محمدحسین مازندرانیان – ۸۳۰۴۰۲۰۶۶ – تمرین شماره۲ شبکه های عصبی

سوال ۱)

سوال ۱)

$$X = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 3 & 3 \\ 4 & 4 \end{pmatrix} \longrightarrow K = XX^T = \begin{pmatrix} 1 & 0 & 3 & 4 \\ 0 & 1 & 3 & 4 \\ 3 & 3 & 18 & 24 \\ 4 & 4 & 24 & 32 \end{pmatrix}$$

$$\text{dual} \longrightarrow \max \alpha_1+\alpha_2+\alpha_3+\alpha_4 - \frac{1}{2}\left(\alpha_1^2 K_{11} + \alpha_2^2 K_{22} + \alpha_3^2 K_{33} + \alpha_4^2 K_{44} + 2\alpha_1\alpha_2 K_{12} + 2\alpha_1\alpha_3 K_{13} + 2\alpha_1\alpha_4 K_{14} + 2\alpha_2\alpha_3 K_{23} + 2\alpha_2\alpha_4 K_{24} + 2\alpha_3\alpha_4 K_{34}\right)$$

$$\alpha_1+\alpha_2-\alpha_3-\alpha_4=0 \quad \alpha_i \geq 0 \xrightarrow{\text{فرض کنید}} \alpha_1=0.5=\alpha_2=\alpha_3=\alpha_4$$

$$w = \sum_{i=1}^{n} \alpha_i y_i x_i \longrightarrow w = 0.5\cdot(-1)\cdot\begin{pmatrix}0\\1\end{pmatrix} + 0.5\cdot(-1)\cdot\begin{pmatrix}3\\3\end{pmatrix} - 0.5\cdot 1\cdot\begin{pmatrix}4\\4\end{pmatrix} = \begin{pmatrix}-3\\-3\end{pmatrix}$$

$$b = y_k - w^T x_k \longrightarrow b = 1 - (-3\cdot 1 + -3\cdot 0) = 4$$

$$w^T x + b = 0 \longrightarrow -3x_1 - 3x_2 + 4 = 0$$

برای بردارهای نقاط مرزی $\longrightarrow y_i(w^T x_i + b) = 1 \longrightarrow$ نقاط (1,0) و (0,1)

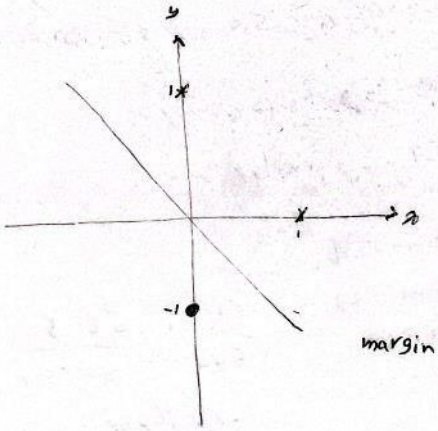
ضرایب لاگرانژ $\longrightarrow$

$$\alpha_1 = \frac{1-4\alpha_4-3\alpha_3}{1}$$

$$\alpha_2 = \frac{1-4\alpha_4-3\alpha_3}{1}$$

$$\alpha_3 = \frac{1-3\alpha_1-3\alpha_2-24\alpha_4}{18}$$

$$\alpha_4 = \frac{1-4\alpha_1-4\alpha_2-24\alpha_3}{32}$$



حل بصورت شهودی:

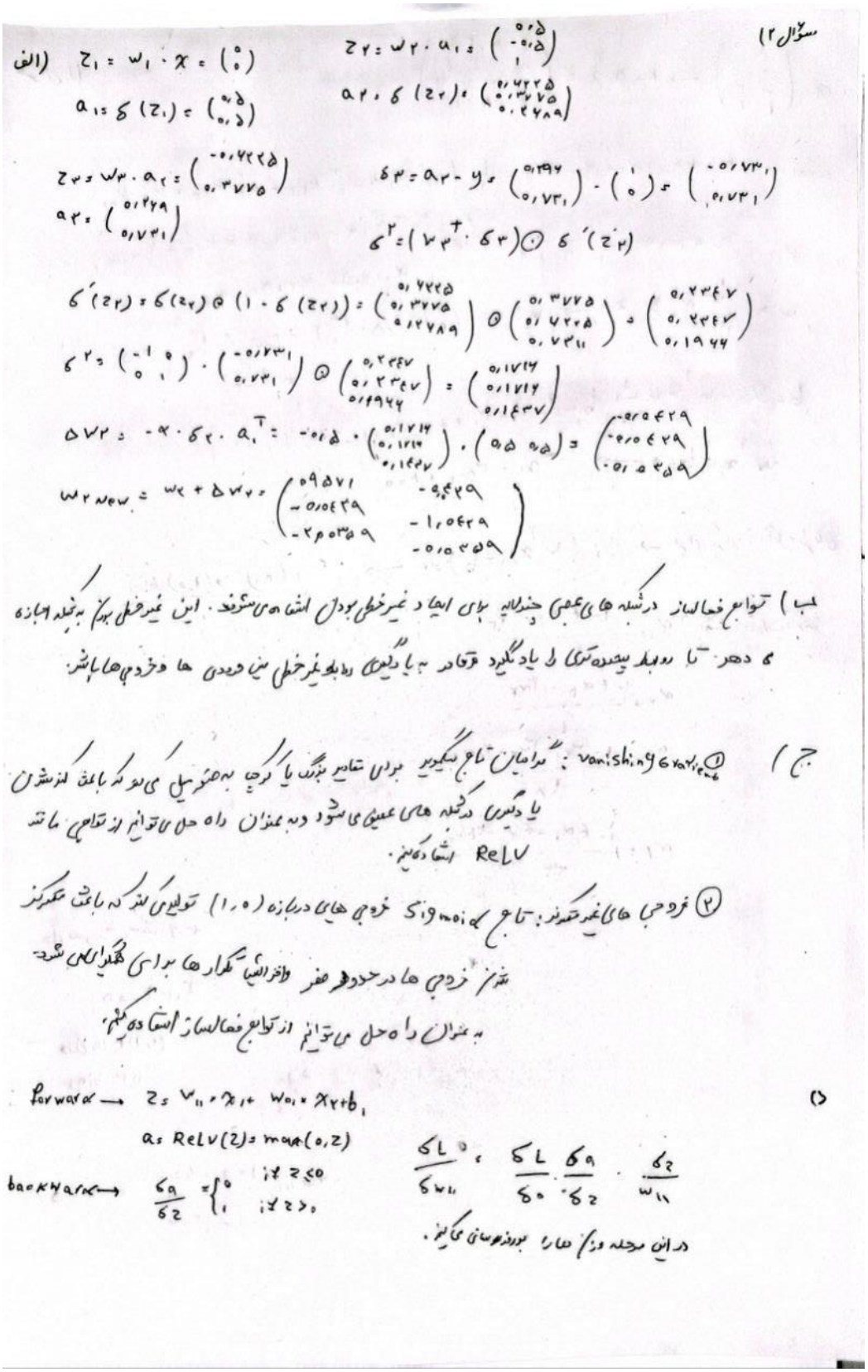
$x_1 + x_2 = 0$

وزن‌ها $\leftarrow (1,1)$

bias $\leftarrow 0$

$$\text{margin} = \frac{|w_1 x_1 + w_2 x_2 + b|}{\sqrt{w_1^2+w_2^2}} = 0.707$$

سوال ۲)



سوال ۳)

ج. بررسی راه‌حل (نیازی به پاسخ طولانی نیست، تنها رساندن مفهوم کافی‌ست)

۱. در مورد مفاهیم Grid Search و Random Search تحقیق کنید و هرکدام را مختصرا توضیح دهید.
۲. در مورد کرنل های مختلف مانند Linear، RBF و Polynomial تحقیق کنید و یک یا دو مورد از مهم ترین پارامترهای هرکدام را مختصر توضیح دهید.
۳. روش‌های one vs rest و one vs all را مختصرا توضیح داده و باهم مقایسه کنید. آیا در این مسئله نیازی به استفاده از آنها داریم؟

Grid Search

در این روش، فضای جستجو به یک شبکه منظم تقسیم می‌شود، و الگوریتم تمام نقاط شبکه را بررسی می‌کند.

ویژگی‌ها:

- **ساختار منظم** :فضای جستجو به‌طور سیستماتیک به نقاط مشخص تقسیم می‌شود.
- **جامعیت** :تمامی ترکیب‌های ممکن از پارامترها مورد بررسی قرار می‌گیرند.
- **مزایا:**
  - ساده و قابل درک است.
  - تضمین می‌کند که هیچ نقطه‌ای از فضای جستجو از قلم نمی‌افتد.
- **معایب:**
  - بسیار زمان‌بر و محاسباتی سنگین است، به‌ویژه در مسائل با تعداد زیاد پارامترها (به دلیل افزایش نمایی نقاط شبکه).

کاربرد:

معمولاً زمانی استفاده می‌شود که فضای جستجو کوچک و تعداد پارامترها محدود باشد.

Random Search

در این روش، نقاطی به‌صورت تصادفی در فضای جستجو انتخاب و بررسی می‌شوند.

ویژگی‌ها:

- **عدم ساختار منظم** :نقاط به‌طور کاملاً تصادفی انتخاب می‌شوند.
- **انعطاف‌پذیری** :برخلافSearch Grid ، نیازی به تقسیم‌بندی سیستماتیک فضای جستجو نیست.
- **مزایا:**
  - در مسائل با فضای جستجوی بزرگ‌تر، کارآمدتر از Search Grid است.
  - امکان کشف نقاط غیرمنتظره‌ای از فضای جستجو را دارد.
- **معایب:**
  - ممکن است برخی نقاط از فضای جستجو نادیده گرفته شوند.
  - نتایج وابسته به شانس و تعداد نمونه‌های تصادفی است.

کاربرد:

در مسائل پیچیده با فضای جستجوی بزرگ، زمانی که منابع محاسباتی محدود باشند.

کرنل خطی(Linear Kernel)

این کرنل ساده‌ترین نوع است و فرض می‌کند داده‌ها در فضای اصلی به‌صورت خطی جداپذیر هستند.

فرمول:

$$y \cdot x = K(x, y)$$

پارامتر مهم:

- **C (پارامتر منظم سازی)**
  - تنظیم تعادل بین حداکثرسازی حاشیه و کاهش خطا در داده‌های آموزشی.
  - مقدار بزرگ‌تر: جریمه بیشتر برای خطاها (احتمال بیش‌برازش).
  - مقدار کوچک‌تر:حاشیه بزرگ‌تر (احتمال کم‌برازش).
  -

کاربرد:

- برای مسائل ساده و داده‌های خطی مناسب است.

کرنل چندجمله‌ای(Polynomial Kernel)

این کرنل برای مسائل پیچیده‌تر با روابط غیرخطی استفاده می‌شود.

فرمول:

$$^{d}(c + y \cdot x) = K(x, y)$$

پارامترهای مهم:

۱. **) dddرجه یا:(Degree**

- تعیین می‌کند تا چه حد ویژگی‌های غیرخطی در داده مدل‌سازی شوند.
- مقدار کوچک‌تر (d=2d=2d=2 یا :(d=3d=3d=3مناسب برای روابط ساده.
- مقدار بزرگ‌تر: ممکن است پیچیدگی زیاد و بیش‌برازش ایجاد کند.

۲. **:c (Bias)**

- مقدار ثابت که تأثیر تعادل بین درجات پایین و بالا را تنظیم می‌کند.
- مقدار مناسب معمولاً با آزمایش مشخص می‌شود.

کاربرد:

- برای داده‌هایی با الگوهای پیچیده‌تر که دارای روابط چندجمله‌ای هستند.

کرنل شعاعی پایه RBF

یکی از پرکاربردترین کرنل‌ها، به‌ویژه برای مسائل غیرخطی. این کرنل داده‌ها را به فضای ویژگی‌های بسیار بزرگ نگاشت می‌کند.

فرمول:

$$(^{2}\|y - \exp(-\gamma\|x = K(x, y)$$

پارامترهای مهم:

۱. **گاما**

- تعیین‌کننده‌ی گستره تأثیر هر نمونه.
- مقدار کوچک‌تر: حاشیه وسیع‌تر (انعطاف کمتر).
- مقدار بزرگ‌تر: حاشیه باریک‌تر (ممکن است بیش‌برازش شود).

۲. **C**

- مشابه کرنل خطی، کنترل تعادل بین دقت در آموزش و تعمیم‌پذیری مدل.

کاربرد:

- برای داده‌هایی که کاملاً غیرخطی هستند و مرز تصمیم‌گیری پیچیده‌ای نیاز دارند.

۱. **روش One-vs-Rest**

در این روش، برای هر کلاس، یک دسته‌بند دوتایی آموزش داده می‌شود.

- **ایده:**
  - یک کلاس به‌عنوان مثبت در نظر گرفته می‌شود، و باقی کلاس‌ها به‌عنوان منفی.
  - به ازای هر کلاس، یک مدل دوتایی ایجاد می‌شود.
- **نحوه عملکرد:**
  - تعداد دسته‌بندها برابر با تعداد کلاس‌ها (kkk) است.
  - هنگام پیش‌بینی، نمونه به کلاسی نسبت داده می‌شود که مدل مربوطه، بالاترین احتمال یا اطمینان را داشته باشد.

مزایا:

- ساده و قابل فهم.
- محاسبات کمتر نسبت به روش.All-vs-All

معایب:

- در مسائل نامتوازن ممکن است عملکرد خوبی نداشته باشد.
- نیازمند مقایسه نتایج برای تعیین کلاس است.

۲. روش One-vs-One

در این روش، برای هر جفت از کلاس‌ها، یک دسته‌بند دوتایی آموزش داده می‌شود.

- **ایده:**
  - هر بار فقط دو کلاس در نظر گرفته می‌شوند.
  - تمام ترکیب‌های ممکن از کلاس‌ها بررسی می‌شود.
- **نحوه عملکرد:**
  - تعداد دسته‌بندها برابر با $k(k-1)2\frac{k(k-1)}{2}2k(k-1)$(تعداد جفت‌های کلاس‌ها) است.
  - هنگام پیش‌بینی، هر مدل دوتایی یک رأی می‌دهد، و کلاس با بیشترین رأی انتخاب می‌شود.

مزایا:

- مناسب برای مسائل نامتوازن.
- می‌تواند مرزهای تصمیم‌گیری دقیق‌تری ارائه دهد.

معایب:

- تعداد مدل‌های بیشتری نیاز است، که محاسبات بیشتری می‌طلبد.
- پیچیدگی افزایش می‌یابد.

در مسئله ی مطرح شده چون دسته بندی فقط دو کلاسه است نیازی به استفاده از این روش ها نیست.

```
print(grid.best_score_)
print(grid.best_estimator_)
```

```
0.7854545454545454
SVC(C=0.1, degree=2, gamma=0.1, kernel='linear', max_iter=1000)
```

```
best_model = grid.best_estimator_
y_pred = best_model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy: {accuracy}")
```

```
Accuracy: 0.7681159420289855
```

سوال ۴)

۵. یکی از مراحل پیش‌پردازش معمولا نرمال سازی یا استاندارد سازی است، ابتدا این دو روش را با هم مقایسه کنید سپس بدون استفاده از توابع آماده، آنها را روی داده‌ها اعمال کنید و نتایج را با قسمت های قبل مقایسه کنید.

نرمال‌سازی مقیاس داده‌ها را به یک بازه مشخص، معمولاً [0,1][0, 1][0,1]یا [−1,1][-1, 1][−1,1]، تبدیل می‌کند.

- فرمول رایج (Min-Max Scaling):

$$X' = \frac{X - min X}{max X - min X}$$

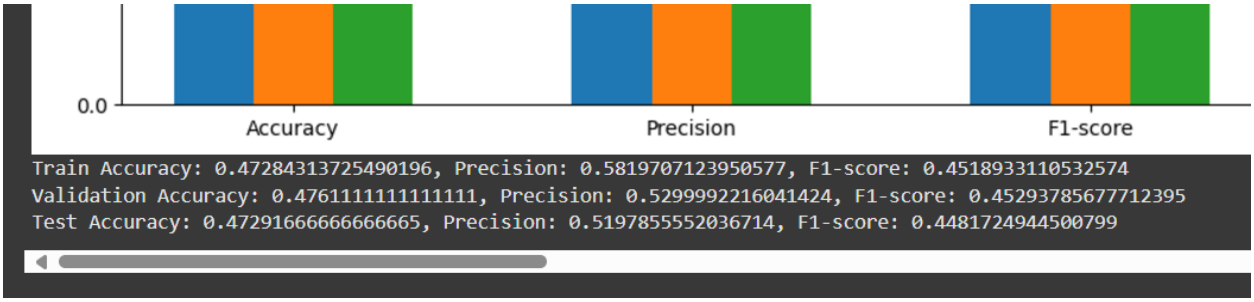
استانداردسازی داده‌ها را به‌گونه‌ای تغییر می‌دهد که **میانگین صفر** و **انحراف معیار واحد** داشته باشند.

- فرمول:

$$X' = \frac{X - \mu}{\sigma}$$

```
Epoch 1/10
797/797 ━━━━━━━━━━━━━━━━━━━━ 2s 3ms/step - accuracy: 0.6358 - loss: 1.3827 - val_accuracy: 0.
Epoch 2/10
797/797 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - accuracy: 0.8782 - loss: 0.4761 - val_accuracy: 0.
Epoch 3/10
797/797 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - accuracy: 0.8960 - loss: 0.3850 - val_accuracy: 0.
Epoch 4/10
797/797 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - accuracy: 0.9025 - loss: 0.3513 - val_accuracy: 0.
Epoch 5/10
797/797 ━━━━━━━━━━━━━━━━━━━━ 3s 2ms/step - accuracy: 0.9118 - loss: 0.3162 - val_accuracy: 0.
Epoch 6/10
797/797 ━━━━━━━━━━━━━━━━━━━━ 3s 2ms/step - accuracy: 0.9143 - loss: 0.3023 - val_accuracy: 0.
Epoch 7/10
797/797 ━━━━━━━━━━━━━━━━━━━━ 2s 2ms/step - accuracy: 0.9193 - loss: 0.2868 - val_accuracy: 0.
Epoch 8/10
797/797 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - accuracy: 0.9206 - loss: 0.2752 - val_accuracy: 0.
Epoch 9/10
797/797 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - accuracy: 0.9247 - loss: 0.2663 - val_accuracy: 0.
Epoch 10/10
797/797 ━━━━━━━━━━━━━━━━━━━━ 2s 2ms/step - accuracy: 0.9286 - loss: 0.2549 - val_accuracy: 0.
313/313 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - accuracy: 0.9247 - loss: 0.2707
Test accuracy: 0.9343000054359436
```

دقت مدل با normalization



دقت مدل بدون normalization

و. نرخ یادگیری را به مقادیر 9e-1 و 1e-5 تغییر دهید و مشاهدات خود را تفسیر کنید.

متاسفانه من تغییری بین این مقادیر مشاهده نکردم، شاید اگر تعداد epoch ها بیشتر باشند میتونیم تغییرات محسوسی داشته باشیم.

ز. در مورد بیش برازش و کم برازش در شبکه‌های عصبی توضیح دهید و برای پیشگیری و حل این مشکلات راهکارهایی پیشنهاد کنید (برای هرکدام دو مورد). نتایج قسمت‌های قبل را با توجه به این مفاهیم تحلیل کنید و بیان کنید در هر مرحله با کدام یک از این مشکلات مواجه بودید.

بیش‌برازش زمانی رخ می‌دهد که مدل به‌قدری پیچیده باشد که الگوهای جزئی یا نویز موجود در داده‌های آموزشی را نیز یاد بگیرد. در نتیجه، مدل در داده‌های آموزشی عملکرد بسیار خوبی دارد اما در داده‌های جدید (آزمایشی) عملکرد ضعیفی نشان می‌دهد.

کم‌برازش زمانی رخ می‌دهد که مدل حتی در داده‌های آموزشی هم عملکرد خوبی ندارد و قادر به یادگیری الگوهای اساسی نیست.

از آنجایی که مدل بدون استفاده از normalization روی داده های آموزشی هم دقت خوبی نداشتند به نظر مید دچار کم برازش شده ایم ولی با استفاده از normalization این مشکل برطرف شده است.

سوال ۴)

سوال ۵)

الف) $g_1 = g_2 \rightarrow x = \frac{1}{2}$ ، $g_1 = g_4 \rightarrow x + y = 1$

$g_1 = g_3 \rightarrow x - y = -2$ ، $g_2 = g_4 \rightarrow x = \frac{3}{2}$

$g_1 = g_4 \rightarrow y = \frac{1}{2}$

$g_2 = g_3 = g_4 \rightarrow y = \frac{5}{2}$

ب) شکل ۱ و ۲ ← کپی گره

شکل ۳ و ۴ ← تابع نورون

ج) قسمت ۱ ←

وزن هر ورودی = 1

$Bias = -\frac{n}{2}$

تابع فعال‌سازی ← $y = g\left(\sum_{i=1}^{n} x_i - \frac{n}{2}\right)$

تابع پله ← $g(z) = \begin{cases} 1 & \text{if } z \geq 0 \\ 0 & \text{if } z < 0 \end{cases}$

a	B	y	δ
$w_{1a} = 0$	$w_1 B = 1$	$w_1 y = 0$	$w_{a\delta} = 1$
$w_{2a} = 1$	$w_2 B = 0$	$w_2 y = 1$	$w_{B\delta} = 1$
$w_a = 0$	$w_B = 4$	$w_y = -3$	$w_\delta = 2$

A	B	C
$w_{\delta A} = 1$	$w_{aB} = 1$	$w_{ac} = -1$
$w_{yA} = 1$	$w_{BB} = 1$	$w_{BC} = 1$
$w_A = -2$	$w_{yB} = -1$	$w_c = -1$
	$w_B = 1$	