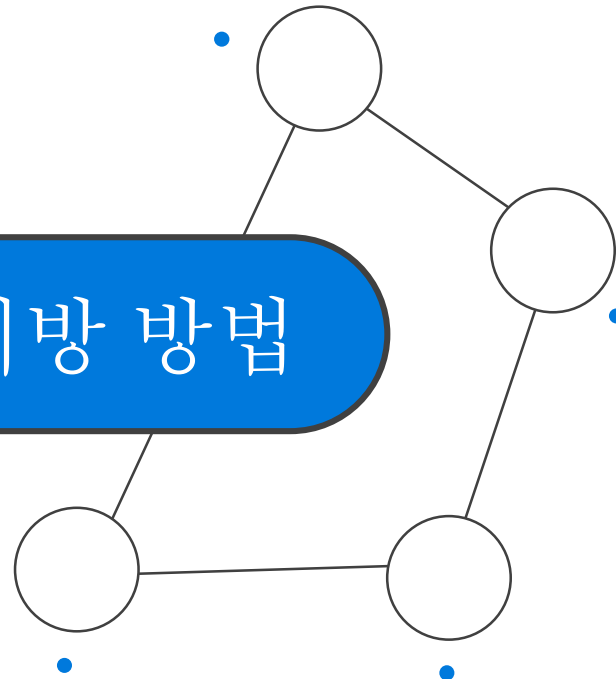


데이터 마이닝

# 유방암 효율적인 예방 방법

강민지 (경영학과, 20161211)



## 주제 선정 이유

- 요즘 유튜브나 sns를 보면 젊은 현대인들이 암에 걸린 경우가 많다.

유방암은 특히 전조 증상이 딱히 없어 예방하기 위해서는 자가진단을 해야한다.

어떤걸 중점으로 자가진단을 해야하는지 알아보기 위해 주제를 선정했다.

# 위스콘신 유방암 진단 데이터

- 위스콘신 대학교에서 제공한 유방암 진단 결과 데이터

```
> str(df)
'data.frame':   569 obs. of  32 variables:
 $ id          : int  87139402 8910251 905520 868871 9012568 906539 925291 87880 862989 89827
 ...
 $ diagnosis   : chr  "B" "B" "B" "B" ...
 $ radius_mean : num  12.3 10.6 11 11.3 15.2 ...
 $ texture_mean : num  12.4 18.9 16.8 13.4 13.2 ...
 $ perimeter_mean : num  78.8 69.3 70.9 73 97.7 ...
 $ area_mean   : num  464 346 373 385 712 ...
 $ smoothness_mean : num  0.1028 0.0969 0.1077 0.1164 0.0796 ...
 $ compactness_mean : num  0.0698 0.1147 0.078 0.1136 0.0693 ...
 $ concavity_mean : num  0.0399 0.0639 0.0305 0.0464 0.0339 ...
 $ points_mean  : num  0.037 0.0264 0.0248 0.048 0.0266 ...
 $ symmetry_mean : num  0.196 0.192 0.171 0.177 0.172 ...
 $ dimension_mean : num  0.0595 0.0649 0.0634 0.0607 0.0554 ...
 $ radius_se    : num  0.236 0.451 0.197 0.338 0.178 ...
 $ texture_se    : num  0.666 1.197 1.387 1.343 0.412 ...
 $ perimeter_se  : num  1.67 3.43 1.34 1.85 1.34 ...
 $ area_se      : num  17.4 27.1 13.5 26.3 17.7 ...
 $ smoothness_se : num  0.00805 0.00747 0.00516 0.01127 0.00501 ...
 $ compactness_se : num  0.0118 0.03581 0.00936 0.03498 0.01485 ...
 $ concavity_se  : num  0.0168 0.0335 0.0106 0.0219 0.0155 ...
 $ points_se     : num  0.01241 0.01365 0.00748 0.01965 0.00915 ...
 $ symmetry_se   : num  0.0192 0.035 0.0172 0.0158 0.0165 ...
 $ dimension_se  : num  0.00225 0.00332 0.0022 0.00344 0.00177 ...
 $ radius_worst  : num  13.5 11.9 12.4 11.9 16.2 ...
 $ texture_worst : num  15.6 22.9 26.4 15.8 15.7 ...
 $ perimeter_worst : num  87 78.3 79.9 76.5 104.5 ...
 $ area_worst    : num  549 425 471 434 819 ...
 $ smoothness_worst : num  0.139 0.121 0.137 0.137 0.113 ...
 $ compactness_worst : num  0.127 0.252 0.148 0.182 0.174 ...
 $ concavity_worst : num  0.1242 0.1916 0.1067 0.0867 0.1362 ...
 $ points_worst  : num  0.0939 0.0793 0.0743 0.0861 0.0818 ...
 $ symmetry_worst : num  0.283 0.294 0.3 0.21 0.249 ...
 $ dimension_worst : num  0.0677 0.0759 0.0788 0.0678 0.0677 ...
```

- \_mean: 평균값
- \_se: 표준오차
- \_worst: 각 세포별 구분  
들에서 제일 큰 3개의  
값 평균

## 변수 설명

Id: 환자 식별 번호

Diagnosis: 양성 여부(M=악성, B: 양성)

각 세포에 대한 정보들

Radius: 반경(중심에서 외벽까지 거리들의 평균값)

Texture: 질감

Perimeter: 둘레

Area: 면적

Smoothness: 매끄러움(반경 길이의 국소적 변화)

Compactness: 조그만 정도 (둘레<sup>2</sup>/면적-1)

Concavity: 오목함(윤곽의 오목한 부분의 정도)

Points: 오목한 점의 수

Symmetry: 대칭

Dimension: 프랙탈 차원

# 위스콘신 유방암 진단 데이터

```
> data.rej %>% str()
'data.frame': 569 obs. of 31 variables:
 $ radius_mean      : num 12.3 10.6 11 11.3 15.2 ...
 $ texture_mean     : num 12.4 18.9 16.8 13.4 13.2 ...
 $ perimeter_mean   : num 78.8 69.3 70.9 73 97.7 ...
 $ area_mean        : num 464 346 373 385 712 ...
 $ smoothness_mean  : num 0.1028 0.0969 0.1077 0.1164 0.0796 ...
 $ compactness_mean : num 0.0698 0.1147 0.078 0.1136 0.0693 ...
 $ concavity_mean   : num 0.0399 0.0639 0.0305 0.0464 0.0339 ...
 $ points_mean      : num 0.037 0.0264 0.0248 0.048 0.0266 ...
 $ symmetry_mean    : num 0.196 0.192 0.171 0.177 0.172 ...
 $ dimension_mean   : num 0.0595 0.0649 0.0634 0.0607 0.0554 ...
 $ radius_se        : num 0.236 0.451 0.197 0.338 0.178 ...
 $ texture_se       : num 0.666 1.197 1.387 1.343 0.412 ...
 $ perimeter_se     : num 1.67 3.43 1.34 1.85 1.34 ...
 $ area_se          : num 17.4 27.1 13.5 26.3 17.7 ...
 $ smoothness_se    : num 0.00805 0.00747 0.00516 0.01127 0.00501 ...
 $ compactness_se   : num 0.0118 0.03581 0.00936 0.03498 0.01485 ...
 $ concavity_se     : num 0.0168 0.0335 0.0106 0.0219 0.0155 ...
 $ points_se        : num 0.01241 0.01365 0.00748 0.01965 0.00915 ...
 $ symmetry_se      : num 0.0192 0.035 0.0172 0.0158 0.0165 ...
 $ dimension_se     : num 0.00225 0.00332 0.0022 0.00344 0.00177 ...
 $ radius_worst     : num 13.5 11.9 12.4 11.9 16.2 ...
 $ texture_worst    : num 15.6 22.9 26.4 15.8 15.7 ...
 $ perimeter_worst  : num 87 78.3 79.9 76.5 104.5 ...
 $ area_worst       : num 549 425 471 434 819 ...
 $ smoothness_worst : num 0.139 0.121 0.137 0.137 0.113 ...
 $ compactness_worst : num 0.127 0.252 0.148 0.182 0.174 ...
 $ concavity_worst  : num 0.1242 0.1916 0.1067 0.0867 0.1362 ...
 $ points_worst     : num 0.0939 0.0793 0.0743 0.0861 0.0818 ...
 $ symmetry_worst   : num 0.283 0.294 0.3 0.21 0.249 ...
 $ dimension_worst  : num 0.0677 0.0759 0.0788 0.0678 0.0677 ...
 $ 구분             : Factor w/ 2 levels "악성","양성": 2 2 2 2 2 2 2 1 2 2 ...
```

## 데이터 전처리

```
#id 변수 제거
data <- df %>% select(-id)
data
#diagnosis 변수 인자 변수로 변환
data$구분 <- factor(ifelse(data$diagnosis=='B','양성','악성'))
data$구분
data.rej <- data[, -1]
```

- id 변수 제거
- M,B로 구분되어 있던 Diagnosis 변수를 양성, 악성으로 구분하는 벡터로 변환
- Diagnosis(양성, 악성 구분) 변수 제거

# 위스콘신 유방암 진단 데이터

```
> data.rej %>% str()
'data.frame':   569 obs. of  31 variables:
 $ radius_mean      : num  12.3 10.6 11 11.3 15.2 ...
 $ texture_mean     : num  12.4 18.9 16.8 13.4 13.2 ...
 $ perimeter_mean   : num  78.8 69.3 70.9 73 97.7 ...
 $ area_mean        : num  464 346 373 385 712 ...
 $ smoothness_mean  : num  0.1028 0.0969 0.1077 0.1164 0.0796 ...
 $ compactness_mean : num  0.0698 0.1147 0.078 0.1136 0.0693 ...
 $ concavity_mean   : num  0.0399 0.0639 0.0305 0.0464 0.0339 ...
 $ points_mean      : num  0.037 0.0264 0.0248 0.048 0.0266 ...
 $ symmetry_mean    : num  0.196 0.192 0.171 0.177 0.172 ...
 $ dimension_mean   : num  0.0595 0.0649 0.0634 0.0607 0.0554 ...
 $ radius_se        : num  0.236 0.451 0.197 0.338 0.178 ...
 $ texture_se       : num  0.666 1.197 1.387 1.343 0.412 ...
 $ perimeter_se     : num  1.67 3.43 1.34 1.85 1.34 ...
 $ area_se         : num  17.4 27.1 13.5 26.3 17.7 ...
 $ smoothness_se    : num  0.00805 0.00747 0.00516 0.01127 0.00501 ...
 $ compactness_se   : num  0.0118 0.03581 0.00936 0.03498 0.01485 ...
 $ concavity_se     : num  0.0168 0.0335 0.0106 0.0219 0.0155 ...
 $ points_se        : num  0.01241 0.01365 0.00748 0.01965 0.00915 ...
 $ symmetry_se      : num  0.0192 0.035 0.0172 0.0158 0.0165 ...
 $ dimension_se     : num  0.00225 0.00332 0.0022 0.00344 0.00177 ...
 $ radius_worst     : num  13.5 11.9 12.4 11.9 16.2 ...
 $ texture_worst    : num  15.6 22.9 26.4 15.8 15.7 ...
 $ perimeter_worst  : num  87 78.3 79.9 76.5 104.5 ...
 $ area_worst       : num  549 425 471 434 819 ...
 $ smoothness_worst : num  0.139 0.121 0.137 0.137 0.113 ...
 $ compactness_worst: num  0.127 0.252 0.148 0.182 0.174 ...
 $ concavity_worst  : num  0.1242 0.1916 0.1067 0.0867 0.1362 ...
 $ points_worst     : num  0.0939 0.0793 0.0743 0.0861 0.0818 ...
 $ symmetry_worst   : num  0.283 0.294 0.3 0.21 0.249 ...
 $ dimension_worst  : num  0.0677 0.0759 0.0788 0.0678 0.0677 ...
 $ 구분             : Factor w/ 2 levels "악성","양성": 2 2 2 2 2 2 2 1 2 2 ...
```

## 데이터 전처리

```
set.seed(1234)

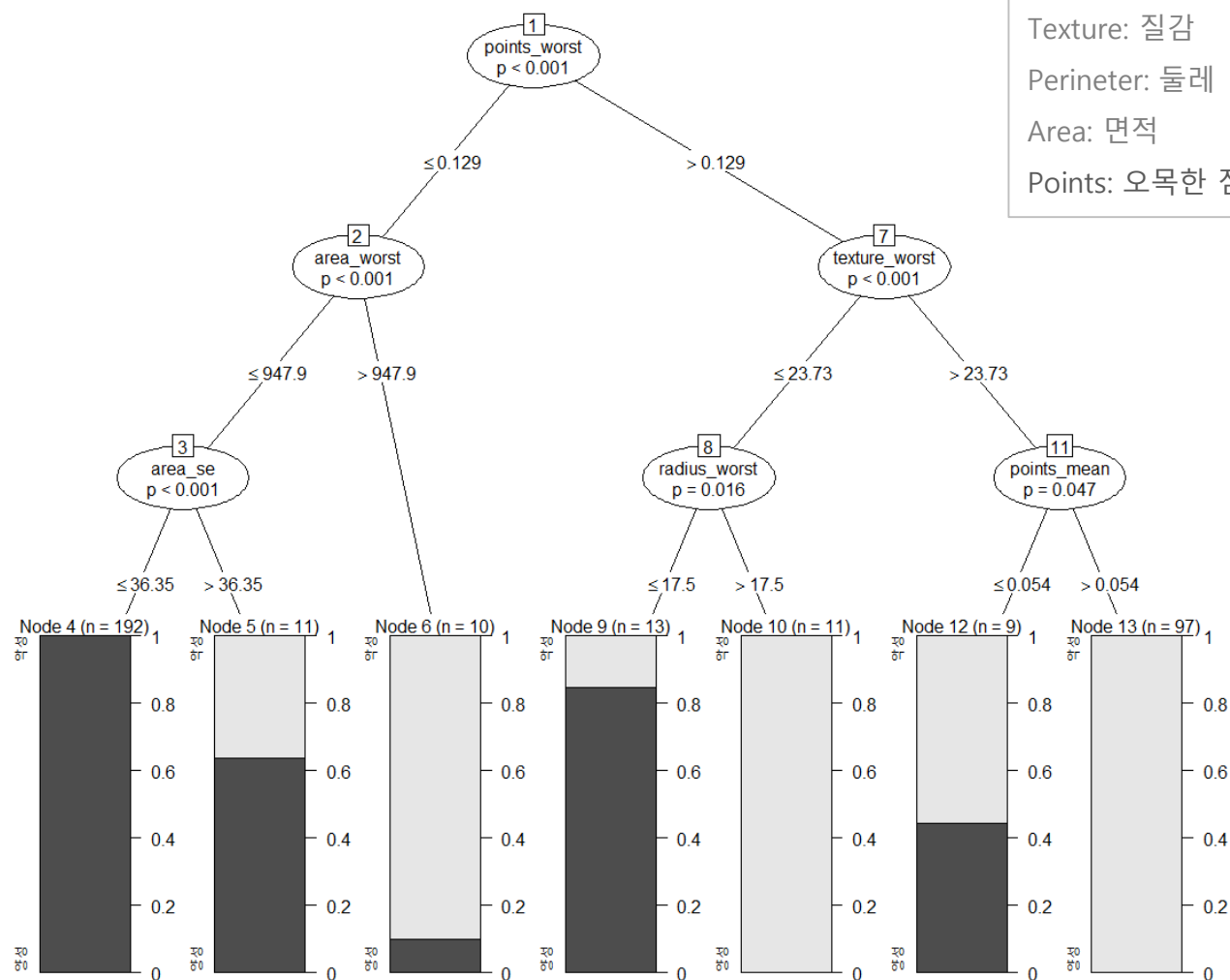
inTrain <- createDataPartition(y = data$구분,
                                p = 0.6,
                                list = FALSE)

# 훈련용과 검정용 자료를 저장
intrain <- data.rej[inTrain,]
intest <- data.rej[!inTrain,]

head(intrain)
summary(intrain)
```

- 샘플 수 맞춰줌
- createDataPartition을 이용하여 훈련 데이터와 테스트 데이터로 분할

# 의사결정나무



## 과정

```
tree <- ctree(구분 ~ ., data= intrain)
```

```
plot(tree)
```

```
testpred <- predict(tree, newdata=intest)
```

```
table(intest$구분, testpred)
```

```
caret::confusionMatrix(reference=intest$구분,  
                        data=testpred, positive="양성")
```

```
roc.curve(intest$구분, testpred)
```

## 해석

- Points(오목한 점의 수)가 0.129 이하이면 거의 양성이지만 area가 947.9 초과이면 악성일 가능성이 높음
- Points(오목한 점의 수)가 0.129 초과이면 질감에 따라 분류
- Texture(질감)이 23.73 초과이면 Points로 분류되지만 point가 작아도 악성일 가능성 높음
- Texture(질감)이 23.73 이하이면 radius(반경)으로 분류

# 의사결정나무

```
> caret::confusionMatrix(reference=intest$구분, data=testpred, positive="양성")  
Confusion Matrix and Statistics
```

| Prediction \ Reference | 악성 | 양성  |
|------------------------|----|-----|
| 악성                     | 78 | 8   |
| 양성                     | 6  | 134 |

Accuracy : 0.9381  
95% CI : (0.8982, 0.9657)

No Information Rate : 0.6283  
P-Value [Acc > NIR] : <2e-16

Kappa : 0.868

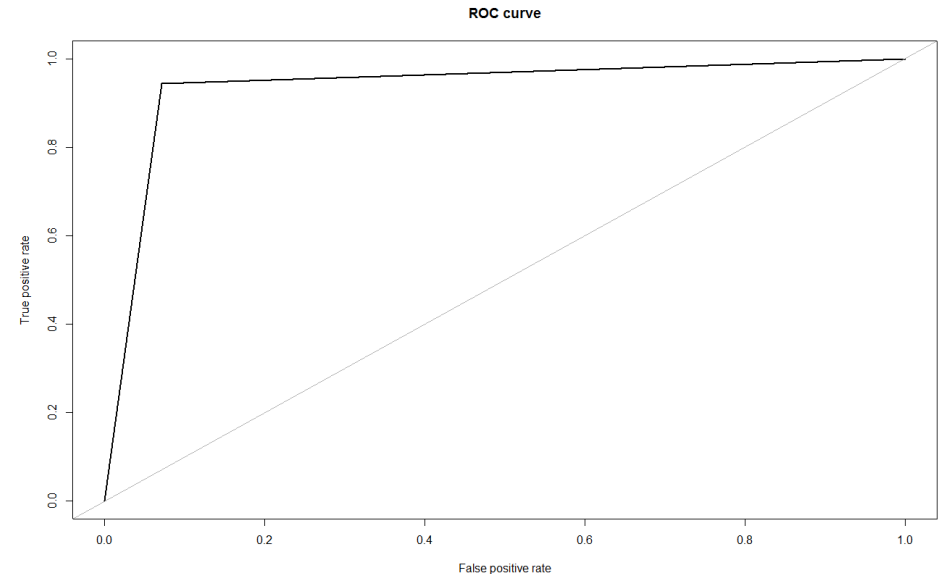
Mcnemar's Test P-Value : 0.7893

Sensitivity : 0.9437  
Specificity : 0.9286  
Pos Pred Value : 0.9571  
Neg Pred Value : 0.9070  
Prevalence : 0.6283  
Detection Rate : 0.5929  
Detection Prevalence : 0.6195  
Balanced Accuracy : 0.9361

'Positive' Class : 양성

- **caret::confusionMatrix**  
**Accuracy로 정확도 측정**
- **Table()함수를 통한 분류 확인**  
**(정오분류표)**

정확도:  $(78+134)/(78+134+6+8) = 0.9380531$



```
> roc.curve(intest$구분, testpred)  
Area under the curve (AUC): 0.936
```

## ROC curve 확인

- AUC가 0.936  
-> 정확도가 높은 편이다.

# SVM(support vector machine)

```
svm.1 <- svm(구분~., data=intrain, type="C-classification",  
             kernel="radial", cost=10, gamma=0.1)  
summary(svm.1)  
  
pred.1 <- predict(svm.1, intest, decision.values = TRUE)  
table(pred.1, intest$구분)
```

```
> table(pred.1, intest$구분)
```

```
pred.1 악성 양성  
악성    79     5  
양성     5   137
```

- Predict() 함수를 통한 예측 수행
- Table() 함수를 통한 분류 확인
- 정확도:  $(79+137)/(79+137+5+5) = 0.9557522$

```
> tuned <- tune.svm(구분~., data=intrain,  
+                  gamma = 10^(-6:-1), cost = 10^(1:2))  
> summary(tuned)
```

Parameter tuning of 'svm':

- sampling method: 10-fold cross validation

- best parameters:

```
gamma cost  
0.001    10
```

- best performance: 0.02327731

- Detailed performance results:

|    | gamma | cost | error      | dispersion |
|----|-------|------|------------|------------|
| 1  | 1e-06 | 10   | 0.37327731 | 0.06609552 |
| 2  | 1e-05 | 10   | 0.31226891 | 0.07865035 |
| 3  | 1e-04 | 10   | 0.05840336 | 0.03373054 |
| 4  | 1e-03 | 10   | 0.02327731 | 0.03318128 |
| 5  | 1e-02 | 10   | 0.02613445 | 0.02507417 |
| 6  | 1e-01 | 10   | 0.04361345 | 0.02467674 |
| 7  | 1e-06 | 100  | 0.31226891 | 0.07865035 |
| 8  | 1e-05 | 100  | 0.05840336 | 0.03373054 |
| 9  | 1e-04 | 100  | 0.02327731 | 0.03318128 |
| 10 | 1e-03 | 100  | 0.03201681 | 0.03189946 |
| 11 | 1e-02 | 100  | 0.03487395 | 0.02632234 |
| 12 | 1e-01 | 100  | 0.04361345 | 0.02467674 |

## Tune() 함수를 통한 초모수 조율

- 최적의 모수: Gamma: 0.001, cost: 10



# SVM(support vector machine)

```
#최적의 모수 SVM
svm.2 <- svm(구분~, data=intrain, type="C-classification",
             kernel="radial", cost=10, gamma=0.001)
summary(svm.2)

pred.2 <- predict(svm.2, intest, decision.values = TRUE)
table(pred.2, intest$구분)
```

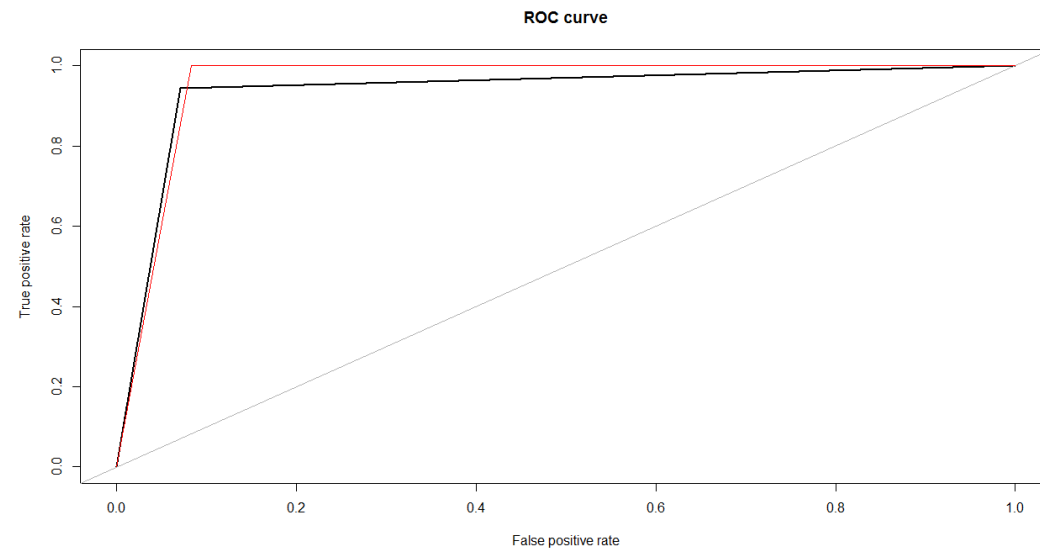
```
> table(pred.2, intest$구분)
```

```
pred.2 악성 양성
악성    77     0
양성     7   142
```

최적의 모수를 가지고 SVM 수행

Table() 함수를 통한 분류 확인

정확도:  $(77+142)/(77+142+7) = 0.9690265$



```
> roc.curve(intest$구분, pred.2, add.roc = T, col="red")
Area under the curve (AUC): 0.958
```

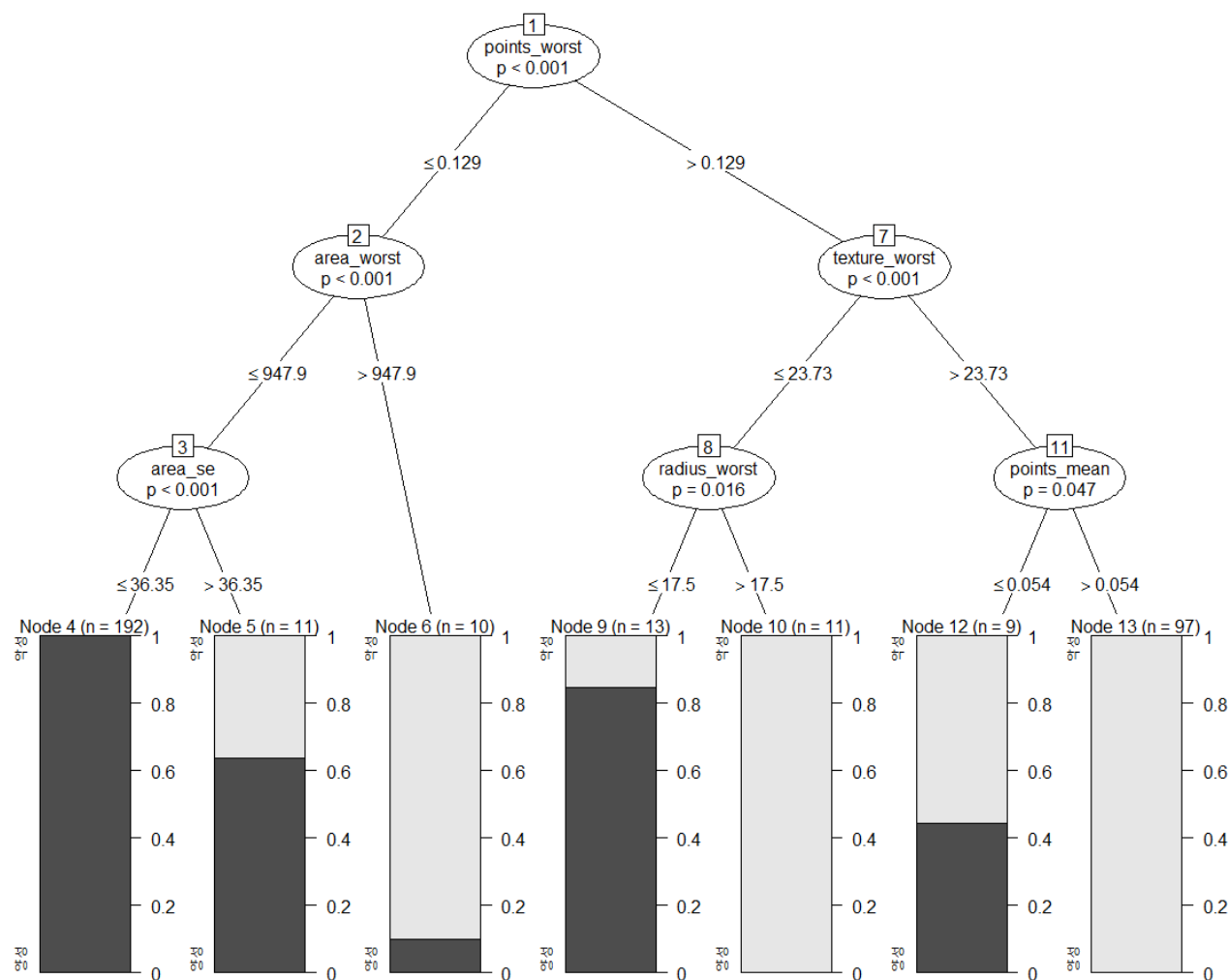
ROC curve 확인

- AUC가 0.958

ROC curve 비교

- SVM 이 의사결정나무보다 정확도가 높음  
-> SVM모형이 더 우수

# 결론



Radius: 반경  
Texture: 질감  
Perimeter: 둘레  
Area: 면적  
Point: 오목한 점의 수

의사결정나무 모형을 보았을 때  
낭종 수, 질감, 둘레, 반경, 면적  
위주로 한번씩 자가진단을 하는 것이  
예방하는, 초기에 치료받을 수 있는 방법이  
라고 생각한다.

## 출처

<https://gomguard.tistory.com/52>

<https://m.blog.naver.com/audgnsdl115/221516274600>

<https://kuklife.tistory.com/53>