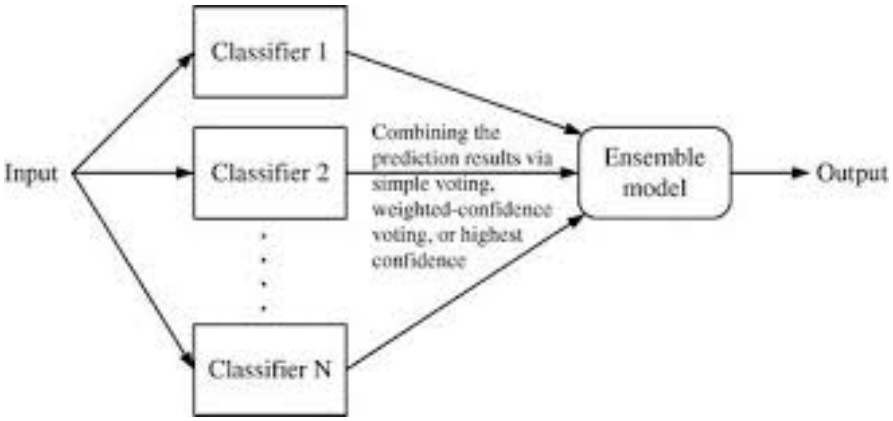
Ensemble



주어진 자료로부터 여러 개의 예측 모형을 만든 후 예측 모형들을 종합하여 하나의 최종 예측 모형을 만드는 방법.

Ex) 오분류율이 5%인 분류기 5개가 있을 때, 해당 모형들이 모두 동일한 결정을 내린다면 모형의 오분류율은 5%지만 각각의 분류기가 상호독립적이어서 전체 분류기들의 절반 이상이 오분류 할 할 때의 오분류율을 측정한다면 0.01%로 떨어진다.

$$E = \sum_{i=3}^{5} (0.05)^i (1 - 0.05)^{5-i} = 0.0001$$

Ensemble 장점

Bias를 감소시킨다: 치우침이 있는 여러 모형의 평균을 취하면, 어느 쪽에도 치우치지 않는 결과(평균)를 얻게 된다.

Variance를 감소시킨다: 한 개 모형으로부터의 단일 의견보다 여러 모형의 의견을 결합하면 분산이 작아진다.

Overfitting(과적합)의 가능성을 줄여준다: 과적합이 없는 각 모형으로부터 예측을 결합(평균, 가중평균, 로지스틱 회귀)하면 과적합의 여지가 줄어든다

Ensemble 단점

해석에 어려움을 가진다.

다른 단일 모델에 비해 예측 시간이 많이 걸린다.

모형의 투명성이 떨어지게 되어 현상에 대한 원인을 분석할 때는 적합하지 않다.

종류

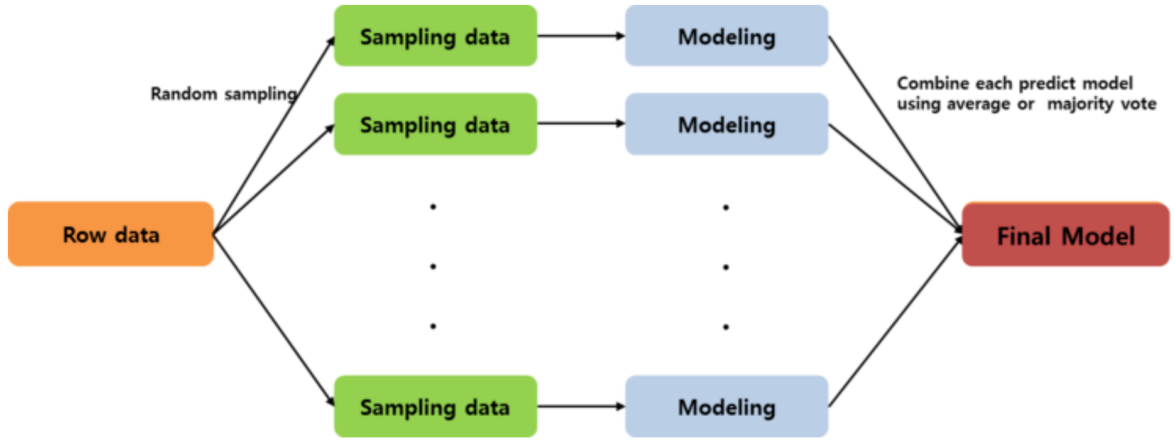
데이터를 조정하는 방법 : Bagging, Boosting

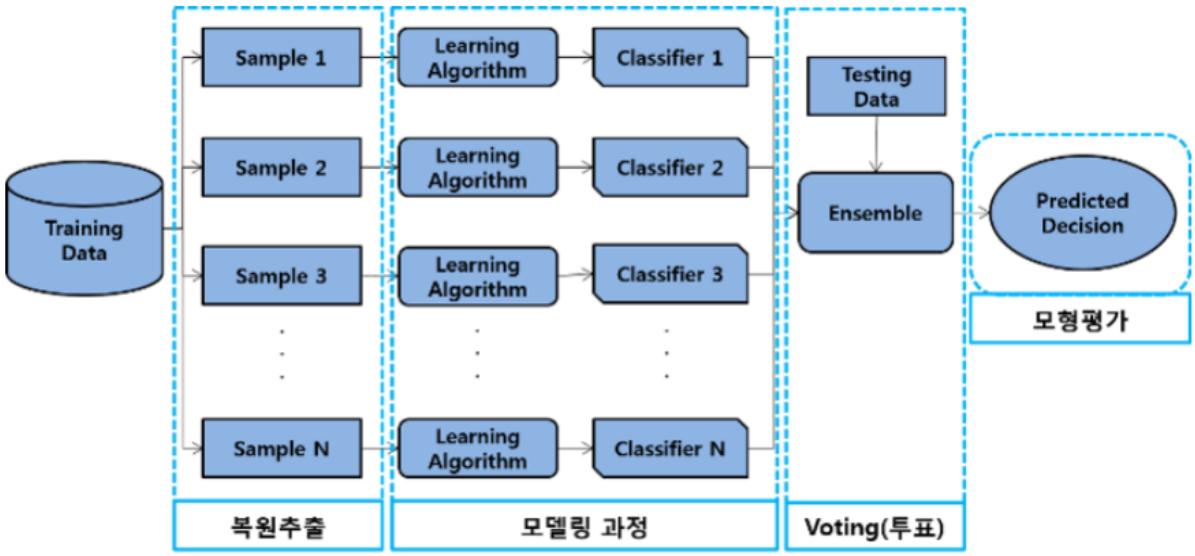
변수의 개수를 조정하는 방법 : Random Forest

집단명을 조정하는 방법

분류모형의 가정을 조정하는 방법 : Stacking

Bagging





Bootstrap + Aggregating의 줄임말

Bootstrap aggregating의 줄임말로 원 데이터 집합으로부터 크기가 같은 표본을 여러 번 단순임의 복원추출하여 각 표본(이를 bootstrap 표본이라 함)에 대해 분류기(classifiers)를 생성한 후 그 결과를 결합(앙상블)하는 방법이다.

주로 decision tree에 대해 많이 쓰이지만 어떤 알고리즘에 관해서도 사용 가능하다. 최적의 decision tree를 구축할 때 가장 어려운 부분이 가지치기(pruning)이지만 bagging에선 가지치기를 하지 않고 최대로 성장한 decision tree를 활용한다. 그래서 각 tree의 분산(variance)은 크고 편향(bias)은 작다. 이러한 tree들을 합침으로서 분산은 줄어들게 된다.

$\mathrm{Z}_1, \ldots, \mathrm{Z}_\mathrm{n}$ 의 n개의 독립적인 관측치가 있을 때 각각의 분산이 σ^2이라 하면 $\bar{Z}$의 분산은 σ^2/n이 되는것과 비슷한 원리

Bagging에서의 Bootstrapping

n개의 원 데이터(Raw data, training data)로부터 B(bag의 개수)번의 랜덤복원추출을 하고 각 샘플의 모델링을 통해 나온 예측변수들을 결합하여 최종 모델을 생성하는 것이다.

n'(bag 안에 포함된 데이터 개수) = n일 때 고유한(중복되지 않은) 데이터의 비율은 보통 n의 63.2%

데이터가 충분히 큰 경우 어떤 데이터가 하나의 bootstrap 표본에서 제외될 확률은 다음과 같다.

$$\lim_{\mathrm{n}\to\infty}\left(1-\frac{1}{\mathrm{n}}\right)^{\mathrm{n}} = e^{-1} = 0.3678$$

$1-\mathrm{e}^{-1}$: 어떤 데이터가 하나의 bootstrap 표본으로 뽑힐 확률. n개 뽑을 때 뽑히지 않은 것의 개수만큼 중복된 데이터 존재.

OOB:

Bootstrap 표본으로 뽑히지 않는 데이터를 OOB(Out-of-Bag) 데이터라고 한다. 따로 test data를 구성할 필요 없이 OOB 데이터를 이용해서 test error를 구하거나 변수의 중요도를 측정할 수 있다.

한 관측값에 대한 test error: OOB가 될 확률이 약 1/3이므로 약 B/3개의 모델들을 사용하게 됨. 해당 관측값이 OOB로 분류된 모델들에 대해 인풋으로 넣은 후 각 결과들을 평균내면 된다. Test에 이용되는 관측치로 학습하지 않았으므로 test 결과는 유효하다. 만약 데이터 개수가 충분히 많다면 OOB error는 Leave One Out Cross Validation error와 같게 된다.

Aggregating:

예측변수들을 결합하는 방법은 변수가 연속형일 경우에는 평균, 범주형일 경우에는 다중투표(majority vote)를 사용하는 것이 일반적이다.

Bagging 장점

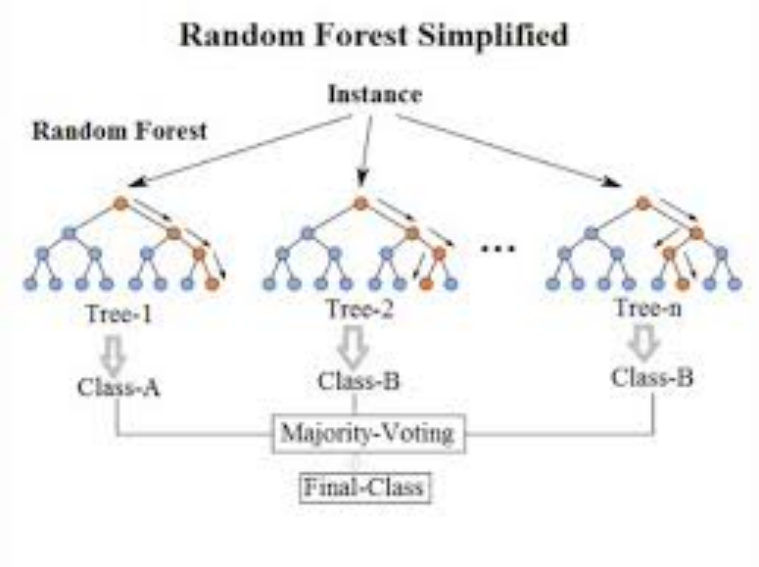
평균예측모형의 기대손실이 단일예측모형의 기대손실보다 항상 작다

Variance를 줄여줌으로써 예측력을 향상시켜 과적합을 방지한다.

Bagging 단점

tree에서 활용되는 독립변수 선정에 대한 고민 없이 무작정 Decision Tree를 양산하기 때문에 tree 간의 높은 상관성이 생기게 된다. (=> 이 문제점을 해결하는 것이 Random Forest)

Random Forest



2001년에 Leo Breiman에 의해 처음으로 소개된 기법으로 Decision tree의 단점을 개선하기위한 알고리즘 중 가장 지배적인 알고리즘 중 하나이다. 다수의 Decision tree를 결합하여 하나의 모형을 생성하는 방법.

Bagging과 Random Subspace method 개념을 결합해 고안한 기법이 Random Forest이다. Bagging과의 차이는 임의성(randomness)를 관측치(observation, instance)뿐만 아니라 변수(feature, variable)에도 적용했다는 것이다. 변수도 임의로 추출하기 때문에(Random Subspace) 변수간 상관성이 높은 변수가 섞일 가능성이 낮아져 성능이 좋아진다.

Random Subspace method

Feature Bagging이라고도 부른다.

기계학습에서 변수를 전부 이용하는 것 대신 몇 가지를 임의 추출하여 이용하는 것을 의미한다.

변수를 임의 추출하는 이유는 한 개 또는 극소수의 변수들이 강한 예측력을 가진다면 훈련 과정에서 여러 tree들에 그 변수들이 중복되어 선택되고 결과적으로 tree들이 상관화되기 때문에 이를 방지하기 위함이다.

전체 변수 개수가 p개이고 선택 변수의 개수가 m개라 하면

회귀 tree의 경우는 m=p/3, 분류 tree의 경우는 m=$\sqrt{p}$ 로 정하는 것이 좋은 예측력을 보인다고 알려져 있다. m=p일 경우는 Bagging과 같다.

Random forest 장점

다양성 극대화하여 예측력이 높고 다수 tree의 예측 결과를 종합하여 안정성도 우수하다.

Random forest 단점

다수 tree를 이용해 의사결정을 내리므로 기존의 tree가 갖는 장점인 설명력이 떨어진다.

변수 중요도 측정

Bagging과 Random forest는 tree 하나보단 예측정확성이 높다. 그러나 결과 모델을 해석하기 어렵다. 변수 중요도를 설명하기 쉽다는 Tree의 장점이 사라진 것이다. 많은 수의 tree를 이용했기 때문에 어떤 변수가 예측과정에서 얼마나 중요한지 정확히 알 수 없다. Tree 하나보단 해석하기 어렵지만 각 독립변수들의 전체적인 중요도를 RSS(Residual Sum of Squares)나 Gini계수를 통해 알 수 있다. 회귀 tree bagging의 경우 RSS를 사용하고, 분류 tree bagging의 경우 Gini계수를 사용한다. 특정인자 값을 재조합하고 재조합 전후의 OOB error를 비교하는 방법(Permutation importance)도 있다.

RSS(Residual Sum of Squares)

$$\sum_{j=1}^{J}\sum_{i\in R_j}(y_i-\hat{y}_{R_j})^2$$

X변수에 대해 J개의 겹치지 않은 영역을 만들고 각 영역의 이름을 R_j라고 한다. 각 R_j에 대한 평균 y값을 $\widehat{y_{R_j}}$ 이라 한다. 실제 y값들과 $\widehat{y_{R_j}}$의 차이의 제곱합을 RSS라고 한다. B개의 tree에 대해 각 변수에서의 split으로 인해 RSS가 감소된 정도를 측정하고 평균을 낸다. 해당 변수로 인해 RSS가 많이 감소하였다면 이는 중요한 변수임을 의미한다.

Gini계수

$$G=\sum_{k=1}^{K}\hat{p}_{mk}(1-\hat{p}_{mk})$$

$\widehat{p_{mk}}$: m영역에 속하는 레코드 중 k범주에 속하는 레코드의 비율

Gini계수가 0에 가까울수록 불균등 분배(낮은 불순도), 1에 가까울수록 균등 분배(높은 불순도)를 의미한다. 역시 B개의 tree에 대해 각 변수에서 split으로 인해 Gini계수가 감소된 정도를 측정하고 평균을 낸다. 해당 변수로 인해 Gini 계수가 많이 감소하였다면 이는 중요한 변수임을 의미한다.

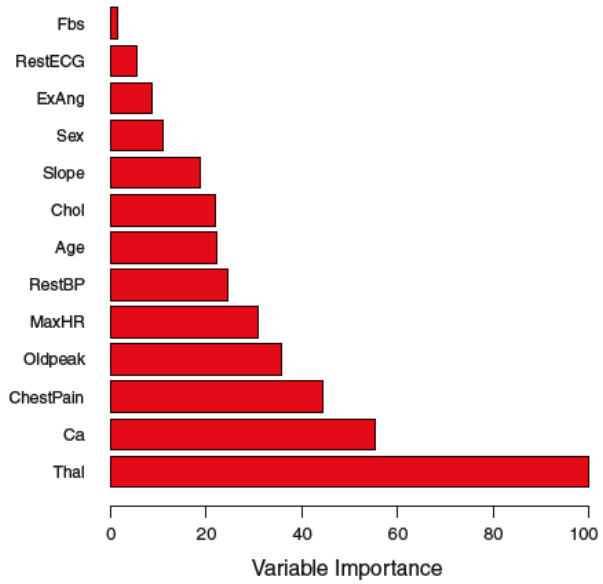
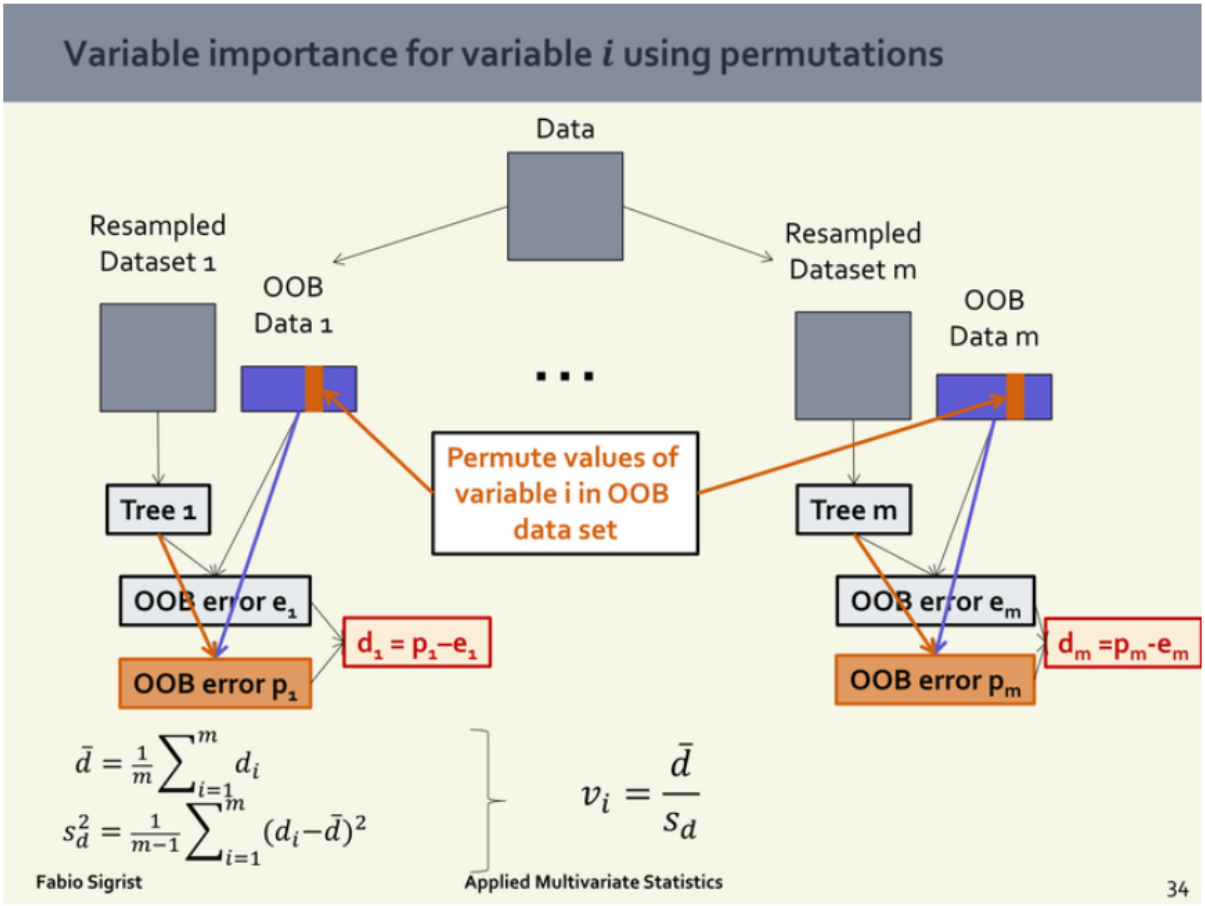


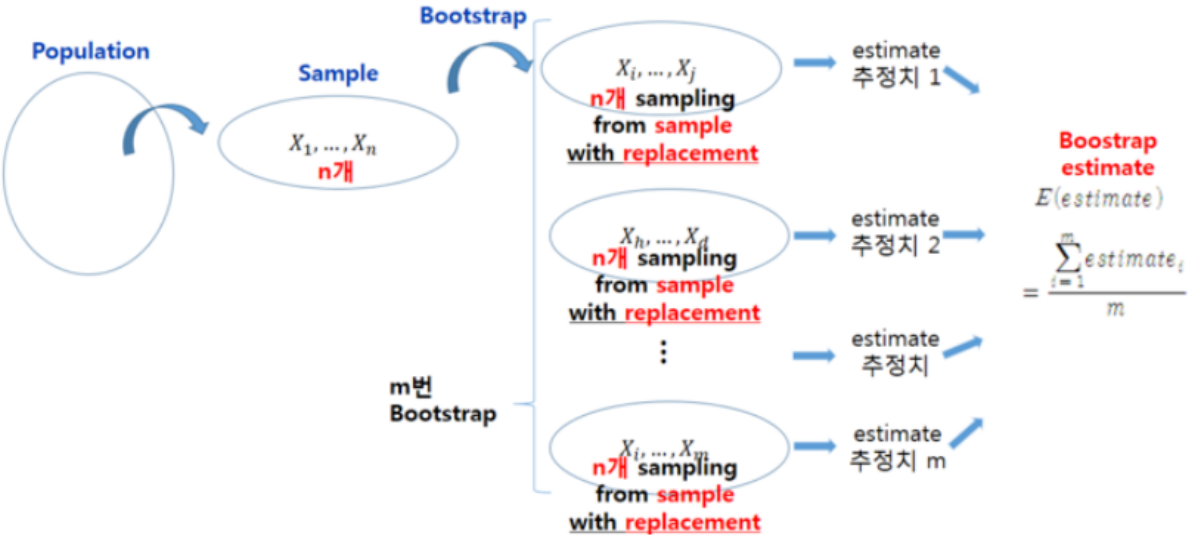
FIGURE 8.9. *A variable importance plot for the* `Heart` *data. Variable importance is computed using the mean decrease in Gini index, and expressed relative to the maximum.*

Permutation importance



각 B개 tree에 대해 OOB 데이터를 이용해 오분류율(error)을 구한다. OOB 데이터의 특정 변수를 선택한 후 그 변수의 값들을 재조합(permutation)한다. 간단히 말해 변수값들의 위치를 바꾼다는 의미이다. 그 후에 OOB error를 다시 구하고 원래 OOB error와의 차이를 구한다. 만약 그 변수가 중요하지 않은 변수라면 OOB error값의 차이가 크지 않을 것이다. 이미 학습된 모델을 사용하므로 재조합 후의 데이터로 다시 학습시킬 필요가 없다. Permutation을 이용해 변수 중요도를 측정하는 방법은 간단하고, 신뢰할 수 있어서 다른 기법에서도 널리 쓰인다고 한다.

Bootstrapping



주어진 데이터셋(표본)을 원래의 모집단을 대표하는 독립 표본으로 가정하고, 그 자료로부터 중복을 허용한 무작위 재추출로 복수의 자료를 작성하고 각각에서 얻어진 통계량을 계산하는 방법. 표본으로 모집단을 추정하는 것처럼(sampled data -> population) 표본에서 다시 재추출한 표본(resampled data -> sampled data)으로 표본을 추정하는 아이디어이다. 재추출 횟수는 보통 1000 번 이상이다.

Bootstrapping 장점

Bootstrapping 을 이용하여 어떤 값을 추정할 때 단순 평균 추정보다 bias 및 variance 가 작다

기존의 통계 방법들은 대부분 증명하기 힘든 가정(정규성 가정 등등..)들을 전제로 하지만 bootstrapping 은 그러한 가정이 없어도 사용 가능하다.

참고자료 및 출처

http://euriion.com/?p=412193

https://adnoctum.tistory.com/296

https://www.kdnuggets.com/2017/11/difference-bagging-boosting.html bagging, boosting 차이

https://swalloow.github.io/bagging-boosting bagging 장점, 이미지

http://blog.naver.com/PostView.nhn?blogId=gksshdk8003&logNo=220898804741&parentCategoryNo=&categoryNo=&viewDate=&isShowPopularPosts=false&from=postView bagging 이미지2 앙상블 종류

http://blog.naver.com/PostView.nhn?blogId=gksshdk8003&logNo=220914969026&parentCategoryNo=&categoryNo=&viewDate=&isShowPopularPosts=false&from=postView RF

http://blog.naver.com/laonple/220838501228 Bagging 적용하면 안되는 경우

https://rpago.tistory.com/56 bagging rss 변수중요도

https://godongyoung.github.io/%EB%A8%B8%EC%8B%A0%EB%9F%AC%EB%8B%9D/2018/02/23/ISL-Tree-Based-Methods_ch8.html bagging rss

https://m.blog.naver.com/wjddudwo209/220013117867 Bootstrapping

https://learningcarrot.wordpress.com/2015/11/12/%EB%B6%80%ED%8A%B8%EC%8A%A4%ED%8A%B8%EB%9E%A9%EC%97%90-%EB%8C%80%ED%95%98%EC%97%AC-bootstrapping/

https://en.wikipedia.org/wiki/Bootstrapping_(statistics)

An Introduction to Statistical Learning with Applications with R 책

머신러닝 기법을 이용한 대졸자 취업 예측 모형(최필선, 민인선) permutation importance

http://blog.naver.com/PostView.nhn?blogId=sw4r&logNo=221032295777 permutation importance 이미지

https://explained.ai/rf-importance/index.html permutation importance