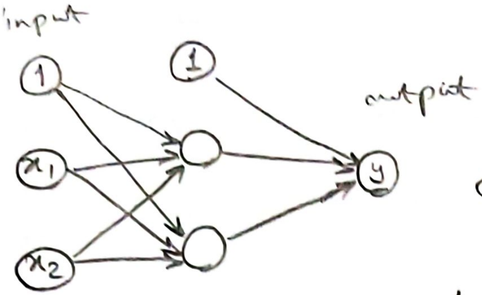
تمرین ۳ یادگیری ماشین

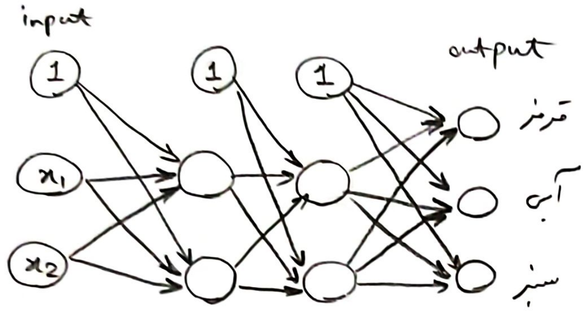
زهرا ریحانیان   شماره دانشجویی: ۸۱۰۱۰۱۱۷۷   تاریخ: ۱۴۰۱/۹/۲۷

Ⓐ


① الف) برای طبقه بندی شکل A به یک منحنی نیاز داریم و بنابراین از لایه مخفی استفاده می کنیم.

$y = 0$ or 1

ورودی، مختصات نقاط است که x_1 و x_2 آن ها را نامیدیم و سه نورون در لایه مخفی که یکی از آن ها بایاس است در نظر گرفتیم و خروجی یک نورون است. 0 یعنی آبی و 1 یعنی قرمز است.

Ⓑ


برای طبقه بندی شکل B، به دو خط نیاز داریم برای همین به لایه مخفی نیاز داریم.

ورودی، مختصات نقاط است که x_1 و x_2 آن ها را نامیدیم و سه نورون در لایه ی مخفی اول که یکی از آن ها بایاس است در نظر گرفتیم. لایه ی مخفی دوم هم به همین صورت. خروجی شامل ۳ نورون است که هر یک از رنگ ها را مشخص می کند.

ب)

Ⓐ

x_1	x_2			
0	0	$\rightarrow$	$z = 0$	$\xrightarrow{z > -0.15}$ $Y = 1$
0	1	$\rightarrow$	$z = -0.1$	$\xrightarrow{\text{"}}$ $Y = 1$
1	0	$\rightarrow$	$z = -0.1$	$\xrightarrow{\text{"}}$ $Y = 1$
1	1	$\rightarrow$	$z = -0.2$	$\xrightarrow{z < -0.15}$ $Y = 0$

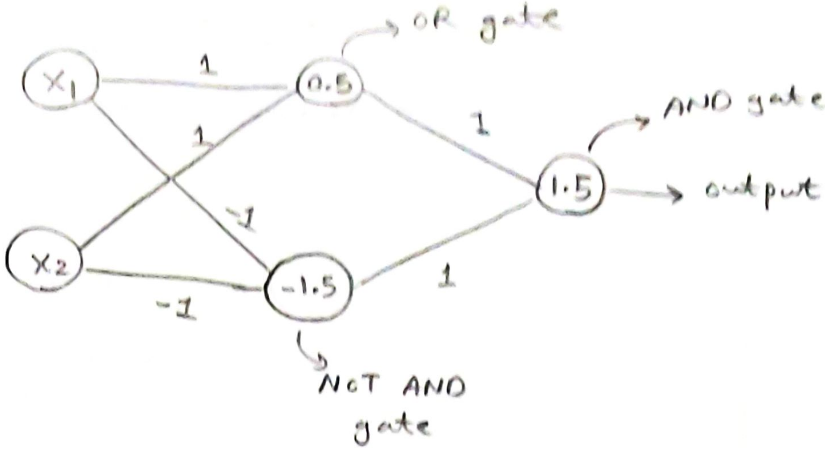
همان طور که مشاهده می شود عملکرد AND! یا همان NAND را دارد.

Ⓑ

x_1	x_2			
0	0	$\rightarrow$	$z = 0$	$\xrightarrow{z > -0.05}$ $Y = 1$
0	1	$\rightarrow$	$z = -0.1$	$\xrightarrow{z < -0.05}$ $Y = 0$
1	0	$\rightarrow$	$z = -0.1$	$\xrightarrow{\text{"}}$ $Y = 0$
1	1	$\rightarrow$	$z = -0.2$	$\xrightarrow{\text{"}}$ $Y = 0$

همان طور که مشاهده می شود عملکرد OR! یا همان NOR را دارد.

(2



مقادیر داخل نورون های لایه مخفی و خروجی ، مقدار آستانه تحریک هستند .

X_1	X_2		
0	0	$\longrightarrow$ $0\times1 + 1\times1 = 1 \longrightarrow Y=0$ ✓	
0	1	$\longrightarrow$ $1\times1 + 1\times1 = 2 \longrightarrow Y=1$ ✓	
1	0	$\longrightarrow$ $1\times1 + 1\times1 = 2 \longrightarrow Y=1$ ✓	
1	1	$\longrightarrow$ $1\times1 + 0\times1 = 1 \longrightarrow Y=0$ ✓	

۲

$$Z_1 = w_{01} + w_{11}X_1 + w_{21}X_2$$

$$Z_2 = w_{02} + w_{12}X_1 + w_{22}X_2$$

$$a_1 = \sigma(Z_1) = \frac{1}{1+e^{-Z_1}}$$

$$a_2 = \sigma(Z_2) = \frac{1}{1+e^{-Z_2}}$$

$$Y = V_0 + V_1 a_1 + V_2 a_2 = \text{predicted}$$

$$\text{Error} = 0.5\,(\text{predicted} - \text{target})^2$$

$$\frac{d\text{Error}}{dv_0} = \frac{d\text{Error}}{d\text{predicted}} \times \frac{d\text{predicted}}{dv_0} \qquad v_0 = v_0' - \alpha\frac{d\text{Error}}{dv_0}$$

$$\frac{d\text{Error}}{dv_1} = \frac{d\text{Error}}{d\text{predicted}} \times \frac{d\text{predicted}}{dv_1} \qquad v_1 = v_1' - \alpha\frac{d\text{Error}}{dv_1}$$

$$\frac{d\text{Error}}{dv_2} = \frac{d\text{Error}}{d\text{predicted}} \times \frac{d\text{predicted}}{dv_2} \qquad v_2 = v_2' - \alpha\frac{d\text{Error}}{dv_2}$$

$$\frac{d\,Error}{dw_{01}} = \frac{d\,Error}{d\,predicted} \times \frac{d\,predicted}{da_1} \times \frac{da_1}{dz_1} \times \frac{dz_1}{dw_{01}} \qquad w_{01} = w_{01}' - \alpha \frac{d\,Error}{dw_{01}}$$

$$\frac{d\,Error}{dw_{11}} = \frac{d\,Error}{d\,predicted} \times \frac{d\,predicted}{da_1} \times \frac{da_1}{dz_1} \times \frac{dz_1}{dw_{11}} \qquad w_{11} = w_{11}' - \alpha \frac{d\,Error}{dw_{11}}$$

$$\frac{d\,Error}{dw_{21}} = \frac{d\,Error}{d\,predicted} \times \frac{d\,predicted}{da_1} \times \frac{da_1}{dz_1} \times \frac{dz_1}{dw_{21}} \qquad w_{21} = w_{21}' - \alpha \frac{d\,Error}{dw_{21}}$$

$$\frac{d\,Error}{dw_{02}} = \frac{d\,Error}{d\,predicted} \times \frac{d\,predicted}{da_2} \times \frac{da_2}{dz_2} \times \frac{dz_2}{dw_{02}} \qquad w_{02} = w_{02}' - \alpha \frac{d\,Error}{dw_{02}}$$

$$\frac{d\,Error}{dw_{12}} = \frac{d\,Error}{d\,predicted} \times \frac{d\,predicted}{da_2} \times \frac{da_2}{dz_2} \times \frac{dz_2}{dw_{12}} \qquad w_{12} = w_{12}' - \alpha \frac{d\,Error}{dw_{12}}$$

$$\frac{d\,Error}{dw_{22}} = \frac{d\,Error}{d\,predicted} \times \frac{d\,predicted}{da_2} \times \frac{da_2}{dz_2} \times \frac{dz_2}{dw_{22}} \qquad w_{22} = w_{22}' - \alpha \frac{d\,Error}{dw_{22}}$$

على فرض ($\alpha = 0.2$) :

$z_1 = 1.1$ $\qquad z_2 = 0.19$ $\qquad a_1 = 0.75$ $\qquad a_2 = 0.55$

$Y = 0.13$ $\qquad Error = 0.008$

$$\frac{d\,Error}{dv_0} = 0.5 \times 2\,(predicted - target) \times 1 = 0.13 \qquad v_0 = -0.3 - 0.2 \times 0.13 = -0.326$$

$$\frac{d\,Error}{dv_1} = 0.5 \times 2\,(predicted - target) \times a_1 = 0.097 \qquad v_1 = 0.5 - 0.2 \times 0.097 = 0.4806$$

$$\frac{d\,Error}{dv_2} = 0.5 \times 2\,(predicted - target) \times a_2 = 0.071 \qquad v_2 = 0.1 - 0.2 \times 0.071 = 0.0858$$

$$\frac{d\,Error}{dw_{01}} = 0.5 \times 2\,(predicted - target) \times v_1 \times \frac{e^{-z_1}}{(1+e^{-z_1})^2} \times 1 = 0.012 \qquad w_{01} = 0.4 - 0.2 \times 0.012 = 0.39$$

$$\frac{d\,Error}{dw_{11}} = 0.5 \times 2\,(predicted - target) \times v_1 \times \frac{e^{-z_1}}{(1+e^{-z_1})^2} \times x_1 = 0.012 \qquad w_{11} = 0.7 - 0.2 \times 0.012 = 0.69$$

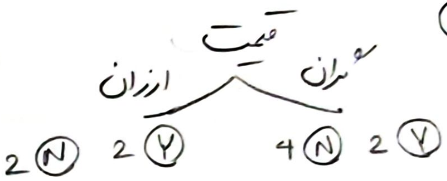
$$\frac{d\,Error}{dw_{21}} = 0.5 \times 2\,(predicted - target) \times v_1 \times \frac{e^{-z_1}}{(1+e^{-z_1})^2} \times x_2 = 0 \qquad w_{21} = -0.2 - 0.2 \times 0 = -0.2$$

$$\frac{dError}{dw_{02}} = 0.5 \times 2\,(predicted - target) \times V_2 \times \frac{e^{-z_2}}{(1+e^{-z_2})^2} \times 1 = 0.003 \qquad w_{02} = 0.6 - 0.2 \times 0.003 = 0.598$$

$$\frac{dError}{dw_{12}} = 0.5 \times 2\,(predicted - target) \times V_2 \times \frac{e^{-z_2}}{(1+e^{-z_2})^2} \times x_1 = 0.003 \qquad w_{12} = -0.4 - 0.2 \times 0.003 = -0.402$$

$$\frac{dError}{dw_{22}} = 0.5 \times 2\,(predicted - target) \times V_2 \times \frac{e^{-z_2}}{(1+e^{-z_2})^2} \times x_2 = 0 \qquad w_{22} = 0.3 - 0.2 \times 0 = 0.3$$

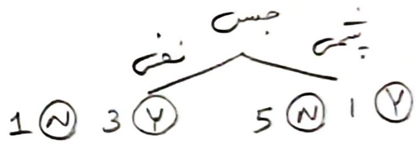
۳



$$H(Y|X=\text{ارزان}) = -\frac{1}{2}\log\frac{1}{2} - \frac{1}{2}\log\frac{1}{2} = -\log\frac{1}{2} = 0.3$$

$$H(Y|X=\text{گران}) = -\frac{1}{3}\log\frac{1}{3} - \frac{2}{3}\log\frac{2}{3} = 0.27$$

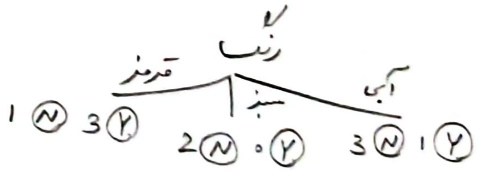
$$H(Y) = \frac{4}{10} \times 0.3 + \frac{6}{10} \times 0.27 = 0.282$$



$$H(Y|X=\text{نخی}) = -\frac{1}{4}\log\frac{1}{4} - \frac{3}{4}\log\frac{3}{4} = 0.244$$

$$H(Y|X=\text{پشمی}) = -\frac{1}{6}\log\frac{1}{6} - \frac{5}{6}\log\frac{5}{6} = 0.195$$

$$H(Y) = \frac{4}{10} \times 0.244 + \frac{6}{10} \times 0.195 = 0.214$$



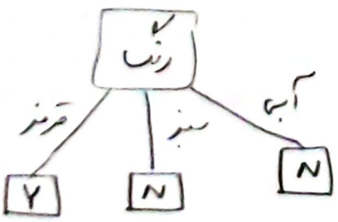
$$H(Y|X=\text{آبی}) = -\frac{1}{4}\log\frac{1}{4} - \frac{3}{4}\log\frac{3}{4} = 0.244$$

$$H(Y|X=\text{سبز}) = 0 - 1\log 1 = 0$$

$$H(Y|X=\text{قرمز}) = -\frac{1}{4}\log\frac{1}{4} - \frac{3}{4}\log\frac{3}{4} = 0.244$$

$$H(Y) = \frac{4}{10} \times 0.244 + \frac{2}{10} \times 0 + \frac{4}{10} \times 0.244 = \frac{8}{10} \times 0.244 = 0.195$$

از آن جایی که آنتروپی برای رنگ از دو ویژگی دیگر کمتر شده است پس این ویژگی را در ریشه قرار می‌دهیم و برچسب اکثریت را به برگ نسبت می‌دهیم:



$$\text{دقت} = \frac{TP+TN}{TP+FP+TN+FN} = \frac{8}{10} = \boxed{0.8}$$

$$\text{صحت} = \frac{TP}{TP+FP} = \frac{3}{4} = \boxed{0.75}$$

۴) الف) میانگین تخمین پارزن به صورت زیر حساب می‌شود:

$$\bar{P}_n(x) = E[P_n(x)] = E\left[\frac{1}{n}\sum_{i=1}^{n}\frac{1}{V_n}\phi\left(\frac{x-x_i}{h_n}\right)\right] = \frac{1}{\cancel{n}} \times \cancel{n}\, E_y\left[\frac{1}{V_n}\phi\left(\frac{x-y}{h_n}\right)\right]$$

$$= \int \frac{1}{V_n}\phi\left(\frac{x-y}{h_n}\right) p(y)\, dy = \frac{1}{V_n}\int p(y)\,\phi\left(\frac{x-y}{h_n}\right) dy$$

طبق فرض سوال $p(y) \sim U(0,a)$ که یعنی اگر خارج بازه‌ی $0<y<a$ باشیم $p(y)=0$ می‌شود.

$$x<0 \longrightarrow \bar{P}_n(x) = \frac{1}{V_n}\int_{-\infty}^{+\infty} p(y)\,\phi\left(\frac{x-y}{h_n}\right) dy = \frac{1}{V_n}\int_0^a p(y)\,\phi\left(\frac{x-y}{h_n}\right) dy$$

$$\xrightarrow[x<y \rightarrow \frac{x-y}{h_n}<0,\ h_n>0]{0<y<a} = \frac{1}{V_n}\int_0^a p(y) \times 0\, dy = 0$$

$$0 \le x \le a \longrightarrow \bar{P}_n(x) = \frac{1}{V_n}\int_{-\infty}^{+\infty} p(y)\,\phi\left(\frac{x-y}{h_n}\right) dy = \frac{1}{h_n}\int_0^x \frac{1}{a}\exp\left(\frac{y-x}{h_n}\right) dy$$

$$= \frac{1}{a\cancel{h_n}}\cancel{h_n}\exp\left(\frac{y-x}{h_n}\right)\Big|_0^x = \frac{1}{a}\left(1-\exp\left(\frac{-x}{h_n}\right)\right)$$

$$= \frac{1}{a}\left(1-e^{-\frac{x}{h_n}}\right)$$

$$x \ge a \longrightarrow \bar{P}_n(x) = \frac{1}{V_n}\int_{-\infty}^{+\infty} p(y)\,\phi\left(\frac{x-y}{h_n}\right) dy = \frac{1}{h_n}\int_0^a \frac{1}{a}\exp\left(\frac{y-x}{h_n}\right) dy$$

$$= \frac{1}{a\cancel{h_n}}\cancel{h_n}\exp\left(\frac{y-x}{h_n}\right)\Big|_0^a = \frac{1}{a}\left(\exp\left(\frac{a-x}{h_n}\right) - \exp\left(\frac{-x}{h_n}\right)\right)$$

$$= \frac{1}{a}\left(\exp\left(\frac{a}{h_n}\right)-1\right)\exp\left(\frac{-x}{h_n}\right)$$

$$= \frac{1}{a}\left(e^{\frac{a}{h_n}}-1\right)e^{-\frac{x}{h_n}}$$

ب) بایاس برابر $E(p(x)-\hat{p}(x)) = p(x)-\bar{p}(x)$ است. بایاس نسبی به صورت زیر محاسبه می‌شود:

طبق فرض سوال در بازه‌ی ۰ تا a را در نظر می‌گیریم

$$\text{bias}(x) = \frac{|p(x)-\bar{p}(x)|}{p(x)} = \frac{\frac{1}{a}-\bar{p}(x)}{\frac{1}{a}} = 1 - a\,\bar{p}(x) = 1 - a \times \frac{1}{a}\left(1-e^{-\frac{x}{h_n}}\right) = e^{-\frac{x}{h_n}}$$

بایاس در بازه‌ی $0<x<a$ کاهش می‌یابد. پس اگر بخواهیم که اندازه‌ی بایاس تخمین پارزن، در ۹۹٪ از طول بازه‌ی ۰ تا a، کمتر از یک درصد یا همان 0.01 باشد نیاز است که رابطه‌ی زیر را محاسبه کنیم:

$$\text{bias}\left(\frac{a}{100}\right) = 0.01 \longrightarrow e^{-a/100h_n} = 0.01$$

$$\xrightarrow{\text{ln می‌گیریم}} -\frac{a}{100h_n} = -\ln 100 \longrightarrow \boxed{h_n = a/(100 \ln 100)}$$

---

(۷) ابتدا کتابخانه های موردنیاز را import می‌کنیم. سپس داده های iris را لود می‌کنیم.

هرکدام از داده های مربوط به نوع setosa و versicolor را در دو دیتافریم قرار می‌دهیم و در نهایت نمودار نقاط آن ها را در یک نمودار رسم می‌کنیم.

ابتدا به آماده سازی داده ها می‌پردازیم. از داده ها، آن هایی که نوعشان setosa یا versicolor است را انتخاب می‌کنیم. X که همان فیچرهای sepal length, sepal width هستند و y که خروجی است را جدا می‌کنیم. نوع setosa را به 1 و versicolor را به 0 map می‌کنیم که عملیات را راحت تر انجام دهیم. داده های test و train را جدا می‌کنیم. در نهایت تابع train_perceptron را با ایپاک 1000 و نرخ یادگیری 10^{-6} اجرا می‌کنیم.

این تابع به این صورت عمل می‌کند که ابتدا bias و وزن های اولیه را 0 قرار می‌دهد. سپس در هر ایپک به این صورت عمل می‌کند:

به ازای هر x، ابتدا label آن را predict می‌کند. برای این کار تابع predict را فراخوانی می‌کند. این تابع به این صورت عمل می‌کند که حاصل wx یعنی ضرب داخلی x و وزن های فعلی را حساب می‌کند و با b مقایسه می‌کند. اگر بزرگتر از مساوی b بود 1 وگرنه 0 برمی‌گرداند. بعد از این مراحل نوبت به بروزرسانی وزن ها و b با استفاده از gradient descent می‌رسد که به این صورت است:

$$w_k = w_k - \alpha \frac{\partial\,\text{error}}{\partial w_i} \longrightarrow w_k = w_k + \alpha\,(y - \hat{y})\, x_k^{(i)}$$

$$b = b - \alpha \frac{\partial\,\text{error}}{\partial b} \longrightarrow b = b + \alpha\,(y - \hat{y})$$

که بر این مبنا که آن ها زده شد و در نهایت b و w برگردانده می‌شود. وزن های برگردانده شده به این صورت بود: $[-4.8e^{-9},\ 8.8e^{-9}]$ و bias به این صورت است: $2e^{-9}$

که چون از b به عنوان threshold استفاده کردیم معادله خط به این صورت درمی‌آید:

$$w_1x_1 + w_2x_2 - b = 0 \longrightarrow x_2 = -(w_1x_1 - b)/w_2$$

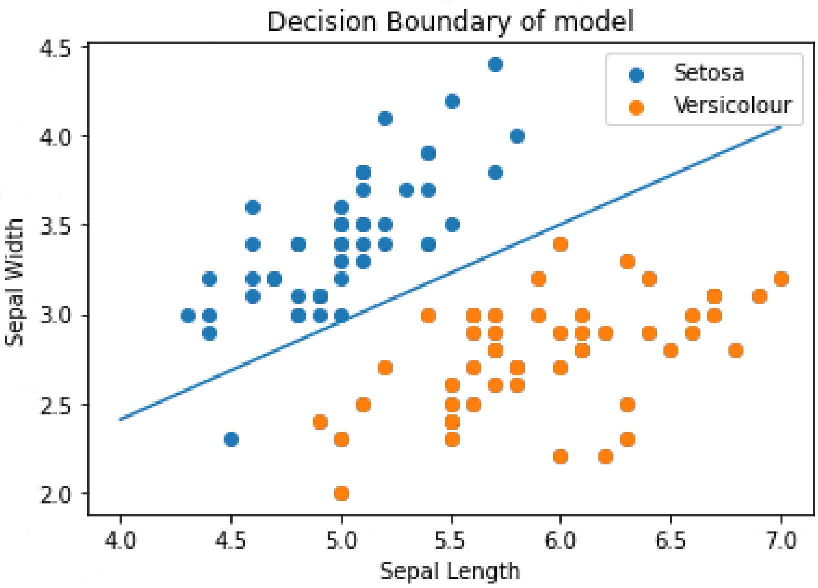
که شیب خط برابر می‌شود با : $-w_1/w_2 = 0.\overline{54}$

و عرض از مبداء برابر است با : $b/w_2 = 0.22\overline{72}$

دقت این الگوریتم نیز به دست آورده شده است که برابر 98.75 % می باشد .

برای این کار تابع test_perceptron فراخوانی شده است . که به ازای هر x داده شده ، y آن را با استفاده از وزن های بدست آمده در مرحله train و threshold که آن همان b بدست آمده در مرحله train است ، predict می‌کند . اگر مقدار پیش بینی شده درست بود یک واحد به تعداد پیش بینی درست اضافه می‌کند و در نهایت دقت را بدست آورده و گزارش می‌کند .

خط بدست آمده برای جدا کردن داده های دوگونه گفته شده در سوال به صورت زیر است :



۸) ابتدا با استفاده از دستورات داده شده ، داده ها تولید شد ، سپس هیستوگرام آن ها را رسم کردم که یک دید کلی نسبت به داده ها پیدا کنم .

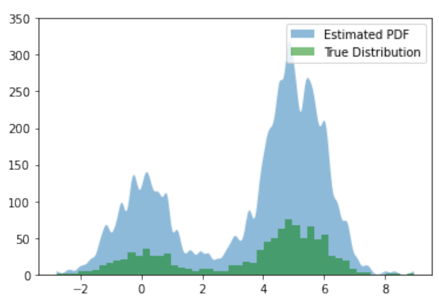
در مرحله بعد پنجره ی پارزن را پیاده سازی کردم ، که اساس کار آن طبق فرمول زیر می باشد:

$$KDE = \sum_{i=1}^{n} \frac{1}{h\sqrt{2\pi}} e^{-0.5\left(\frac{x-x_i}{h}\right)^2}$$

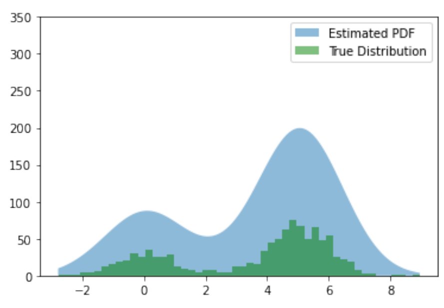
ابتدا به تعداد داده هایی که تولید کردیم ، یک دنباله ی عددی که min و max و تعداد آن برابر داده ی تولید شده است ، با استفاده از دستور np.linspace تولید می کنیم و آن ها را در متغیر points می ریزیم.
سپس به ازای هر x_i در داده ای که اول تولید کردیم ، یک توزیع گوسی با فرمول $\frac{1}{h\sqrt{2\pi}} e^{-0.5\left(\frac{x-x_i}{h}\right)^2}$ که x همان points است ، در نظر می گیریم و مجموع این توزیع ها را برمی گردانیم. به ازای ۳ مقدار مختلف برای h ، این تابع را صدا می زنیم . ابتدا h = 0.1 را امتحان کردم و نتیجه یک توزیع تقریبا اسپایکی شد که نویز زیادی دارد.

مرحله بعد تابع را با h = 1 صدا زدم که یک توزیع smooth و ساده تری نسبت به قبلی بدست آمد.
مرحله بعد تابع را با h = 10 صدا زدم که یک توزیع تقریبا flat بدست آمد و underfit بود یعنی توزیع مناسب دیتایی که داریم نیست .

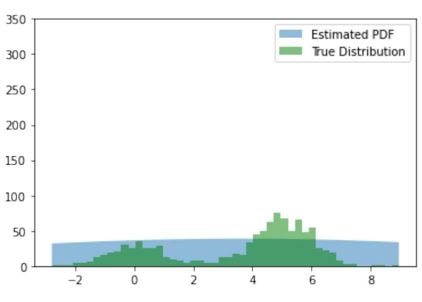
می توان نتیجه گرفت با افزایش h ، تخمین توزیع smooth تر و ساده تر می شود و برعکس با کاهش آن تخمین توزیع پیچیده تر می شود و اگر خیلی کوچک باشد یک توزیع اسپایکی که نویز زیادی دارد می دهد. این همان ویژگی تخمین گر پارزن است که با افزایش اندازه پنجره یک توزیع احتمال نرم و smooth می دهد و با کاهش آن توزیع احتمال پیچیده تر می شود و اگر خیلی کوچک باشد توزیع احتمال اسپایکی می شود و قله های با شیب زیاد داریم.



h = 0.1



h = 1



h = 10