What is Bagging?

Reference

- Bagging Predictors, Leo Breiman

논문의 제목은 Bagging Predictors이다. 여기서 Predictors는 ML에서 예측 변수가 아니라, test data의 y를 생산하는 모델을 의미한다. Bagging은 무엇을 의미할까? 이는 **B**ootstrap **Agg**regat**ing**을 의미한다. 즉, Bootstrap samples들을 Aggregating 했다는 뜻이다. Bootstrap은 n개의 훈련 데이터가 있을 때, 복원추출을 통해서 새로운 데이터 세트를 만드는 기법이다. 새롭게 뽑은 데이터를 Aggregate 한다는 것인데, 구체적으로 어떻게 합치는건지 알아보자. 먼저 계속 사용하게될 notation을 정리하자.

- $\mathcal{L}$은 learning set, 즉 training data이고 $\{(y_n, \boldsymbol{x}_n),\ n = 1, \cdots, N\}$로 구성된다.
- 훈련 데이터 $\mathcal{L}$을 이용하여 predictor인 $\varphi(\boldsymbol{x}, \mathcal{L})$을 만든다. 앞서 언급했듯이, predictor는 예측 변수가 아니라 예측 모형이다. 따라서 $\boldsymbol{x}$는 훈련 도중에 사용되지 않은 데이터, 즉 테스트 데이터이고 이에 대응되는 unknown label인 y를 $\varphi(\boldsymbol{x}, \mathcal{L})$로 예측한다.
- $\{\mathcal{L}_k\}$는 $\mathcal{L}$과 동일한 underlying distribtion에서 뽑은 N개의 독립 관측치로 구성된 또 다른 훈련 데이터이다.
- 목표는 $\mathcal{L}$ 하나만이 아니라, $\{\mathcal{L}_k\}$을 이용하여 $\varphi(\boldsymbol{x}, \mathcal{L}_k)$을 만들고 이를 합쳐서 더 나은 predictor $\varphi(\boldsymbol{x}, \mathcal{L})$을 만드는 것이다.
- y가 numeric이라면 $\varphi(\boldsymbol{x}, \mathcal{L})$은 $\varphi(\boldsymbol{x}, \mathcal{L}_k)$의 평균으로 대체한다. 즉, $\varphi_A(\boldsymbol{x}) = E_{\mathcal{L}}\varphi(\boldsymbol{x}, \mathcal{L})$이고, $E_{\mathcal{L}}$은 $\mathcal{L}$에 대한 기댓값, 즉 $\mathcal{L}$의 가능한 모든 값에 대하여 확률을 부여하여 기댓값을 취한 것이다. 또한 φ_A의 A는 aggregation을 의미한다.
- $\varphi(\boldsymbol{x}, \mathcal{L})$이 class $j \in \{1, \cdots, J\}$을 예측한다면 voting으로 aggregation을 한다. 즉, $N_j = \#\{k; \varphi(\boldsymbol{x}, \mathcal{L}_k) = j\}$이면 $\varphi_A(\boldsymbol{x}) = argmax_j N_j$이다. 즉, 다수결 투표를 해서 가장 많은 표를 얻은 class로 $\boldsymbol{x}$의 class를 배정하는 것이다.

보통, $\mathcal{L}_k$과 같이, 데이터가 여러개로 주어지지 않는다. 따라서 $\mathcal{L}$로부터 bootstrap samples, $\{\mathcal{L}^{(B)}\}$을 취한다. 그리고 이를 이용하여 여러개의 predictors인 $\{\varphi(\boldsymbol{x}, \mathcal{L}^{(B)}\}$을 만든다.

Bagging이 정확도를 향상시키는지의 여부는 φ를 만드는 과정의 안정성에 주요하게 기인한다. $\mathcal{L}$로부터 얻은 bootstrap samples이 φ간에 큰 차이를 만들지 않는다면 φ와 φ_B는 비슷할 것이다. bootstrap samples로 만든 φ들의 차이가 크다면, 정확도에서 향상이 있을 것이다. 그 이유를 살펴보도록 하자.

Why Bagging Works?

훈련 데이터 $\mathcal{L}$의 각 케이스 $(y, \boldsymbol{x})$가 확률 분포 P에서 독립적으로 뽑혔다고 가정하자. y는 numeric이고 $\varphi(\boldsymbol{x}, \mathcal{L})$은 predictor이다. aggregated predictor는

$$\varphi_A(\boldsymbol{x}, P) = E_{\mathcal{L}}\varphi(\boldsymbol{x}, \mathcal{L}) \tag{1}$$

(1)의 기댓값이 의미하는 바를 생각해보면 다음과 같다. 훈련 데이터 $\mathcal{L}$을 통해서 predictor $\varphi(\boldsymbol{x}, \mathcal{L})$을 만든다. 그런데 우리가 사용할 훈련 데이터는 bootstrap samples을 이용하여 만들었으므로 여러개가 있을 것이다. 따라서 이들의 평균을 기댓값의 개념을 이용하여 구하는 것이다.
$Y, \boldsymbol{X}$를 확률 분포 P를 가지고 $\mathcal{L}$과 독립이라고 가정하자. $\varphi(\boldsymbol{x}, \mathcal{L})$의 average prediction error e는

$$e = E_{\mathcal{L}}E_{Y,\boldsymbol{X}}(Y - \varphi(\boldsymbol{X}, \mathcal{L}))^2 \tag{2}$$

$E_{Y,\boldsymbol{X}}(Y - \varphi(\boldsymbol{X}, \mathcal{L}))^2$는 Y와 predictor를 이용해서 만든 $\varphi(\boldsymbol{X}, \mathcal{L})$과의 차이에 대한 평균이다.
aggregated predictor φ_A의 error는

$$e_A = E_{Y,\boldsymbol{X}}(Y - \varphi_A(\boldsymbol{X}, P))^2 \tag{3}$$

(3)의 기댓값이 의미하는 바를 생각해보면 다음과 같다. $\varphi_A(\boldsymbol{X}, P)$는 (1)에서 정의된 aggregated predictor인데, 이를 사용해서 Y값을 예측한 값과 실제 Y 값과의 차이에 대한 평균을 기댓값의 개념을 사용하여 구하는 것이다. Jensen's Inequality에 의해서

$$(EZ)^2 \leq EZ^2 \tag{4}$$

따라서 (2)와 (3)의 관계를 아래와 같이 유도한다.

$$\begin{aligned}
e &= E_{\mathcal{L}}E_{Y,\boldsymbol{X}}(Y - \varphi(\boldsymbol{X}, \mathcal{L}))^2 \\
&= E_{\mathcal{L}}E_{Y,\boldsymbol{X}}Y^2 - 2E_{\mathcal{L}}E_{Y,\boldsymbol{X}}\left(Y\varphi(\boldsymbol{X}, \mathcal{L})\right) + E_{\mathcal{L}}E_{Y,\boldsymbol{X}}\varphi^2(\boldsymbol{X}, \mathcal{L}) \\
&= E_{\mathcal{L}}E_{Y,\boldsymbol{X}}Y^2 - 2E_{\mathcal{L}}\varphi(\boldsymbol{X}, \mathcal{L})E_{Y,\boldsymbol{X}}Y + E_{\mathcal{L}}E_{Y,\boldsymbol{X}}\varphi^2(\boldsymbol{X}, \mathcal{L}) \\
&= E_{\mathcal{L}}E_{Y,\boldsymbol{X}}Y^2 - 2\varphi_A E_{Y,\boldsymbol{X}}Y + E_{Y,\boldsymbol{X}}E_{\mathcal{L}}\varphi^2(\boldsymbol{X}, \mathcal{L}) \ \because (1) \\
&\geq E_{\mathcal{L}}E_{Y,\boldsymbol{X}}Y^2 - 2\varphi_A E_{Y,\boldsymbol{X}}Y + E_{Y,\boldsymbol{X}}\left(E_{\mathcal{L}}\varphi(\boldsymbol{X}, \mathcal{L})\right)^2 \ \because (4) \\
&= E_{Y,\boldsymbol{X}}Y^2 - 2\varphi_A E_{Y,\boldsymbol{X}}Y + E_{Y,\boldsymbol{X}}\left(\varphi_A\right)^2 \ \because (1) \\
&= E_{Y,\boldsymbol{X}}(Y - \varphi_A)^2 = e_A
\end{aligned} \tag{5}$$

따라서 아주 general하게, φ_A의 mean squared prediction이 φ보다 낮음을 알 수 있다. 얼마나 차이가 나는지는Jensen's inequality을 이용하여 inequality가 발생하는 지점인

$$\left(E_{\mathcal{L}}\varphi(\boldsymbol{X}, \mathcal{L})\right)^2 \leq E_{\mathcal{L}}\varphi^2(\boldsymbol{X}, \mathcal{L}) \tag{5}$$

에 달려있다. (5)는 자세히 살펴보면, $\varphi(\boldsymbol{X}, \mathcal{L})$의 분산임을 알 수 있는데, 따라서 $\varphi(\boldsymbol{X}, \mathcal{L})$가 얼마나 variable한지에 따라서 inequality의 차이가 심해질 것이다. 즉, bootstrap에서 뽑는 훈련 데이터 $\mathcal{L}$가 φ에 심한 변동을 일으키면, bagging의 효과가 커짐을 알 수 있다.

bagging을 이용한 predictor는 (1)에 나와있다. 하지만 이는 기댓값이고, 분포를 모른다면 구할 수 없다. 실제로는, bootstrap approximation을 사용한다.

$$\varphi_B(\boldsymbol{x}) = \varphi_A(\boldsymbol{x}, P_{\mathcal{L}}) \tag{6}$$

$P_{\mathcal{L}}$은 P와는 다른데, 훈련 데이터 $(y_n, \boldsymbol{x}_n) \in \mathcal{L}$의 각 점에서 mass $1/N$을 가지는 분포이다. 이를 bootstrap approximation to P라고 부른다. (모분포 대신 사용하는, empirical 분포인 것 같다.) φ_B도 φ_A와 비슷하게, 변동이 크다면 aggregation을 통해서 정확도를 향상시킬 수 있고 stable하다면 거의 비슷한 결과를 낸다.
여태까지는 y가 numeric인 경우를 살펴보았다. regression이 아니라 classification에서도 위와 동일한 결과를 논문에서 도출한다. 유도 과정은 생략!

Simulataion Structure

데이터를 generate하여, bagging의 효과를 알아본다. 데이터는 아래와 같이 생성하였다.

- 데이터는 $y = \sum_m \beta_m x_m + \epsilon$, $\epsilon \sim N(0,1)$ 모델에서 생성됐다.
- 변수의 갯수는 $M = 30$이고 표본의 크기는 $n = 60$이다.
- x_m는 평균이 0인 joint normal distribution에서 뽑혔는데, correlation rnwhsms $EX_iX_j = \rho^{|i-j|}$이고 ρ는 $unif(0,1)$에서 랜덤하게 생성하였다.
- 결과를 best subset selection과 비교를 해본다.

subset selection은 0에 가깝게 작은 β_m가 많을 때, 성능이 좋지 않다고 알려져 있다. spectrum을 bridge하기 위해, 세 종류의 계수가 사용되었다. 계수는 세 개의 클러스터를 형성한다. $m = 5, 15, 25$가 중심이다. 각 클러스터는 아래와 같은 형태의 계수를 취한다.

$$\beta_m = c\left[(h - |m-k|)^+\right],\ m = 1, \cdots, 30$$

여기서 k는 클러스터 센터 (5,15,25)이고 각 클러스터에 대해서는 $h = 1, 3, 5$이다.
각 계수 세트에 대해서 아래 절차가 250번 반복되었다.

1. 훈련 데이터 $\mathcal{L} = \{(y_n, \boldsymbol{x}_n),\ n = 1, \cdots\}$가 $y = \sum \beta_m x_m + \epsilon$에서 뽑힌다. x_m의 correlation 구조는 위에서 언급하였다.

2. $\mathcal{L}$을 이용하여 예측 모형 $\varphi_1(\boldsymbol{x}), \cdots, \varphi_M(\boldsymbol{x})$을 얻기 위해 forward entry of variables을 수행한다. 여기서 $\varphi_m(\boldsymbol{x})$는 M개의 변수 중 m개의 변수만 사용하는 모형이다. 각각의 mean-squared prediction error를 $e_1, \cdots, e_M$로 저장한다.

3. $\mathcal{L}$로부터 50개의 bootstrap replicates 세트를 얻는다. 각각에 대해서 모형, $\{\varphi_1(\boldsymbol{x}, \mathcal{L}^{(B)}), \cdots, \varphi_M(\boldsymbol{x}, \mathcal{L}^{(B)})\}$을 세우기 위해 forward stepwise regression을 수행한다. 여기서 $\varphi_m(\boldsymbol{x}, \mathcal{L}^{(B)})$은 M개의 변수 중 m개만 사용하며 훈련 데이터로는 bootstrap 표본을 사용한 모형이다. 하나의 bootstrap replicate 세트에 대해서 $\{\varphi_1(\boldsymbol{x}, \mathcal{L}^{(B)}), \cdots, \varphi_M(\boldsymbol{x}, \mathcal{L}^{(B)})\}$을 만드는 것인데, 이것을 50번 반복한다. 그러면 예를 들어 $\varphi_1(\boldsymbol{x}, \mathcal{L}^{(B_1)}), \cdots, \varphi_1(\boldsymbol{x}, \mathcal{L}^{(B_{50})})$와 같이, 50개이 모형이 만들어질 것이고 이를 평균하여 bagged sequence를 만든다; $\varphi_1^{(B)}(\boldsymbol{x}), \cdots, \varphi_M^{(B)}(\boldsymbol{x})$. 각각의 prediction error를 $e_1^{(B)}, \cdots, e_M^{(B)}$으로 저장한다.

4. 2번, 3번에서 수행한 것을 총 250번 반복한다. 그러면 $e_1^{(B)}, \cdots, e_M^{(B)}$도 250개의 세트가 생길 것이고, 이를 평균내서 $\left\{\bar{e}_m^{(S)}\right\}, \left\{\bar{e}_m^{(B)}\right\}$를 계산한다.

Results

가장 눈에 띄는 점은, best bagged predictors가 항상 최소 best subset predictor만큼 좋았다는 점이다. 이 점은, 앞선 논의를 empirical하게 보여준다; unbagged된 모델이 optimal이 아니라면, bagging이 효과적이라는 것이다.

그런데, bagged predictor가 unbagged보다 더 큰 prediction error를 가지는 지점이 있었다. 이는 다음과 같이 설명할 수 있다; 모든 변수를 사용하는 선형 회귀는 꽤 안정적이다. 모형에 사용되는 변수의 수가 적어질 수록 안정성이 감소한다. 앞서 안정적인 모형 (optimal)에 대해서 bagging을 사용하는 것은 별다른 변화가 없거나 오히려 더 좋지 않을 수도 있음을 지적했다. 바로 이러한 점이 결과에 나타난 것이라고 해석할 수 있다.