

Sat. Sun. Mon. Tue. Thu. Wed. Fri.

Subject: -----

به نام خدا

تقریبات : بخش اول

استاد : آقای محمد احمدزاده

درس : صباحت و پرده

اعضای گروه : سید احمد بخاری ، سید احمد علی

Subject:

Year, Month, Date

Sa Su Mo Tu We Th Fr

۱- **unsupervised Learning** تفاوتی دارند؟ تفاوت این بین  
**unsupervised Learning** در این است که مشکل در ساخت مدل است در حالی  
 تفاوت این بین **underfitting**  
**unsupervised Learning** نوع یادگیری ماشین است. بیاید این  
 تفاوت را دقیق تر بررسی کنیم

خاصیت: یک مشکل یا وضعیت نامطلوب در یادگیری ماشین است

علت: مدل بسیار ساده است (نی توان از الگوی های موجود در داده ها را یاد بگیرد)

تتبع: عملکرد ضعیف در داده های آموزشی و همچنین در داده های

رابط: استفاده از مدل های پیچیده تر اضافه کردن ویژگی های بیشتر یا افزایش زمان آموزش

**unsupervised Learning**

خاصیت: یک رویکرد در یادگیری ماشین است

هدف: کشف الگوها ساخت روابط بین داده های برچسب دار

(labeled data)

کاربردها: خوشه بندی (clustering) ابعاد (dimensionality reduction)

به عبارت دیگر **underfitting** یک مشکل است که می تواند در هر نوع یادگیری ماشین از جمله  
**supervised** و روش خاص برای یادگیری از داده ها است در حالی که یک حالت  
 که در آن مدل شما به اندازه کافی خوب عمل نمی کند

Subject:

Year: \_\_\_\_\_ Month: \_\_\_\_\_ Date: \_\_\_\_\_

Sa	Su	Mo	Tu	We	Th	Fr
----	----	----	----	----	----	----

۲- چرا ضروری است Feature Scaling در الگوریتم های Machine Learning

Feature Scaling

Machine Learning به دلیل زیر ضروری است!  
معماریت الگوریتم ها به مقیاس بسیار از الگوریتم های یادگیری ماشین به خصوص الگوریتم

های که از فاصله (Distance) برای اندازه گیری شباهت یا تفاوت بین داده ها استفاده

می کنند مانند الگوریتم های (کمرسیون خطی) (Linear)

سرعت هدری:

الگوریتم های که از روش های مبتنی بر گرادیان (Gradient-Based) برای بسط

سازی استفاده می کنند مانند رگرسیون خطی و لجستیک با گرادیان کاهش (Gradient

کشیه های معینی در صورت عدم مقیاس بندی ویژگی ها ممکن است به سختی هدر شوند یا به

زمان بیشتر برای هدری نیاز داشته باشند.

مقیاس بندی ویژگی ها باعث می شود



Subject:

Year. Month. Date.

Sa Su Mo Tu We Th Fr

۳- Normalization و تفاوتی دارند؟

standardization

دو تکنیک هم در پیش پردازش داده های هستند برای مقیاس بندی ویژگی

های (Features) در داده های ورودی استفاده می شوند ویژگی را این تکنیک ها

کاربرد خاصی دارند و تفاوت های اساسی بین آنها وجود دارد.

۴- چرا min-max-normalization برای مقیاس بندی داده ها استفاده

می شود؟

۱- محدودیت کردن دامنه داده ها - ۲- بهبود سرعت الگوریتم ها - ۳-

۴- عزیمت کردن دامنه داده ها می تواند داده ها را به دامنه ای مشخص

محو لا بین و واسطه کن این کار می تواند به کاهش آسیب پذیری الگوریتم های

یادگیری ماشین به مقدار بی افراطی (outliers) کمک کند

۲- بهبود سرعت الگوریتم ها: با مقیاس بندی داده ها الگوریتم های یادگیری

ماشین معمولاً سریع تر و بهتر عمل می کنند این به ویژه در الگوریتم های محاسبات

به مقیاس مانند (K-NN) K-Nearest Neighbors هم است

Subject:

Year. Month. Date.

Sa	Su	Mo	Tu	We	Th	Fr
----	----	----	----	----	----	----

## Z-score Normalization ۵۴

یا نرمال سازی Z-score روشی برای تبدیل داده‌ها به توزیع میانگین صفر و انحراف صیارک است. در این روش هر مقدار داده با کم کردن میانگین از آن تقسیم بر انحراف صیارک نرمال می‌شود. فرمول آن به این صورت است:

$$Z = \frac{(x - \mu)}{\sigma}$$

که در آن:

$Z$ : مقدار نرمال شده Z-score       $x$ : مقدار اصلی داده

$\mu$ : میانگین داده‌ها       $\sigma$ : انحراف صیارک داده‌ها

## Z-score Normalization که مبره‌های فراوانی دارد از جمله

مقایسه داده‌ها با مقیاس‌های مختلف: وقتی داده‌ها با مقیاس‌های متفاوت (مثلاً وزن و قد) در یک فضای قرار می‌گیرند، به طور مستقیم آن‌ها را مقایسه کنید Z-score Normalization این مشکل را حل می‌کند و همه داده‌ها را به یک مقیاس واحد می‌آورد.

بهبود عملکرد الگوریتم‌های یادگیری ماشین: بسیاری از الگوریتم‌های یادگیری ماشین (مثل شبکه‌های عصبی و درخت تصمیم) حساس به مقیاس داده‌ها هستند. نرمال‌سازی داده‌ها باعث می‌شود الگوریتم‌ها سریع‌تر و دقیق‌تر عمل کنند و در نتیجه دقت آن‌ها افزایش یابد. این به ویژه زمانی اهمیت پیدا می‌کند که متغیرها مقیاس‌های بسیار متفاوتی داشته باشند.

شناسایی نقاط پرت (outliers): با استفاده از Z-score می‌توان به راحتی نقاط پرت را شناسایی کرد. مقدار Z-score با ۳ یا بیشتر از ۳ (مثلاً بیشتر از ۳ یا کمتر از -۳) احتمالاً احتمالاً نقاط پرت هستند.

استاندارد سازی داده‌ها: Z-score با این روش می‌تواند که توزیع داده‌ها به توزیع استاندارد نرمال (با میانگین ۰ و انحراف صیارک ۱) نزدیک شود. این ویژگی برای تحلیل‌های آماری و مقایسه‌ها بسیار مفید است. تقسیم ساده تر نتایج: با نرمال‌سازی داده‌ها، مقایسه نتایج آسان‌تر می‌شود زیرا همه متغیرها در یک مقیاس قرار دارند.

در کل Z-score Normalization یک تکنیک ساده و کاربردی جهت نرمال‌سازی داده‌ها و تحلیل آماری است که به بهبود دقت و کارایی مدل‌های یادگیری ماشین و همچنین تفسیر ساده تر نتایج کمک می‌کند.



## Regularization در الگوریتم های Machine Learning چیست؟

منظوم سازی (Regularization) در یادگیری ماشین دو هدف اصلی است: جلوگیری از بیش برازش (overfitting) مدل و پیش برداشتن از داده های جدید. داده های آموزشی را پیش از حد دقیق یاد می گیریم و در نتیجه در پیش بینی داده های جدید (داده های دیده نشده) عملکرد ضعیفی از خود نشان می دهد. مدل در واقع نویز و جزئیات بی اهمیت در داده های آموزشی را نیز یاد می گیرد که در داده های واقعی وجود ندارند. منظور از بیش برازش (overfitting) این است که مدل بیش از حد به ویژگی های خاص داده های آموزشی (Penalty) به نفع هزینه مدل از پیچیدگی بیش از حد مدل جلوگیری می کند. این هزینه برآوردی از وزن های مدل را کنترل می کند و وزن های بزرگ تر نشان دهنده مدل پیچیده تر و مستعد تر به بیش برازش هستند. در روش اصلی منظوم سازی عبارتند از:

**L1 Regularization (Lasso):** در این روش هزینه ای اضافه شده به تابع هزینه برای به هم چسبیدن وزن ها است. این روش تمایل دارد برخی از وزن ها را به صفر برساند که منجر به انتخاب ویژگی (feature selection) می شود به عبارت دیگر ویژگی های کم اهمیت تر را حذف می کند.

**L2 Regularization (Ridge):** این روش هزینه ای اضافه شده به تابع هزینه برای به هم چسبیدن وزن ها است. این روش وزن های را به سمت صفر کوچک می کند تا مدل منظوم سازی.

گاهی بیش برازش را کنترل وزن ها از پیچیدگی بیش از حد مدل جلوگیری می شود. به عبارت دیگر، داده های دیده نشده: مدل منظوم سازی عملکرد بهتری در پیش بینی داده های جدید دارد.

افزایش تعمیم پذیری: مدل قابلیت تعمیم پذیری بیشتری به داده های جدید پیدا می کند. انتخاب بین L1 و L2: انتخاب بین L1 و L2 به نوع داده ها و مسئله مورد مطالعه بستگی دارد. L1 معمولاً برای انتخاب ویژگی و L2 برای کاهش بیش برازش در مواردی که بسیاری از ویژگی ها در L2 برای کاهش بیش برازش در مواردی که بسیاری از ویژگی ها اهمیت دارند ترجیح داده می شود همچنین می توان از ترکیب هر دو روش (Elastic Net) نیز استفاده کرد.

Subject:

Year.

Month.

Date.

Sa Su Mo Tu We Th Fr

~~Underfitting~~ Underfitting, Overfitting  
Model building به وجود می آورند:

در مدل سازی اورفیتینگ (overfitting) و اندرفیتینگ (underfitting) دو مشکل رایج هستند که می توان به دقت عملکرد مدل را به شدت تحت تاثیر قرار دهند اورفیتینگ (overfitting):

اورفیتینگ زمانی اتفاق می افتد که مدل بیش از حد به داده های آموزشی حساسیت داشته باشد و الگوهای تعدادی و غیر معمولی در داده های آموزشی را به عنوان الگوهای کلی در نظر می گیرد. این یعنی مدل با داده های آموزشی بسیار خوب عمل می کند اما در مواجهه با داده های جدید (Test Data) که در ~~مجموعه~~ مجموعه آموزشی وجود نداشته اند عملکرد ضعیفی از خود نشان می دهد به عبارت دیگر مدل توانایی تعمیم به داده های جدید را ندارد مشکلات اورفیتینگ:

عملکرد ضعیف در داده های جدید: این اصلی ترین مشکل اورفیتینگ است مدل در داده های آموزشی عملکرد عالی دارد اما در داده های جدید به شدت دقت بسیار پایینی نشان می دهد پیچیدگی بیش از حد مدل:

مدل دارای اورفیت شده اغلب بسیار پیچیده هستند و شامل تعداد زیادی پارامتر می باشند این پیچیدگی باعث می شود که مدل به جزئیات بی اهمیت داده های آموزشی توجه کند و الگوهای واقعی فقط داده های آموزشی را حفظ کرده

عدم تعمیم پذیری: مدل قادر به تعمیم یافته های خود به داده های جدید نیست و در واقع فقط داده های آموزشی را حفظ کرده است هزینه محاسباتی بالا:

مدل های پیچیده اورفیت شده اغلب نیاز به محاسبات بیشتر دارند که می تواند زمان و منابع بیشتری را مصرف کند



Subject:

Date:

cross-validation چیه؟ Train/Test split، برادر؟

Cross-validation (انتخاب، منجی، مقادیر) یک تکنیک برای ارزیابی عملکرد مدل هاست. یادگیری ماشین استفاده می شود به گونه ای که Train/Test split (تقسیم داده) به روشی آکسوس (تکرار) تکرار دارد و در اینجا دلیل اصلی استفاده از cross-validation

1 ارزیابی دقیق تر و جامع تر عملکرد مدل:

در روش Train/Test split، داده ها فقط یک بار به دو بخش آموزش و تست تقسیم می شوند. این بدان معناست که عملکرد مدل فقط بر اساس یک مجموعه داده تست خاص به طور متناوب تعیین می شود. به همین دلیل، ارزیابی عملکرد مدل ممکن است گمراه کننده باشد و نتایج دقیق به دست نیاید.

مزیت Cross-validation:

با تقسیم داده ها به چندین زیر مجموعه (fold) و تکرار فرایند آموزش و تست بر روی هر یک از این مجموعه ها، ارزیابی جامع تر از عملکرد مدل را می دهد. این کار باعث می شود که تاثیر تصادفی بودن یک مجموعه داده آزمون خاص کاهش یابد و ارزیابی دقیق تر از عملکرد مدل به دست آید.

2 انتخاب بهترین تنظیمات (Hyperparameter Tuning):

انتخاب مدل، نوعی، چندین مدل مختلف برای یک مسئله، در تنظیم cross-validation می تواند به دست آید. انتخاب مدل که به طور کلی بهترین عملکرد را در چندین زیر مجموعه داده می تواند عملکرد مدل را به حداکثر برساند و به همین دلیل، انتخاب بهترین مدل به طور کلی به دست می آید.

تنظیم پارامترها:

بسیاری از مدل های یادگیری ماشین دارای پارامترهای قابل تنظیم هستند. به عنوان مثال، در cross-validation، به جای یک بار تنظیم پارامترها، این پارامترها به گونه ای تنظیم می شوند که بهترین عملکرد را در چندین زیر مجموعه داده به دست آید.

عملکرد مدل در این فرایند Hyperparameter Tuning



## 3 استفاده بجهت از داده ها:

Cross-validation  
 کاربرد دارد: در بسیاری از پروژه های یادگیری ماشین داده ها محدود هستند  
 به طوری که نمی توان از تمام داده ها به طور موثر استفاده کنیم در حقیقت داده ها برای آموزش  
 و تست کردن استفاده می شوند در حالی که در روش Train/Test Split بخشی از داده فقط برای  
 آزمون استفاده می شود و در غایت آن داده های تست باقی می ماند

## 4 تعیین واریانس عملکرد مدل:

یک تعیین Cross-validation از عملکرد مدل ارائه می دهد بلکه یک تعیین واریانس  
 این عملکرد نیز ارائه می دهد این اطلاعات به ما کمک می کنند تا میزان اطمینان خود را از نتایج عملکرد مدل  
 برانجامیم اگر واریانس بالا باشد ممکن است لازم باشد داده ها را به روشی بهیچ آوری کنیم یا از  
 تکنیک های دیگری برای بهبود یادآوری مدل استفاده کنیم

یک ابزار قدرتمند است Cross-validation که به ما کمک می کند تا عملکرد مدل های

یادگیری ماشین را به طور دقیق تر و جامع تر ارزیابی کنیم بهترین مدل را انتخاب کنیم

پارامترهای مدل بجهت کنیم و از تمام Train/Test Split در تمام Cross-validation

از مجموع نتایج و قابلیت اطمینان بهیچ و یادگیری ماشین تعیین کنند

## 1- Gradient Descent چگونه کار می کند؟

تئوری (Gradient Descent) این الگوریتم بجهت سازی است که برای پیدا کردن  
 کمترین مقدار یک تابع معمولاً تابع هزینه استفاده می شود در واقع این الگوریتم با حرکت برداشتن در سمت  
 مخالف گرادیان تابع سعی می کند به کمترین مقدار آن برسد  
 در اصل گرادیان نامش

Subject:

Date:

1 تابع تعریف هزینه (Cost function):  
ابتدا باید تابعی تعریف کنیم که میزان اشتباه یا هزینه مدل ما را نشان دهد. هدف این است که این تابع را به حداقل برسانیم.

2 مقداردهی اولیه (Initialization):  
پارامترهای مدل را مانند وزن ها و بایاس ها را با مقادیر تصادفی یا صفر مقداردهی اولیه می کنیم.

3 محاسبه گرادیان (Gradient calculation):  
گرادیان تابع هزینه را نسبت به پارامترهای مدل محاسبه می کنیم. گرادیان جهت بیشترین افزایش تابع را نشان می دهد. به عبارتی دیگر، گرادیان جهت بیشترین کاهش تابع را نشان می دهد.

4 به روز رسانی پارامترها (Parameter update):  
پارامترهای مدل را با استفاده از گرادیان و نرخ یادگیری به روز می کنیم.

پارامترهای مدل را با استفاده از گرادیان و نرخ یادگیری به روز می کنیم. نرخ یادگیری یک مقدار کوچک است که تعیین می کند چقدر در هر بار به روز رسانی کنیم. اگر نرخ یادگیری خیلی بزرگ باشد ممکن است از کمترین مقدار بگذریم و اگر خیلی کوچک باشد ممکن است در رسیدن به کمترین مقدار خیلی طول بکشد.

5 تکرار (Iteration):  
مراحل 3 و 4 را تکرار می کنیم تا به نقطه توقف برسیم. به سیم بندی می کنیم که تا زمانی که تغییر در تابع هزینه کمتر از یک مقدار مشخص باشد یا زمانی که تعداد تکرارها به یک حد مشخص رسیده باشد، فرآیند را متوقف می کنیم.

بعد از اتمام فرآیند یادگیری، مدل ما به روز شده است. ما می توانیم مدل ما را با داده های جدید تست کنیم. اگر مدل ما نتواند داده های جدید را به خوبی پیش بینی کند، ممکن است نیاز به تنظیم مجدد پارامترها یا تغییر نرخ یادگیری داشته باشیم.



Subject:

Date:

1. چهار Deep Learning برای پیچیده ترین مسائل استفاده می شود.  
یادگیری عمیق (Deep Learning) به عنوان یکی از جدیدترین و قدرتمندترین روش‌های استفاده از ماشین‌های یادگیری شناخته می‌شود.

1 قابلیت یادگیری ویژگی‌های پیچیده (Learning Features):  
• یادگیری عمیق برخلاف روش‌های سنتی، یادگیری را به گونه‌ای قادر است به طور خودکار ویژگی‌های مناسب و مهم را از داده‌ها استخراج کند که برای حل مسئله در دنیای واقعی تخصص بالاست.  
• شبکه‌های عصبی عمیق قادرند الگوهای پیچیده و غیرخطی را در داده‌ها تشخیص دهند و به این ترتیب می‌توانند الگوهای اولیه و ویژگی‌های ساده را استخراج می‌کنند و الگوهای جدید را ترکیب این ویژگی‌ها و ویژگی‌های پیچیده‌تری را بسازند.

2. مقایسه پذیری با داده‌های زیاد:

• عملکرد الگوریتم‌های یادگیری عمیق با افزایش حجم داده‌ها به طور قابل توجهی بهبود می‌یابد و نتایجی

که بسیاری از الگوریتم‌های سنتی با افزایش داده‌ها به دست می‌آورند به دست می‌آورند و به دست می‌آورند.

تجارب نشان می‌دهد که شبکه‌های عصبی برای یادگیری الگوهای پیچیده و غیرخطی استفاده می‌کنند.

• در مسائل پیچیده حجم داده‌ها معمولاً بسیار زیاد است و یادگیری عمیق بهترین گزینه برای

استفاده از این داده‌ها است.

3. انعطاف پذیری و قابلیت تعمیم: یادگیری عمیق به دلیل ساختار انعطاف‌پذیری

شبکه‌های عصبی می‌تواند برای حل انواع مختلفی از مسائل مورد استفاده قرار گیرد و به سادگی می‌تواند



Subject:

Date:

شبکه عصبی (مانند شبکه های لایه پنهان یا RNN) می تواند با تغییراتی جزئی برای مسائل

مختلف مانند تشخیص خطوط و پردازش زبان طبیعی و تشخیص صدا استفاده می شود

شبکه های عصبی عموماً می توانند الگوهای عصبی را از داده های یادگیرنده مربوطه فشرده بردارند

و این فشرده نگار یا نقش می شود که مدل های یادگیری عمیق در دنیای واقعی که داده های

معمولاً در حال تغییر هستند و عملکرد خوبی دارند باشند

4- قدرت مدل برای روابط غیر خطی:

بسیاری از مسائل پیچیده در دنیای واقعی دارای روابط غیر خطی بین تغییرات هستند. شبکه های

عصبی عمیق با استفاده از توانایی برای غیر خطی در لایه های مختلف قادرند این روابط غیر

خطی را نقش می شود که یادگیرش عمیق برای حل مسائل مانند تشخیص بیماری ها

که دارای پیچیدگی های زیاد هستند مناسب باشد

مثال ها

خود رهای خود را از تغییرات عموماً برای تشخیص اشیا (مانند عابر پیاده و خودروهایی  
دیگر) مانند راهنمایی و ترافیک و تعمیرات و در صورت غمزه اندکی استفاده می شود

تشخیص پزشکی: یادگیری عمیق برای تشخیص بیماری ها از روشی عموماً پزشکی استفاده می کنند

Subject :

Date :

رادیو لفظی و (NLP) و پیش بینی خطر ابتلا به بیماری های استفاده می شود

موضوع ماسترین : یادگیری عمیق برای ترجمه متون از یک زبان به زبان دیگر با دقت بالا

استفاده می شود

پیدا کردن زبان طبیعی (NLP) : یادگیری عمیق برای تحلیل متون در یک زبان طبیعی

صنایع استفاده می شود