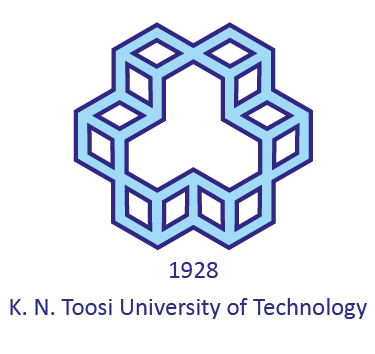
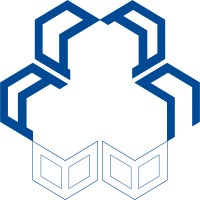
به نام خدا





دانشگاه صنعتی خواجه نصیرالدین طوسی

دانشکده برق

**مبانی هوشمند سازی**

**میانترم**

**سجاد رجبی باغستان**

**40005393**

**استاد : جناب دکتر مهدی علیاری**

بهمن 1402

##### فهرست مطالب

|  |  |
| --- | --- |
| عنوان | شماره صفحه |

[بخش۱: سوالات تحلیلی 2](#_Toc128223055)

[سوالات هماهنگ شده 3](#_Toc128223056)

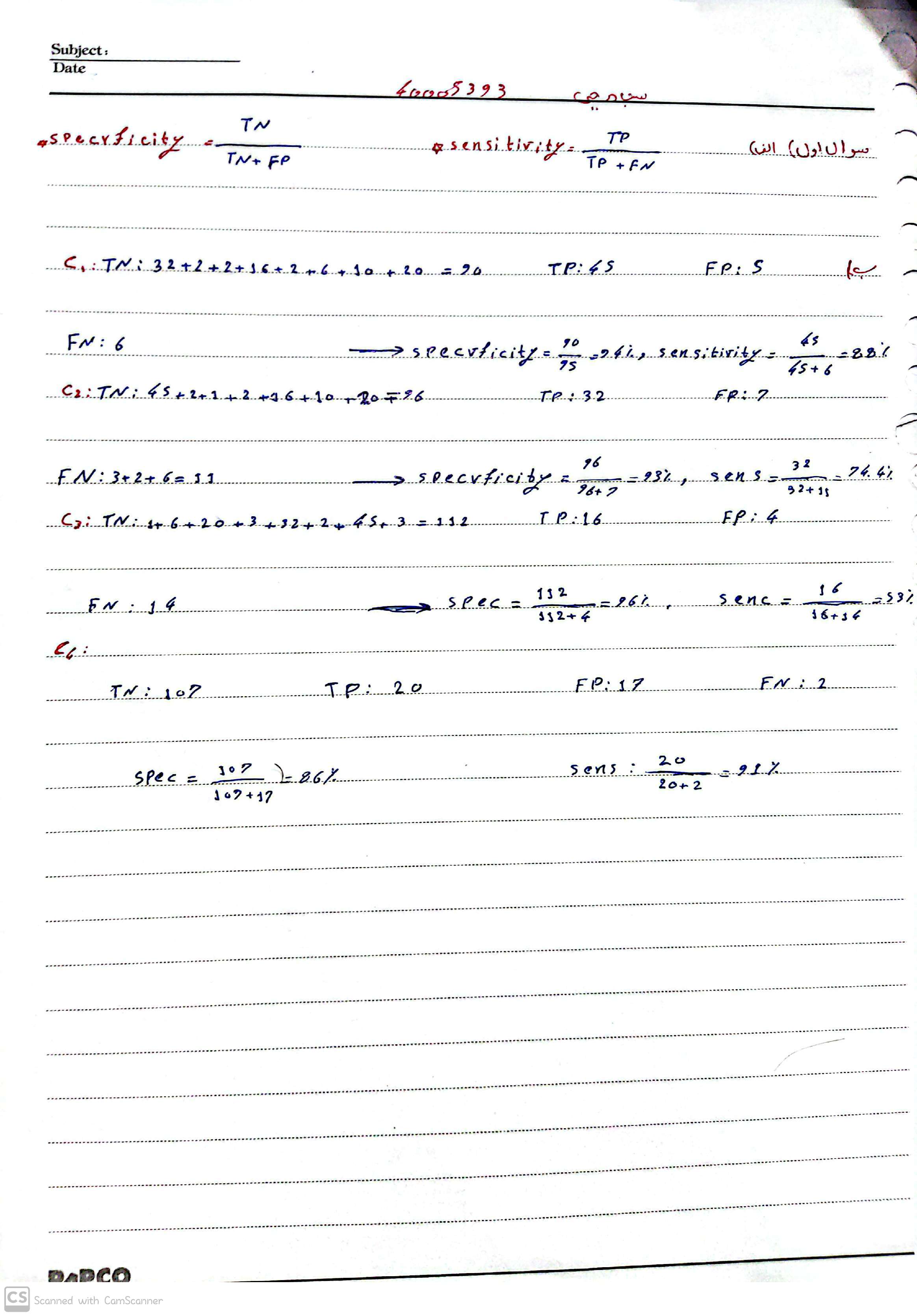
[سوالات هماهنگ نشده 5](#_Toc128223056)

[بخش۲: سوالات شبیه سازی 11](#_Toc128223058)

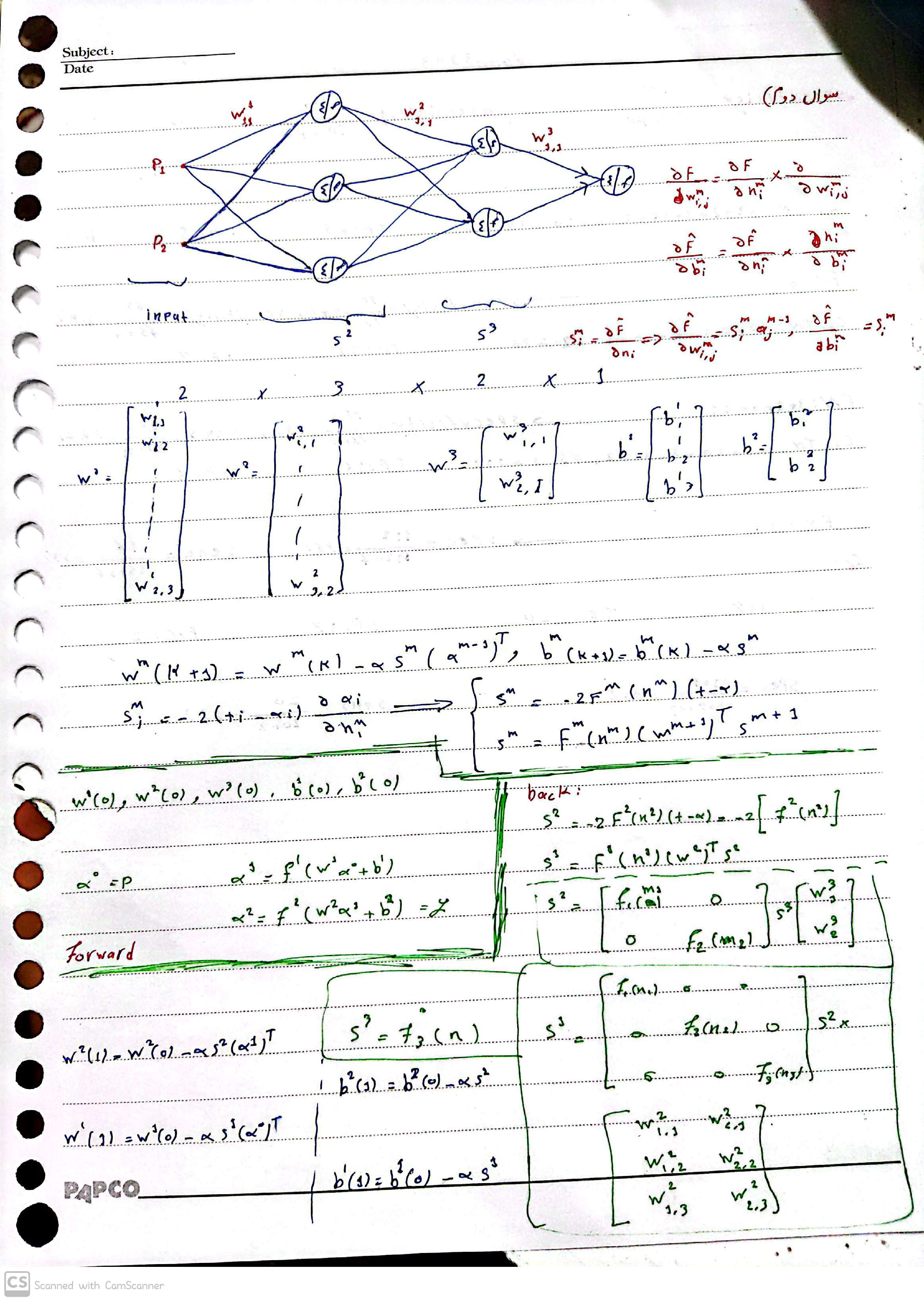
[سوال اول 11](#_Toc128223059)

# بخش۱: سوالات تحلیلی

## سوال اول



## سوال دوم



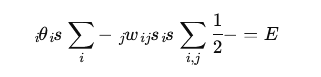
## سوال سوم

الف.

در یک شبکه هافیلد، اگر نشانه‌های گره‌ها معکوس شوند (یعنی علامت‌های مثبت به منفی و منفی به مثبت تغییر کنند)، وضعیت یا الگوی جدید همچنان همان الگوی اصلی باقی می‌ماند. دلایل این امر عبارتند از:

### ****تقارن وزن‌ها در شبکه هافیلد****

در شبکه‌های هافیلد، وزن‌های بین گره‌ها (wij​) تقارن دارند، به این معنا که wij​=wji​) ) .این تقارن باعث می‌شود که جهت تغییر نشانه گره‌ها (از مثبت به منفی یا برعکس) تأثیری بر انرژی کلی شبکه نداشته باشد، زیرا انرژی تعریف‌شده در شبکه هافیلد به صورت زیر است:



اگر نشانه تمام si​‌ها معکوس شود (si→−sis \_ si​→−si​)، جای‌گذاری این تغییر در فرمول انرژی منجر به یک تغییر علامت کلی می‌شود که تاثیری بر دینامیک انرژی ندارد. بنابراین، الگوی پایدار همچنان حفظ می‌شود.

### 2. ****ماهیت دوتایی گره‌ها****

گره‌های شبکه هافیلد دو حالت ممکن دارند یا+1)یا (−1، و رفتار دینامیکی شبکه بر پایه همبستگی میان گره‌هاست. معکوس کردن نشانه گره‌ها معادل معکوس کردن کل بردار حالت است، که تأثیری بر همبستگی‌ها و ساختار ذخیره‌سازی حافظه شبکه ندارد. به زبان دیگر، حالت پایدار شبکه (الگوهای حافظه ذخیره‌شده) نسبت به تغییر همزمان تمام نشانه‌ها (یک وارونگی کلی) ناوردا است.

ب.

### ****شبکه هافیلد****

#### **مزایا**

1. **پایداری و قابلیت ذخیره‌سازی حافظه:**
   * شبکه هافیلد می‌تواند الگوهای باینری را به صورت پایدار ذخیره کند و به عنوان یک حافظه انجمنی عمل کند.
   * مثال: اگر مجموعه‌ای از تصاویر باینری ساده (مانند حروف یا ارقام) در شبکه ذخیره شود، شبکه می‌تواند یک تصویر نویزی یا ناقص را به حالت کامل و ذخیره‌شده نزدیک کند.
2. **ساده بودن طراحی:**
   * قواعد به‌روزرسانی و ساختار شبکه هافیلد ساده هستند و می‌توانند با استفاده از قوانین انرژی طراحی شوند.

#### **معایب**

1. **ظرفیت محدود حافظه:**
   * تعداد الگوهایی که شبکه می‌تواند ذخیره کند بسیار محدود است .
   * مثال: یک شبکه با 100 گره می‌تواند فقط حدود 15 الگو را با موفقیت ذخیره کند.
2. **پایداری محدود و خطای بازخوانی:**
   * اگر تعداد الگوها زیاد شود یا داده‌ها همپوشانی زیادی داشته باشند، شبکه ممکن است الگوهای نادرست تولید کند.
3. **نیاز به انرژی زیاد در شبکه‌های بزرگ:**
   * زمان همگرایی به حالت پایدار ممکن است در شبکه‌های بزرگ افزایش یابد.

### ****رگرسیون لجستیک****

#### **مزایا**

1. **سادگی محاسبات:**
   * مدل رگرسیون لجستیک خطی بوده و به راحتی قابل پیاده‌سازی است. از آن برای پیش‌بینی احتمال یا دسته‌بندی استفاده می‌شود.
2. **تفسیرپذیری بالا:**
   * پارامترهای مدل نشان‌دهنده تأثیر هر ویژگی بر خروجی هستند که تفسیر نتایج را ساده می‌کند.
   * مثال: در یک مدل پیش‌بینی بیماری، ضرایب رگرسیون لجستیک می‌توانند تأثیر ویژگی‌هایی مانند سن، وزن و سابقه بیماری را نشان دهند.
3. **کاربرد گسترده:**
   * برای مسائل باینری و چندکلاسه مناسب است.

#### **معایب**

1. **عدم کارایی برای روابط غیرخطی:**
   * اگر رابطه بین ویژگی‌ها و خروجی پیچیده و غیرخطی باشد، رگرسیون لجستیک نمی‌تواند عملکرد خوبی داشته باشد.
2. **حساسیت به هم‌خطی:**
   * در صورت وجود هم‌خطی شدید بین متغیرهای ورودی، عملکرد مدل کاهش می‌یابد.

### ****مقایسه با یک مثال مشترک****

فرض کنید می‌خواهیم تشخیص دهیم که آیا یک ایمیل اسپم است یا خیر:

* **شبکه هافیلد:**
  + ایمیل‌ها به صورت الگوهای باینری ذخیره می‌شوند و شبکه می‌تواند ایمیل‌های مشابه را با یک ایمیل ذخیره‌شده تشخیص دهد.
  + محدودیت: ظرفیت محدود شبکه هافیلد ممکن است ذخیره تعداد زیادی الگو را دشوار کند.
* **رگرسیون لجستیک:**
  + از ویژگی‌هایی مانند تعداد کلمات کلیدی، وجود پیوست‌ها، و فرمت ایمیل برای پیش‌بینی احتمال اسپم بودن استفاده می‌شود.
  + محدودیت: اگر رابطه بین ویژگی‌ها و احتمال اسپم بودن پیچیده باشد، ممکن است دقت کاهش یابد.

# بخش۲: سوالات شبیه سازی

## سوال اول

لینک درایو :

<https://drive.google.com/file/d/1lpZ9OA-NCF6A6P8yzd_rKQioPSiF2h1y/view?usp=sharing>

بخش الف، ب، ج، د)

ساخت یک شبکه عصبی ساده که وظیفه‌اش تشخیص یک ناحیه خاص بر اساس شرایط خاص است. وزن‌ها و بایاس‌ها به‌طور دستی تنظیم شده‌اند تا شبکه دقیقاً این نواحی را شبیه‌سازی کند. روند محاسبه وزن‌ها و بایاس‌ها در این شبکه عصبی به صورت زیر انجام شده است:

### 1. ****تعریف شرایط اولیه****

اولین قدم این است که شما دو شرط (condition1) و (condition2) را به‌صورت منطقی تعریف کنید. این شرایط برای تعیین نواحی خاص در فضای x و y هستند که در آن‌ها شبکه باید فعال شود.

* **شرط اول (condition1)**: ناحیه‌ای که بین ۲ و ۶ در محور x و بین ۲ و ۶ در محور y قرار دارد.
* **شرط دوم (condition2)**: ناحیه‌ای که در آن 4≤x≤6 و y ≤x+2 برای y بین ۶ و ۸ است.

**.2شبیه‌سازی شرایط با وزن‌ها و بایاس‌ها**

در اینجا شما باید با استفاده از یک شبکه عصبی چند لایه، شرایط منطقی را به شبکه منتقل بکنید. برای هر لایه، وزن‌ها و بایاس‌ها به‌طور دستی تنظیم شود تا شبکه درست عمل کند.

#### لایه اول:

در لایه اول، پنج نورون تعریف شده است که هرکدام به‌طور جداگانه یک شرط خاص را شبیه‌سازی می‌کنند. این نورون‌ها می‌توانند دربردارنده بررسی شرایط برای x و y باشند.

* برای بررسی اینکه آیا x بین ۲ و ۶ است، از دو نورون استفاده می شود:
  + یکی برای ) x≥2 با وزن مثبت و بایاس منفی (
  + دیگری برای x≤6 (با وزن منفی و بایاس مثبت(
* مشابه همین روش برای بررسی شرایط y در 2≤y≤6 استفاده می‌شود.
* و در نهایت، نورن پنجم برای بررسی اینکه y≤x+2 (برای شرایط شرط دوم) استفاده می‌شود.

#### لایه دوم:

در لایه دوم، دو نورون .ج.د دار که به‌طور منطقی دو شرط مختلف را با یکدیگر ترکیب می‌کنند:

* نورون اول: ترکیب‌کردن شرط اول (با استفاده از چهار ورودی که نتیجه‌های چهار شرط اول از لایه اول هستند).
* نورون دوم: ترکیب‌کردن شرط دوم.

این نورون‌ها با استفاده از عملیات منطقی AND (با بایاس‌ها و وزن‌ها) نتایج نهایی را تولید می‌کنند.

#### لایه خروجی:

در لایه آخر، برای ترکیب شرایط AND دو لایه قبلی از یک نورون استفاده میشود. این نورون از ورودی‌های لایه دوم استفاده می‌کند و نتیجه نهایی را از طریق یک عملیات OR تولید می‌کند.

### 3. ****فرآیند پیش‌بینی****

برای پیش‌بینی، داده‌ها از طریق شبکه عبور داده می‌شوند:

1. ورودی‌ها به لایه اول وارد می‌شوند و برای هر نورون در این لایه، محاسبه می‌شود که آیا شرط مربوطه برآورده می‌شود یا نه.
2. سپس، خروجی لایه اول وارد لایه دوم می‌شود و این لایه ترکیب شرایط مختلف را بررسی می‌کند.
3. در نهایت، لایه خروجی نتیجه نهایی را به‌عنوان پیش‌بینی نهایی می‌دهد.

### 4. ****توابع فعال‌سازی****

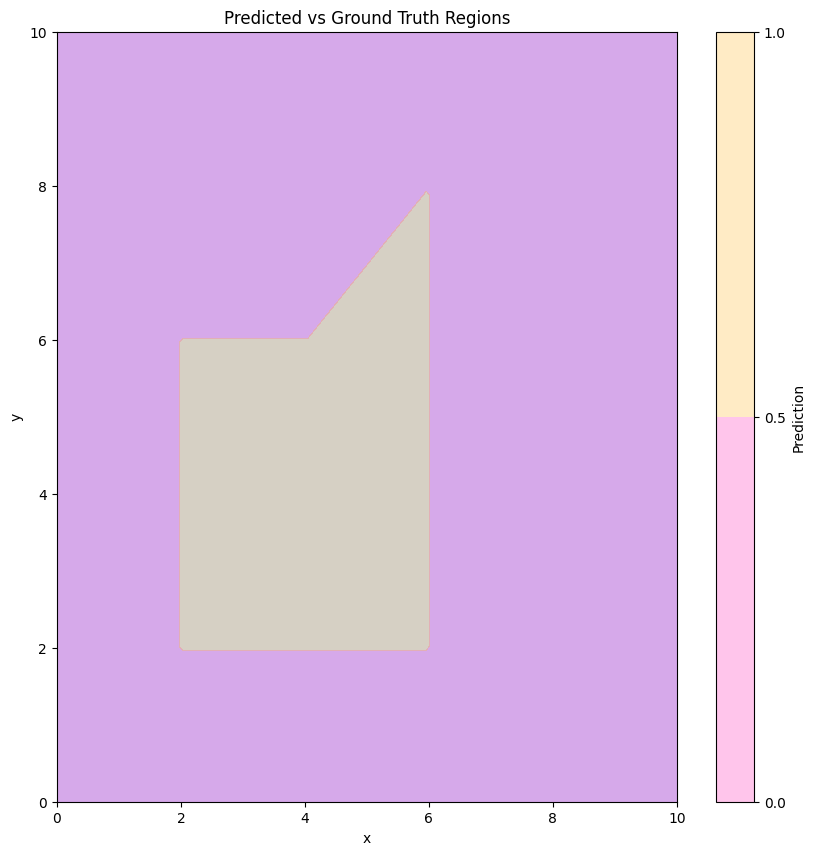
برای تمام نورون‌ها، از تابع فعال‌سازی **ReLU**  استفاده شده است که مقادیر منفی را صفر می‌کند و مقادیر مثبت را حفظ می‌کند. این کمک می‌کند تا شبکه از صفرهای غیر ضروری خلاص شود.

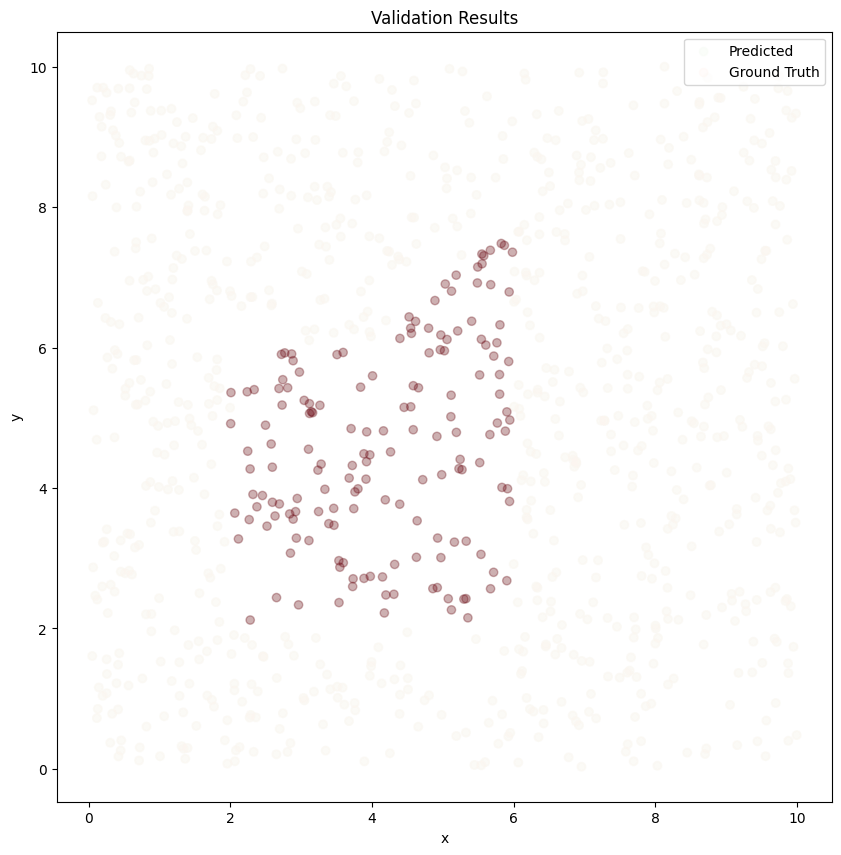
در لایه خروجی، از تابع **Step** استفاده شده است که فقط اگر خروجی بالاتر از صفر باشد، نتیجه ۱ (فعال) را تولید می‌کند و در غیر این صورت ۰ (غیرفعال).

### 5. ****تعیین وزن‌ها و بایاس‌ها****

برای تعیین وزن‌ها و بایاس‌ها به‌صورت دستی، از اطلاعات هندسی مربوط به شرایط استفاده می شود. به‌طور خاص:

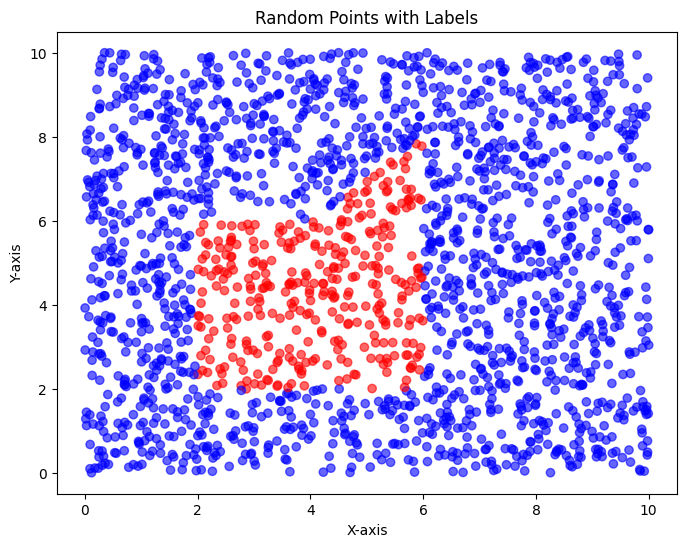
* **وزن‌ها (Weights)**: مقادیری که در معادلات خطی برای نورون‌ها استفاده می‌شوند. این وزن‌ها تعیین می‌کنند که هر ورودی چقدر بر خروجی تاثیر می‌گذارد. در اینجا، از وزن‌های مثبت و منفی برای شبیه‌سازی شرایط خاص استفاده کرده‌اید.
* **بایاس‌ها (Biases)**: بایاس‌ها برای تنظیم میزان آستانه‌ای که یک نورون برای فعال شدن نیاز دارد استفاده می‌شود. این بایاس‌ها کمک می‌کنند تا شرایط خاص را به‌درستی برآورده کنید.





نمودار ای بالا نشان دهنده خروجی شبکه دستی هستند که مشخص میکند شبکه به درست کار میکند.

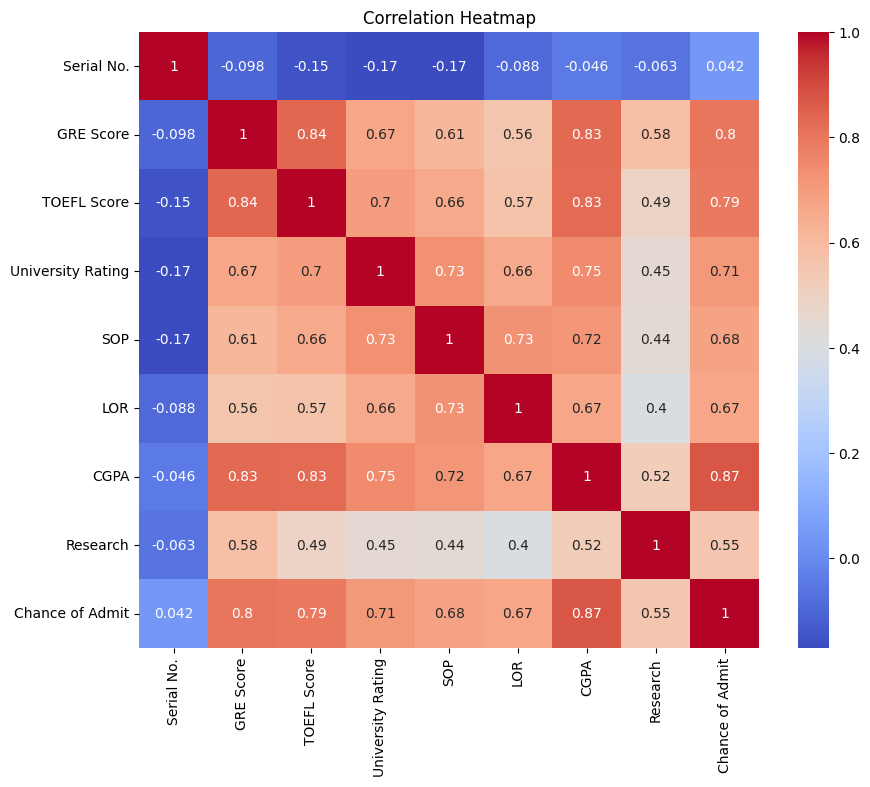
بخش ه)



ساخت دیتاست جهت آموزش مدل mlp .

## سوال دوم

الف.



Feature correlations with target (descending):

Chance of Admit 1.000000

CGPA 0.873289

GRE Score 0.802610

TOEFL Score 0.791594

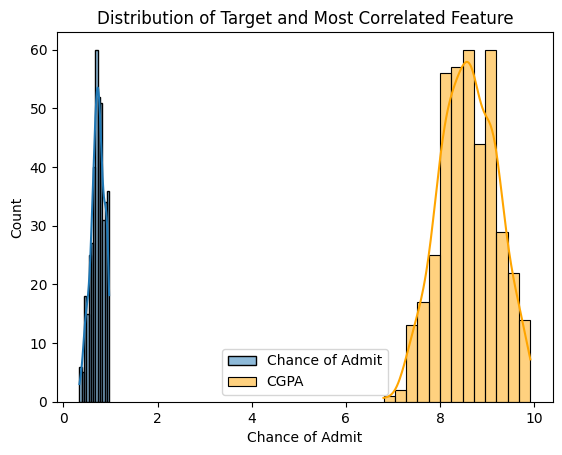
University Rating 0.711250

SOP 0.675732

LOR 0.669889

Research 0.553202

Serial No. 0.042336



نمودار **هیت‌مپ همبستگی** (Correlation Heatmap) اطلاعات ارزشمندی درباره رابطه بین ویژگی‌ها و هدف فراهم می‌کند. تحلیل آن می‌تواند کمک کند تا بفهمید کدام ویژگی‌ها بیشترین تاثیر را بر متغیر هدف دارند و کدام ویژگی‌ها ممکن است برای مدل مفید یا غیرضروری باشند.

### نکاتی که می‌توان از هیت‌مپ استنباط کرد:

1. **ویژگی‌های با همبستگی بالا با هدف:** اگر یک ویژگی همبستگی بالایی (مثلاً بالای ۰.۷ یا ۰.۸) با متغیر هدف (مانند "شانس پذیرش") داشته باشد، این نشان‌دهنده تأثیر بالای آن ویژگی در پیش‌بینی هدف است. مثلاً، اگر معدل (GPA) یا نمره آزمون (GRE) همبستگی بالایی با "شانس پذیرش" داشته باشد، به احتمال زیاد این ویژگی‌ها اطلاعات کلیدی برای مدل خواهند بود.
2. **ویژگی‌های با همبستگی کم با هدف:** ویژگی‌هایی که همبستگی کمی (مثلاً کمتر از ۰.۲) با متغیر هدف دارند، ممکن است برای مدل اهمیت کمتری داشته باشند. این ویژگی‌ها را می‌توان به دقت بررسی کرد و حتی ممکن است بتوان آنها را از مدل حذف کرد.
3. **همبستگی بالا بین ویژگی‌ها:** اگر دو یا چند ویژگی همبستگی بسیار بالایی با یکدیگر داشته باشند (بالاتر از ۰.۸ یا ۰.۹)، احتمالاً اطلاعات مشابهی را به مدل ارائه می‌دهند. این امر می‌تواند منجر به **هم‌خطی** (Multicollinearity) شود که بر عملکرد مدل تأثیر منفی می‌گذارد. در این حالت، ممکن است بخواهید یکی از این ویژگی‌ها را حذف کنید.
4. **پراکندگی داده‌ها:** نمودار هیت‌مپ می‌تواند به شما ایده‌ای درباره پراکندگی همبستگی‌ها بدهد. اگر همه ویژگی‌ها همبستگی پایینی با هدف داشته باشند، ممکن است پیش‌بینی سخت‌تر باشد، اما اگر چند ویژگی همبستگی بالایی داشته باشند، مدل شما احتمالاً بهتر عمل خواهد کرد.

### برای مثال در این داده‌ها:

* **ویژگی‌هایی مانند GRE و GPA:** معمولاً همبستگی بالایی با "شانس پذیرش" دارند، چون نشان‌دهنده عملکرد تحصیلی دانشجو هستند.
* **تجربه تحقیقاتی (Research Experience):** می‌تواند به عنوان یک ویژگی باینری (0 یا 1) اثر مستقیم یا غیرمستقیم بر پذیرش داشته باشد.
* **ویژگی‌های کیفیت (SOP و LOR):** ممکن است همبستگی کمتری داشته باشند، زیرا این معیارها معمولاً ذهنی‌تر ارزیابی می‌شوند.

ب .

### ****انتخاب روش پیش‌پردازش:****

#### - **استانداردسازی (Standardization):**

استانداردسازی مقادیر هر ویژگی را طوری تغییر می‌دهد که میانگین صفر و واریانس یک داشته باشد. این روش زمانی مناسب است که داده‌ها دارای توزیع نرمال یا نزدیک به نرمال باشند و اندازه مقادیر متفاوت در مقایسه ویژگی‌ها اهمیت داشته باشد.

#### - **نرمال‌سازی (Normalization):**

نرمال‌سازی مقادیر را به یک بازه مشخص (معمولاً بین ۰ و ۱) تبدیل می‌کند. این روش زمانی مناسب است که داده‌ها در بازه‌های مختلف و بسیار پراکنده باشند یا مدل‌هایی که حساس به دامنه مقادیر هستند (مانند شبکه‌های عصبی).

#### **انتخاب روش:**

با توجه به داده‌های مورد استفاده( مانند GRE، GPA):

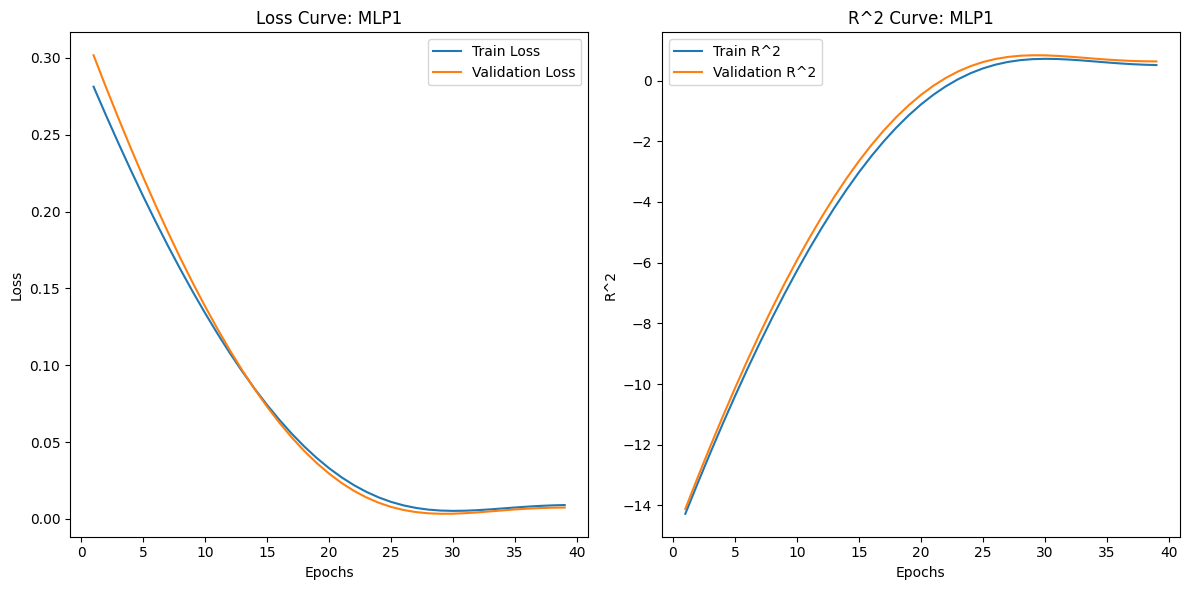
* مقادیر ویژگی‌ها در بازه‌های متفاوتی قرار دارند (مثلاً GPA در بازه ۰ تا ۱۰ و GRE در بازه ۰ تا ۳۴۰). بنابراین **نرمال‌سازی** روش مناسب‌تری است.

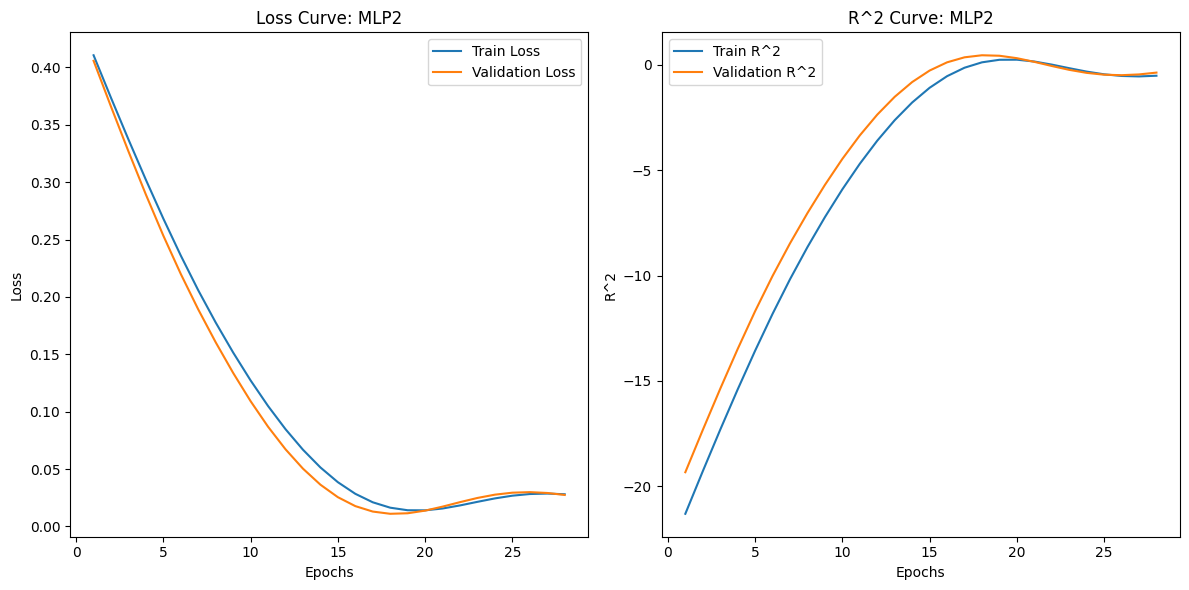
### ****آیا باید داده‌های مجموعه آزمون را در پیش‌پردازش لحاظ کنیم؟****

خیر، داده‌های مجموعه آزمون نباید در محاسبات نرمال‌سازی یا استانداردسازی شرکت کنند. دلیل این موضوع:

* مدل شما نباید اطلاعاتی از داده‌های آزمون داشته باشد، چرا که این داده‌ها شبیه‌سازی شرایط واقعی هستند و استفاده از آنها در فرآیند آموزش یا پیش‌پردازش باعث **نشت داده** می‌شود.

ج.





Epoch 10: Train Loss = 0.13381744921207428, Val Loss = 0.13818477094173431, Train R^2 = -6.2680, Val R^2 = -5.9228

Epoch 20: Train Loss = 0.0329534187912941, Val Loss = 0.029419057071208954, Train R^2 = -0.7898, Val R^2 = -0.4738

Epoch 30: Train Loss = 0.005217362195253372, Val Loss = 0.003376180538907647, Train R^2 = 0.7166, Val R^2 = 0.8309

Early stopping at epoch 39

Epoch 10: Train Loss = 0.12720416486263275, Val Loss = 0.10911460220813751, Train R^2 = -5.9088, Val R^2 = -4.4664

Epoch 20: Train Loss = 0.014027001336216927, Val Loss = 0.013772852718830109, Train R^2 = 0.2382, Val R^2 = 0.3100

Early stopping at epoch 28

### نتایج مدل‌ها:

#### **مدل MLP1:**

1. **Loss و R² در طول زمان:**
   * **Epoch 10:**
     + Train Loss: 0.1338
     + Val Loss: 0.1382
     + Train R²: -6.2680
     + Val R²: -5.9228
   * **Epoch 20:**
     + Train Loss: 0.0330
     + Val Loss: 0.0294
     + Train R²: -0.7898
     + Val R²: -0.4738
   * **Epoch 30:**
     + Train Loss: 0.0052
     + Val Loss: 0.0034
     + Train R²: 0.7166
     + Val R²: 0.8309
2. **توقف زودهنگام:**  
   مدل در **Epoch 39** به دلیل نوسان کم در بهبود Loss متوقف شده است.
3. **تحلیل:**
   * در ابتدا، Loss بالا بوده و R2R^2R2 مقدار منفی نشان می‌دهد که مدل عملکرد بسیار ضعیفی دارد.
   * پس از چند Epoch، مدل شروع به یادگیری کرده و R2R^2R2 در نهایت به مقادیر مثبت رسیده است (Train R² = 0.7166، Val R² = 0.8309).
   * عملکرد نهایی روی داده‌های اعتبارسنجی بهتر از داده‌های آموزشی است که نشان‌دهنده تعمیم خوب مدل است.

#### **مدل MLP2:**

1. **Loss و R² در طول زمان:**
   * **Epoch 10:**
     + Train Loss: 0.1272
     + Val Loss: 0.1091
     + Train R²: -5.9088
     + Val R²: -4.4664
   * **Epoch 20:**
     + Train Loss: 0.0140
     + Val Loss: 0.0138
     + Train R²: 0.2382
     + Val R²: 0.3100
2. **توقف زودهنگام:**  
   مدل در **Epoch 28** به دلیل نوسان کم در بهبود Loss متوقف شده است.
3. **تحلیل:**
   * این مدل نسبت به MLP1 در مراحل ابتدایی (Epoch 10) عملکرد مشابه یا کمی بهتر دارد.
   * در Epoch 20، مدل پیشرفت قابل‌توجهی داشته است، اما همچنان R2R^2R2 بسیار پایین‌تر از مدل اول است.
   * توقف زودهنگام در Epoch 28 نشان می‌دهد که این مدل به سرعت به نقطه اشباع رسیده و پتانسیل کمتری برای یادگیری داشته است.



#### تحلیل مقایسه‌ای:

* **MLP1:**  
  این مدل عملکرد بهتری در هر دو معیار Loss و R2R^2R2 داشته است. پیشرفت پایدارتر و تعمیم بهتر روی داده‌های اعتبارسنجی از ویژگی‌های بارز این مدل است.
* **MLP2:**  
  مدل دوم در اوایل آموزش سریع‌تر بهبود پیدا کرده، اما توانایی کمتری در یادگیری عمیق‌تر و تعمیم به داده‌های جدید داشته است.

د.

### تحلیل عملکرد مدل:

1. **دقت پیش‌بینی‌ها برای نمونه‌های تصادفی:**
   * میانگین دقت پیش‌بینی برای نمونه‌های تصادفی بین 86.66% تا 99.62% متغیر است.
   * **میانگین خطای مطلق (MAE):** مقدار بسیار کوچک 0.028 نشان‌دهنده عملکرد مناسب در خطای مطلق است.
   * با وجود پیش‌بینی‌های نزدیک به مقادیر واقعی، اختلاف نسبی در بعضی نمونه‌ها (مانند Sample 4 با دقت 86.66%) نشان‌دهنده وجود فضای بهبود است.
2. **عملکرد کلی مدل روی داده‌های آزمون:**
   * **R2R^2R2:** مقدار 0.7769 نشان می‌دهد که مدل حدود 77.69% از واریانس داده‌ها را توضیح می‌دهد.
   * اگرچه این مقدار خوب است، برای مسائل حساس مانند پیش‌بینی شانس پذیرش ممکن است دقت بیشتری موردنیاز باشد.

### آیا این عملکرد مناسب است؟

* **مناسب بودن یا نبودن مدل به زمینه کاربرد بستگی دارد.** اگر نیازمند پیش‌بینی‌های بسیار دقیق هستید (مثلاً برای تصمیم‌گیری‌های حساس)، این مدل ممکن است نیاز به بهبود داشته باشد.
* اگر به دنبال یک مدل ابتدایی و سریع برای تحلیل داده هستید، این مدل عملکرد مناسبی دارد.

### پیشنهاداتی برای بهبود عملکرد:

1. **افزایش پیچیدگی مدل:**
   * افزایش تعداد لایه‌ها یا نورون‌های مخفی در شبکه عصبی.
   * استفاده از معماری‌های پیشرفته‌تر مانند شبکه‌های Residual یا Transformers.
2. **افزایش داده‌ها:**
   * اگر داده‌های بیشتری در دسترس دارید، اضافه کردن آن‌ها به مدل می‌تواند دقت را بهبود دهد.
   * استفاده از تکنیک‌های افزایش داده (Data Augmentation) در صورتی که داده‌های محدودی دارید.
3. **بهینه‌سازی هایپرپارامترها:**
   * تنظیمات دقیق‌تر برای نرخ یادگیری، تعداد نورون‌ها و لایه‌ها.
   * استفاده از ابزارهایی مانند **Grid Search** یا **Bayesian Optimization**.
4. **انتخاب ویژگی‌ها (Feature Selection):**
   * بررسی تأثیر ویژگی‌های مختلف روی عملکرد مدل و حذف ویژگی‌های کم‌اهمیت.
   * استفاده از تکنیک‌هایی مانند PCA (تحلیل مؤلفه‌های اصلی) برای کاهش ابعاد.
5. **استفاده از داده‌های متنوع‌تر:**
   * اگر توزیع داده‌های آموزشی و آزمون متفاوت است، این مسئله را با داده‌برداری مجدد یا جمع‌آوری داده‌های متنوع برطرف کنید.
6. **آزمون مدل‌های دیگر:**
   * مقایسه با مدل‌های غیرشبکه‌ای مانند **XGBoost**، **Random Forest** یا **Linear Regression**.
   * ممکن است مدل‌های ساده‌تر برای این مجموعه داده مناسب‌تر باشند.
7. **بهبود معیار هزینه (Loss Function):**
   * استفاده از معیار هزینه مناسب‌تر برای کاهش خطاها، مانند **Huber Loss** یا **Mean Absolute Percentage Error (MAPE)**.
8. **نرمال‌سازی دقیق‌تر:**
   * بررسی دقیق‌تر نرمال‌سازی و استانداردسازی داده‌ها و تست چندین روش مختلف.