1st see what is additive modeling.

Algorithm

Monday, September 20, 2021 8:23 AM

feature target column

$\{(x_i, y_i)\}_{i=1}^{n}$ $n=3$ আমাদের ক্ষেত্রে, $n = ৩$

training data x_i y_i

Input: training set $\{(x_i, y_i)\}_{i=1}^{n}$, a differentiable loss function $L(y, F(x))$, number of iterations M.

1. Initialize $f_0(x) = \arg\min_\gamma \sum_{i=1}^{N} L(y_i, \gamma)$.
2. For $m = 1$ to M:
   (a) For $i = 1, 2, \ldots, N$ compute

   residual / pseudo-residual

   row → i/m →

$$r_{im} = -\left[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}\right]_{f=f_{m-1}}.$$

   (b) Fit a regression tree to the targets r_{im} giving terminal regions R_{jm}, $j = 1, 2, \ldots, J_m$.

   (c) For $j = 1, 2, \ldots, J_m$ compute

$$\gamma_{jm} = \arg\min_\gamma \sum_{x_i \in R_{jm}} L\left(y_i, f_{m-1}(x_i) + \gamma\right).$$

   (d) Update $f_m(x) = f_{m-1}(x) + \sum_{j=1}^{J_m} \gamma_{jm} I(x \in R_{jm})$.
3. Output $\hat{f}(x) = f_M(x)$.

① $f_1(x) = f_0(x) + dT$

$f_2(x) = f_1(x) + dT2$

$f_0(x) + dT$

$f_3(x) = f_2(x) + dT3$

$f_1(x) + dT2$

$f_0(x) + dT1$

→ recursion

$f_4(x) = f_0(x) + \ldots\ldots$

$f_1(x)$ $f_2(x)$

Additive Modelling

Monday, September 20, 2021 8:28 AM

ML এ আমাদের ১টা target column আর বাকী গুলো feature column থাকে। আমরা ultimately এদের মধ্যে relation বের করি।

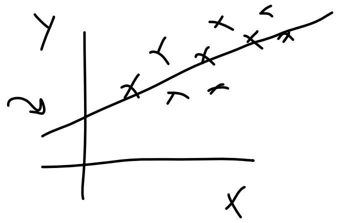
x_1, x_2

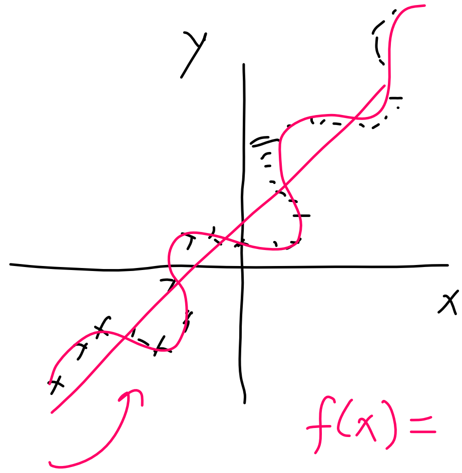
$\underline{x} \mid \cancel{y} \longrightarrow f(\,) \quad \Rightarrow \quad y = f(x)$

$x \quad y$

$x_1 \; x_2 \; x_3 \mid y$

$y = f(x_1, x_2, x_3)$





$f(x) = x + \sin x$

$y = x \qquad y = \sin(x)$

additive

$F(x) = f_0(x) + f_1(x) + f_2(x) + \dots$

DT   DT

এখন, আমাদের এই f^n এর ২টা গ্রাফ থাকবে। এখন, f^n টা একটা line দিয়ে classify করা গেলে এতে আমরা LR ব্যবহার করবো। কিন্তু, function টা আরেক complex হলে, (উপরের মতো) আমরা চেষ্টা করবো polynomial regression apply করার জন্য। কিন্তু, Polynomial এ x এর power বাড়ালেই ভালো result আসবে না। কারণ, polynomial এ x এর power বাড়ালে

→ edge গুলো উপরের দিকে উঠে যায়। যেটা সাথে সাথে Runge's Phenomenon ব্যাখ্যা করে (search google for more)। এর জন্য আমরা polynomial regression ব্যবহার করতে পারবো না। এই সমস্যার সমাধান হচ্ছে additive modeling। Additive modeling এ আমরা একটা f^n কে, sum of more than one function এ প্রকাশ করি। যেটা overall graph কে ভালোভাবে explain করতে পারে।

$$F(x) = \underbrace{f_0(x)}_{\text{(mean calculation করবে)}} + \underbrace{f_1(x) + f_2(x) + f_3(x) + \dots f_n(x)}_{\text{(decisition tree)}}$$

Gradient boosting এ

step: 1 of Algorithm → [loss fn বলে আমার algorithm কতটুকু ভুল করছে।]

Explanation

Monday, September 20, 2021 8:25 AM

Differentiable loss fn → (least square error)

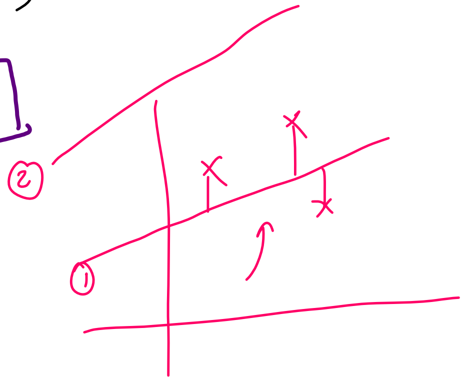
$$L = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \longrightarrow L(y, F(x)) \rightarrow L(y, \hat{y})$$

$\boxed{\hat{y}}$

actual pred

ls

$$L = \frac{1}{2} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \longrightarrow \text{diff}$$



এখানে, $\frac{1}{2}$ যোগ করলে result এর কোন পরিবর্তন আসে না calculation easy করার জন্য ½ গুন করেছি।

① $= \frac{10}{2} = 5$

② $= \frac{20}{2} = 10$

$y = f(x)$

mean DT

$$f(x) = f_0(x) + f_1(x) + f_2(x) + \dots + f_n(x)$$

$$L = \frac{1}{2} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

$$f_0(x) = \arg\min_{\gamma} \sum_{i=1}^{n} L(y_i, \gamma)$$

$$\boxed{f_0(x) = \arg\min_{\gamma} \frac{1}{2} \sum_{i=1}^{n} (y_i - \gamma)^2}$$

→ γ এর কোন মানের জন্য $\frac{1}{2}\sum_{i=1}^{n} (y_i - \gamma)^2$ মান সর্বনিম্ন হবে?? differentiation করা লাগবে। আর আমাদের example এ column 3 টা তাই n=3

$$\frac{d f_0(x)}{d\gamma} = \frac{d}{d\gamma} \frac{1}{2} \sum_{i=1}^{n} (y_i - \gamma)^2 = \frac{1}{2} \sum_{i=1}^{n} \frac{d}{d\gamma} (y_i - \gamma)^2$$

$$\sum_{i=1}^{n} (y_i - \gamma) \frac{d}{d\gamma} (y_i - \gamma) = -\sum_{i=1}^{n} (y_i - \gamma) = 0$$

$$\sum_{i=1}^{n} (\gamma - y_i) = 0$$

$$\sum_{i=1}^{3} (\gamma - y_i) = 0 \Rightarrow (\gamma - 192) + (\gamma - 144) + (\gamma - 91) = 0$$

$$\sum_{i=1}^{3} (\gamma - y_i) = 0 \Rightarrow (\gamma - 192) + (\gamma - 144) + \ldots$$

$$3\gamma = 192 + 144 + 91$$

mean

$F_m(x) = f_0(x)$ → mean of output

$$\boxed{\gamma = \frac{192 + 144 + 91}{3}}$$

→ এ জন্য আমরা mean বের করি mathematically প্রমাণিত

→ least square এর জন্য always mean আসবে।

step: 2(a)

$$F(x) = f_0(x) + \boxed{f_1(x) + f_2(x) + \ldots + f_m(x)}$$

$f_1(x)$ → DT

$f_0(x)$ → mean (leaf)

→ সব গুলো DT তাই loop ব্যবহার করতেছি।

$\boxed{m=1}$ → Number of DT

2(a) i = row number, m = dt number

ওখানে, r হচ্ছে pseudo residual. (see-106)

$$r_{im} = -\left[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}\right]_{f = f_{m-1}}$$

→ অর্থাৎ প্রত্যেক row এর জন্য r_{im} এর value বের করতে হবে। আমাদের কাছে, total row হচ্ছে 3টা

(3) row → r_{11}, r_{21}, r_{31}

r_{11}: 1st dt এর জন্য 1st row এর pseudo residual

r_{21}: 1st dT এর জন্য 2nd row এর pseudo residual

$$r_{11} = -\left[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}\right]_{f = f_0}$$

$$\hat{y}_i = f(x_i)$$

$$L = \frac{1}{2}\sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

$$r_{11} = -\left[\frac{\partial}{\partial \hat{y}_i} L(y_i, \hat{y}_i)\right]_{f = f_0}$$

$$r_{11} = -\left[\frac{\partial}{\partial \hat{y}_i} \frac{1}{2} (y_i - \hat{y}_i)^2\right]_{f = f_0}$$

$$= \left[(y_i - \hat{y}_i)\right]_{f = f_0} = \left[(y_i - f(x_i))\right]_{f = f_0}$$

↳ (f এর জায়গায় f_0 হবে)

$$= L(y_i - \hat{y}_i)\Big]_{f=f_0} = L(y_i - f(x_i))\Big]_{f=f_0}$$

$$r_{i1} = (y_i - f_0(x_i))$$

Psedo Residual for 1st DT

Now,

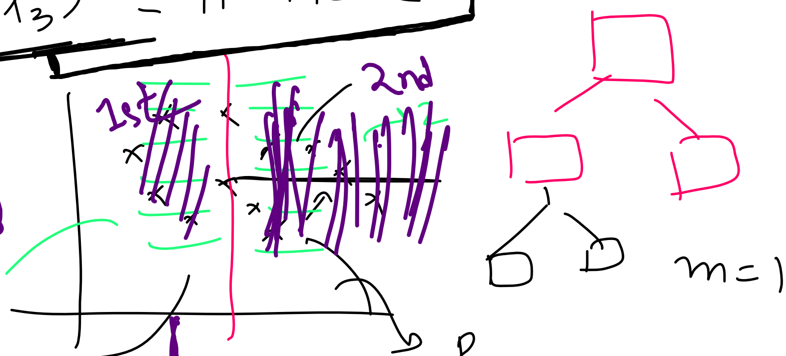
$r_{11} = y_1 - f_0(x_1) = 192 - 142 =$ → mean 1st row

$r_{21} = y_2 - f_0(x_2) = 144 - 142 =$ → mean 2nd row

$r_{31} = y_3 - f_0(x_3) = 91 - 142 =$ → " 3rd row

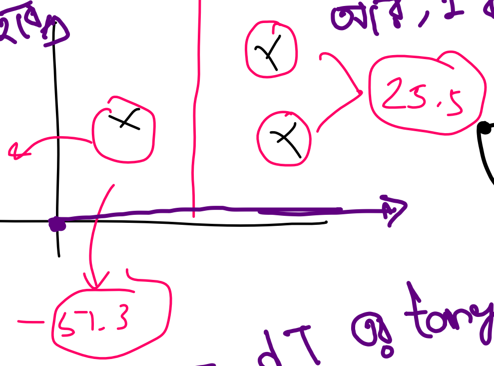
যেইটা 1st step এ বের করেছিলাম

2(b) fit a regression tree with max-depth 1। কারণ আমাদের dataset ছোট।



আমরা, max_depth=1 নিয়ে DT কে train করেছি। যার ফলে, DT, data কে দুইভাগে ভাগ করেছে। 1st region কে terminal region1, R_{11}, 2nd কে R_{21} বলতে পারি আর, 1 কারণ 1st decision tree এর জন্য।

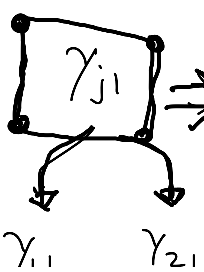
plot-decision করলে, 2টা value পাবো। আর value গুলো হবে residual-error এর জন্য, আমাদের যদি mean থাকতো value। আমাদের dataset এ 3টা data আছে। আর residual error কারণ, dT এর target column ছিল residual error।



R_{jm} → J → terminal region number
m → DT Number

2(c) Gradient boosting এর algorithm এ

$$\gamma_{jm} = \underset{\gamma}{\arg\min} \sum_{x_i \in R_{jm}} L(y_i, f_{m-1}(x_i) + \gamma)$$



আমাদের ক্ষেত্রে J এর মান 2টা

$$\Rightarrow \gamma_{11} = \underset{\gamma}{\arg\min} \sum_{x_i \in R_{11}} L(y_i, f_{m-1}(x_i) + \gamma)$$

→ এর মানে, আমাদের terminal region এর মধ্যে যেই row গুলো আছে, শুধু সেইগুলোই নিব।

$\boxed{m=1\text{, আমাদের 1st DT}}$

$\downarrow \qquad \downarrow$

$\gamma_{11} \qquad \gamma_{21}$

$$\gamma_{11} = \underset{\gamma}{\arg\min} \ \frac{1}{2}\left(y_i - (f_0(x_i) + \gamma\right)^2$$

$$\frac{dL}{d\gamma} = \frac{1}{\cancel{2}} \times \cancel{2}\left(y_i - (f_0(x) + \gamma\right) \frac{d}{d\gamma}\left(y_i - f_0(x) - \gamma\right) = 0$$

$$= -\left(y_i - f_0(x) - \gamma\right) = 0$$

$$= y_i - f_0(x) - \gamma = 0$$

(value from code)

$$\gamma_{11} = 91 - 142 - \gamma = 0$$

$$\boxed{\gamma = 91 - 142 = -51}$$

$$\gamma_{21} = \underset{\gamma}{\arg\min} \sum_{x_i \in R_{21}} L\left(y_i, f_0(x_i) + \gamma\right)$$

summation two টাই বার আমাদের terminal region এ ২টো পয়েন্ট আছে।

$$= \underset{\gamma}{\arg\min} \ \frac{1}{2} \sum_{i=1}^{2} \left(y_i - (f_0(x_i) + \gamma\right)^2 =$$

$$= -\sum_{i=1}^{2} \left(y_i - f_0(x_i) - \gamma\right) = 0$$

$$= \sum_{i=1}^{2} \left(y_i - f_0(x_i) - \gamma\right) = 0$$

$$= y_1 - f_0(x_1) - \gamma + y_2 - f_0(x_2) - \gamma = 0$$

336
284

$$= 192 - 142 - \gamma + 144 - 142 - \gamma = 0$$

$\underbrace{336 - 28}4$

$52 - 2\gamma = 0$

$\gamma = \frac{\cancel{52}}{\cancel{2}} \; 26$.

খেয়াল কর , যদি loss fn Least square ব্যবহার করি, তাহলে, dT এর leaf Node এর value কে, Gradient boosting এর output এর value সেম আসবে।