作業 1

結構化機器學習
7107018026 劉俊廷

I. 模型參數

模型可用參數 (括號內為預設參數):
1. 節點數:nodes(40)
2. 學習次數:epochs(100)
3. 學習率:η(0.0001)
4. 最小步輻:mini batchsize(100)
5. 避免重複訓練:shuffle(T)
6. 神經層數:layers(3/4/5)
7. 隨機種子:randomseed(1)

II. 模型結構

激活函數
Relu

$$R(z) = \begin{cases} z, & \text{if } x \geq 0 \\ 0, & \text{if } x < 0 \end{cases}$$

Relu 之反向傳遞

$$R_b(a) = \begin{cases} a, & \text{if } x > 0 \\ 0, & \text{if a=0} \end{cases}$$

初始化參數

$W^k_{m\times n}$:

每個元素皆由常態分佈 $N(0, 0.1)$ 中生成, 共 $(n-1)$ 層

B_k :

初始化為 0, 共 $(n-1)$ 層

Forward
n 層神經網路中:

$$Z_2 = X \times W_1 + B1$$

$$A_2 = R(Z_2)$$

$$Z_k = A_{k-1} \times W_{k-1} + B_{k-1}$$

$$A_k = R(Z_k)$$

最後一層不加 Relu

$$A_n = Z_n$$

Cost function:SSE

$$SSE = \Sigma\|\widehat{y} - y\|^2, \text{where } \widehat{y} = a^n$$

Backward
我們知道

$$\begin{aligned} \delta_{out} &= \frac{\partial SSE}{\partial Z_n} \\ &= \frac{\partial (A_n - Y)^T (A_n - Y)}{\partial A_n} \frac{\partial A_n}{\partial Z_n} \\ &= 2(A_n - Y) \\ &= 2(Z_n - Y) = \delta_n \end{aligned}$$

$$\begin{aligned} \delta_k &= \frac{\partial SSE}{\partial Z_k} \\ &= \frac{\partial SSE}{\partial Z_{k+1}} \frac{\partial Z_{k+1}}{\partial Z_k} \\ &= \delta_{k+1} \frac{\partial A_k}{\partial Z_k} \\ &= R_b(\delta_{k+1} W_k^T) \end{aligned}$$

所以第 n-1 層的 W 和 B 的 gradient 為

$$\begin{aligned} grad(W_{n-1}) &= \frac{\partial SSE}{\partial W_{n-1}} \\ &= \frac{\partial SSE}{\partial Z_n} \frac{\partial Z_n}{\partial W_{n-1}} \\ &= \delta_n \frac{\partial Z_n}{\partial W_{n-1}} \\ &= 2(Z_n - Y) \frac{\partial Z_n}{\partial W_{n-1}} \\ &= 2A_{n-1}^T (Z_n - Y) \end{aligned}$$

$$\begin{aligned} grad(B_{n-1}) &= \frac{\partial SSE}{\partial B_{n-1}} \\ &= \frac{\partial SSE}{\partial Z_n} \frac{\partial Z_n}{\partial B_{n-1}} \\ &= \delta_n \frac{\partial Z_n}{\partial B_{n-1}} \\ &= 2[\Sigma_{i=1}^h (Z_{ji} - Y_{ji})]_{m\times 1} \text{ ,m 為該層維度} \end{aligned}$$

而對每一層的 W 和 B 的 gradient 為

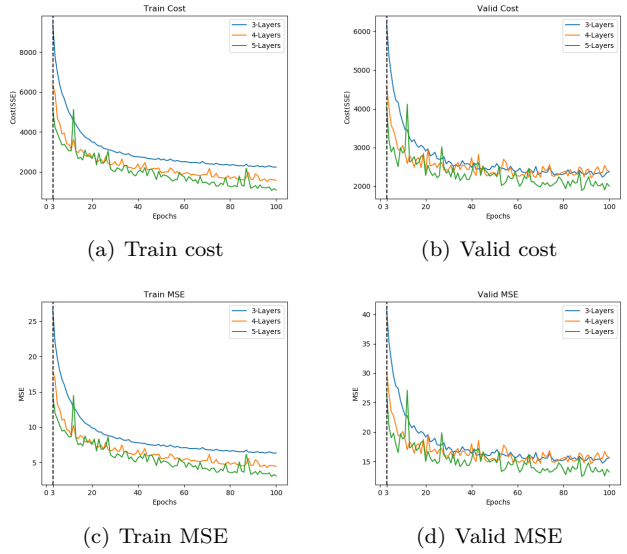
$$
\begin{aligned}
grad(W_k) &= \frac{\partial SSE}{\partial W_k} \\
&= \frac{\partial SSE}{\partial Z_{k+1}} \frac{\partial Z_{k+1}}{\partial W_k} \\
&= \delta_{k+1} \frac{\partial Z_{k+1}}{\partial W_k} \\
&= A_{k-1}^T \delta_{k+1} \\
gradB_k &= \frac{\partial SSE}{\partial B_k} \\
&= \frac{\partial SSE}{\partial Z_{k+1}} \frac{\partial Z_{k+1}}{\partial B_k} \\
&= \delta_{k+1} \frac{\partial Z_{k+1}}{\partial B_k} \\
&= [\delta_{k+1}]_{m\times 1} \text{ ,m 為資料維度}
\end{aligned}
$$

更新參數

$$
\begin{aligned}
W_k &= W_k - \eta grad(W_k) \\
B_k &= B_k - \eta grad(B_k)
\end{aligned}
$$

III. 結果

在其他條件相同下，設定了 3 層,4 層和 5 層的神經網路來比較，先由訓練每次學習的 Cost(SSE) 與 MSE 來觀察他們的學習效果:



(a) Train cost (b) Valid cost

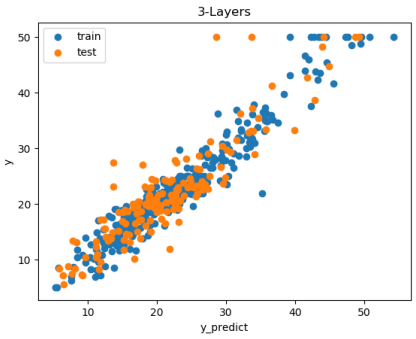
(c) Train MSE (d) Valid MSE

Fig. 1. 不同層數神經網路比較

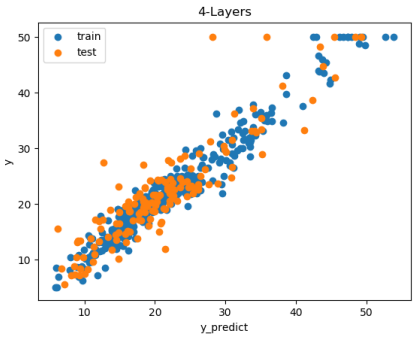
可以發現不論從驗證資料或訓練資料來看，5 層神經網路學習效果都是最好的, 而會發現雖然 4 層學習效果比 3 層還好，但最後學習完的模型與 3 層差異不大。
但值得注意的是，5 層神經網路雖然學習效果較好，但降低的 Cost 來回擺盪的幅度有點大。可以想到的原因是因為 5 層神經網路的學習速度較快，因此在接近 local minimun 的時候會衝過頭，所以會呈現來回擺盪的狀態。
接著我們個別比較各層網路訓練後的一些數據:

網路層數	R^2	MSE
3-layers train	0.92521	6.33935
4-layers train	0.94752	4.44878
5-layers train	0.96340	3.10243
3-layers valid	0.81264	15.60093
4-layers valid	0.81183	15.66832
5-layers valid	0.84127	13.21652

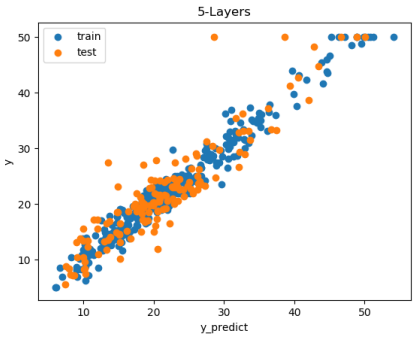
可以發現雖然 5 層神經網路的 R^2 或者 MSE 都比其他層低，但 4 層的神經網路雖然在訓練資料的數據較 3 層好，但是在驗證資料卻略差於第 3 層。



(a) 3-layers-predict



(b) 4-layers-predict



(c) 5-layers-predict

Fig. 2. 不同層數神經網路預測圖比較

比較各層數神經網路的預測圖，可以發現在 y=50 的時

模型有些失準，因此我們抓出 y=50 的資料來觀察

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV
161	1.46336	0	19.58	0	0.605	7.489	90.8	1.9709	5	403	14.7	374.43	1.73	50
162	1.83377	0	19.58	1	0.605	7.802	98.2	2.0407	5	403	14.7	389.61	1.92	50
163	1.51902	0	19.58	1	0.605	8.375	93.9	2.162	5	403	14.7	388.45	3.32	50
166	2.01019	0	19.58	0	0.605	7.929	96.2	2.0459	5	403	14.7	369.3	3.7	50
186	0.05602	0	2.46	0	0.488	7.831	53.6	3.1992	3	193	17.8	392.63	4.45	50
195	0.01381	80	0.46	0	0.422	7.875	32	5.6484	4	255	14.4	394.23	2.97	50
204	0.02009	95	2.68	0	0.4161	8.034	31.9	5.118	4	224	14.7	390.55	2.88	50
225	0.52693	0	6.2	0	0.504	8.725	83	2.8944	8	307	17.4	382	4.63	50
257	0.61154	20	3.97	0	0.647	8.704	86.9	1.801	5	264	13	389.7	5.12	50
267	0.57834	20	3.97	0	0.575	8.297	67	2.4216	5	264	13	384.54	7.44	50
283	0.01501	90	1.21	1	0.401	7.923	24.8	5.885	1	198	13.6	395.52	3.16	50
368	4.89822	0	18.1	0	0.631	4.97	100	1.3325	24	666	20.2	375.52	3.26	50
369	5.66998	0	18.1	1	0.631	6.683	96.8	1.3567	24	666	20.2	375.33	3.73	50
370	6.53876	0	18.1	1	0.631	7.016	97.5	1.2024	24	666	20.2	392.05	2.96	50
371	9.2323	0	18.1	0	0.631	6.216	100	1.1691	24	666	20.2	366.15	9.53	50
372	8.26725	0	18.1	1	0.668	5.875	89.6	1.1296	24	666	20.2	347.88	8.88	50

(a) Y=50 時的資料 (標紅為與其他值差距較大的資料)

Fig. 3. Y=50 時的資料，共 16 筆

由圖可以知道在 y=50 時，儘管自變數間有所差異，但應變數卻是一樣的，會造成我們用的自變數項難以預測

IV. 結論

儘管模型存在一些問題，但模型解釋力整體上表現不差 (五層的 R^2 可以達到 0.96/train,0.84/valid)。而在各模型間的比較中，5 層的效果最好，而 4 層和 3 層的解釋力在驗證資料上的差異不大。