

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

Subject: -----

به نام خدا

تقریبات : بخش اول

استاد : آقای محمد احمدزاده

درس : صباحت و پرده

اعضای گروه : سیامه بخاری . سید اصابی

Subject:

Year:

Month:

Date:

Sa Su Mo Tu We Th Fr

۱- unsupervised Learning تفاوتی دارند؟ تفاوت اصلی بین unsupervised Learning در این است که مشکل در ساخت مدل است در حالی که تفاوت اصلی بین underfitting و unsupervised Learning نوع یادگیری ماشین است می بیند این تفاوت را دقیق تر بررسی کنیم

ماهیت: یک مشکل یا وضعیت نامطلوب در یادگیری ماشین است

علت: مدل بسیار ساده است نمی تواند الگوی های موجود در داده ها را یاد بگیرد

تتبع: عملکرد ضعیف در داده های آموزشی و تعیین دمی ضعیف به داده های جدید

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

unsupervised Learning

ماهیت: یک رویکرد در یادگیری ماشین است

هدف: کشف الگوها ساختارها و روابط پنهان در داده های برچسب دار

(labeled data)

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

به عبارت دیگر underfitting یک مشکل است که می تواند در مدل 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 به دیتا نمی گیرد



Subject:

Year. Month. Date.

Sa Su Mo Tu We Th Fr

۲ چرا ضروری است machine Learning در الگوریتم‌های Feature

Featurescaling

machine Learning به دلیل زیر ضروری است!  
مساویت الگوریتم‌ها به مقیاس: بسیاری از الگوریتم‌های یادگیری ماشین به خصوص الگوریتم‌ها

های که از فاصله (distance) برای اندازه‌گیری شباهت یا تفاوت بین داده‌ها استفاده

می‌کنند مانند الگوریتم‌های رگرسیون خطی (Linear)

سرعت هدرایی

الگوریتم‌های که از روش‌های مبتنی بر گرادیان (Gradient-Based) برای بهینه

سازی استفاده می‌کنند مانند رگرسیون خطی و لجستیک با گرادیان کاهش (Gradient)

شبکه‌های عصبی در صورت عدم مقیاس بندی ویژگی‌ها ممکن است به سختی هدر شوند یا به

زمان بیشتر برای هدرایی نیاز داشته باشند.

مقیاس بندی ویژگی‌ها باعث می‌شود

Subject:

Year. Month. Date.

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

## ۳- Normalization و Standardization تفاوتی دارند؟

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

های (Features) در داده های ورودی استفاده می شوند و در 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-

۱- معنویت کردن دامنه داده های تداوم دار به دامنه ای مشخص

عموماً بین ۰ و ۱ مسطح کنونی کار می تواند به کاهش آسیب پذیری الگوریتم های

یادگیری ماشین به مقدار اضافه ای (outliers) کمک کند

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

ماشین عمده سریع تر و بهتر عمل می کنند این به ویژه در الگوریتم های حساس

به مقیاس مانند (K-NN) K-Nearest Neighbor حساس است



Subject:

Year.

Month.

Date.

Sa Su Mo Tu We Th Fr

## ۵۴ Z-Score Normalization چیست و چرا کاربرد دارد؟

بازنормال سازی Z-score، روشی برای تبدیل داده‌ها به توزیع میانگین صفر و انحراف معیار یک است. در این روش هر مقدار داده با کم کردن میانگین از آن تقسیم بر انحراف معیار 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 معیار داده‌ها

## Z-Score Normalization کاربردهای فراوانی دارد از جمله

مقایسه داده‌ها با مقیاس‌های مختلف: وقتی داده‌های با مقیاس‌های متفاوت (مثلاً متران و قد) داریم نمی‌توانیم به هم مقایسه کنیم. با Z-score Normalization این مشکل را حل می‌کنند و همه داده‌ها را به یک مقیاس واحد می‌آورند.

جمع‌بندی عملکرد الگوریتم‌های یادگیری ماشین: بسیاری از الگوریتم‌های یادگیری ماشین (مثل شبکه‌های عصبی و درخت تصمیم) بر داده‌های نرمال شده حساس هستند. با Z-score Normalization می‌توانیم عملکرد الگوریتم‌ها را به هم مقایسه کنیم و بفهمیم کدام یک از الگوریتم‌ها عملکرد بهتری دارند.

شناسایی نقاط خارج از محدوده (outliers): با استفاده از Z-score می‌توانیم به راحتی نقاط خارج از محدوده را شناسایی کنیم. اگر Z-score یک نقطه خیلی بزرگ یا خیلی کوچک باشد (مثلاً بیشتر از 3 یا کمتر از -3) احتمالاً آن نقطه یک outlier است.

استاندارد سازی داده‌ها: Z-score با نرمال کردن داده‌ها به توزیع میانگین صفر و انحراف معیار یک (یا 1) می‌تواند به ما کمک کند تا داده‌ها را به هم مقایسه کنیم. این کار برای تحلیل‌های آماری و همچنین برای مقایسه نتایج مدل‌های مختلف بسیار مفید است.

در کل Z-Score Normalization یک تکنیک ساده و کاربردی برای نرمال کردن داده‌ها و تحلیل آماری است که به ما کمک می‌کند تا داده‌های مختلف را به هم مقایسه کنیم و نتایج بهتری از مدل‌های خودمان بگیریم.



## Regularization در الگوریتمهای Machine Learning چیست؟

منظوم سازی (Regularization) در یادگیری ماشین روشی است برای

جلوگیری از بیش برازش (overfitting) مدل بیش برازش زمانی رخ می دهد که مدل

داده های آموزشی را بیش از حد دقیق یاد می گیرد و در نتیجه در پیش بینی داده های جدید

(داده های دیده نشده) عملکرد ضعیفی از خود نشان می دهد و مدل در واقع نویز و جزئیات

بی اهمیت در داده های آموزشی را نیز یاد می گیرد که در داده های واقعی وجود ندارند.

منظوم سازی با افزودن یک جریمه (penalty) به تابع هزینه مدل از پیچیدگی بیش از حد

مدل جلوگیری می کند. این جریمه بزرگی وزن های مدل را کنترل می کند و وزن های بزرگ

تر نشان دهنده مدل پیچیده تر و ضعیف تر به پیش برازش هستند.

دو روش اصلی منظوم سازی عبارتند از:

**L1 Regularization (Lasso):** در این روش جریمه ای اضافه شده به تابع هزینه برابر با

مجموع قدر مطلق 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 را حذف می کند.

**L2 Regularization (Ridge):** این روش جریمه ای اضافه شده به تابع

هزینه برابر با مجموع مربعات وزن ها است. این روش وزن های را به سمت صفر کوچک می کند

تا آنکه منظوم سازی.

کاهش بیش برازش: با کنترل وزن ها از پیچیدگی بیش از حد مدل جلوگیری می شود

و عملکرد مدل در داده های دیده نشده بهبود می یابد. منظور از منظوم سازی عملکرد بهتر در

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

Underfitting (اورفیتینگ) و Overfitting (اورفیتینگ) در مدل سازی اورفیتینگ که می توان به دقت و 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 حساس و ویژگی های خاص داده های موجود در داده های آموزشی را به عنوان ویژگی های 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 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 ضعیفی در داده های جدید: این اصلی ترین مشکل اورفیتینگ است مدل در داده های آموزشی عملکرد عالی دارد اما در داده های جدید شده دقت بسیار پایینی نشان می دهد  
پیچیدگی بیش از حد مدل:

مدل های 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 ویژگی های 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 بیشتری دارند که می تواند زمان و منابع بیشتری را مصرف کند



Subject:

Date:

cross-validation چرادر Train/Test split، برادر؟

Cross-validation (اعتبار، سنجی، شش‌طالع) یک تکنیک نه برای ارزیابی عملکرد مدل‌های یادگیری ماشین استفاده می‌شود به‌ویژه در کنار روش Train/Test split (تقسیم داده به آموزش و آزمون) کاربرد دارد در انتخاب مدل اصلی استفاده از cross-validation

1 ارزیابی دقیق‌تر و جامع‌تر عملکرد مدل:  
معمولاً در روش Train/Test split، داده‌ها فقط یک بار به دو بخش آموزش و آزمون تقسیم می‌شوند این بدان معناست که عملکرد مدل فقط بر اساس یک مجموعه داده آزمون خاص به‌طور تصادفی تعیین می‌شود. با روش ارزیابی عملکرد مدل، امکان استوار کردن نتایج دقیق‌تر دست نیاید

تجزیه cross-validation:  
با تقسیم داده‌ها به چندین زیرمجموعه (fold) و تکرار فرایند آموزش و آزمون بر روی هر یک از این زیرمجموعه‌ها، ارزیابی جامع‌تر از عملکرد مدل را می‌توان داشت. به عبارت دیگر، با تکرار فرایند آموزش و آزمون بر روی داده‌های مختلف، می‌توان به دست آورد

2 انتخاب بهترین مدل و تنظیم پارامترها (Hyperparameter Tuning):  
انتخاب مدل و روش چندین مدل مختلف و تنظیم پارامترها (Hyperparameter Tuning) می‌تواند به دست آوردن بهترین مدل و تنظیم پارامترها کمک کند. به‌طور کلی، بهترین مدل و تنظیم پارامترها را می‌توان با استفاده از cross-validation به دست آورد. این فرایند شامل ارزیابی عملکرد مدل‌های مختلف و تنظیم پارامترها بر اساس نتایج cross-validation است.

تنظیم پارامترها:  
بسیاری از مدل‌های یادگیری ماشین دارای پارامترهای قابل تنظیم هستند. به‌عنوان مثال، در مدل‌های خطی، پارامترهای مختلف می‌توانند به دست آورده شوند. cross-validation به ما کمک می‌کند تا بهترین تنظیمات پارامترها را برای مدل‌های مختلف پیدا کنیم.

عملکرد را به دست می‌آوریم به این فرایند Hyperparameter Tuning گفته می‌شود.  
MICRO



## 3 استفاده بهینه از داده ها:

Cross-validation  
 محبوب و داده سازد به بیاری از پروژه های یادگیری ماشین داده ها محدود هستند  
 به ما کمک می کند تا از تمام داده های موجود، موثر استفاده کنیم در هر مدل خاص، تمام داده ها برای آموزش  
 و آزمون استفاده می شوند در زمانی که در روش Train/Test split بخشی از داده فقط برای  
 آزمون استفاده می شود و در فرآیند آزمون نتایج کمی می کنند

## 4 تخصیص واریانس عملکرد مدل:

یک تخصیص Cross-validation، عملکرد مدل ارائه می دهد بلکه یک تخصیص واریانس  
 این عملکرد نیز ارائه می دهد این اطلاعات به ما کمک می کنند تا میزان اطمینان خود را در ارزیابی عملکرد مدل  
 را بسنجیم اگر واریانس بالا باشد ممکن است لازم باشد داده ها را به روشی بهتر به جمع آور می کنیم تا از  
 تغییرات های دیگری برای بهبود یادگیری مدل استفاده کنیم

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

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

پارامترهای مدل را بهینه کنیم و از تمام Train/Test split در Cross-validation

از بهبود کیفیت و قابلیت اطمینان به پروژه یادگیری ماشین کمک کند

## 1- Gradient Descent چگونه کار می کند؟

در دنیای واقعی (Gradient Descent) این الگوریتم بهینه سازی است که برای پیدا کردن  
 کمترین مقدار یک تابع محدوداً تابع هزینه استفاده می شود در واقع این الگوریتم با قدم برداشتن به سمت  
 مخالف گرادیان تابع می تواند به کمترین مقدار آن برسد  
 در ادامه به دنیای واقعی

Subject:

Date:

1 تابع تعریف هزینه (Cost Function):

ابتدا باید تابع 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 و 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 عبارت دیگر گام های افزایش خطای آن حرکت کنیم تا به کمترین مقدار تابع برسیم

4 به روز رسانی پارامترها (Parameter Update): پارامترهای مدل را با استفاده از فرمول زیر به روز می کنیم

پارامتر جدید = پارامتر قبلی - نرخ یادگیری  $\times$  گرادیان  
نرخ یادگیری یک مقدار کوچک است که تعیین می کند چه مقدار در هر قدم حرکت کنیم اگر نرخ یادگیری خیلی بزرگ باشد ممکن است از کمترین مقدار بگذریم و اگر خیلی کوچک باشد ممکن است رسیدن به کمترین مقدار خیلی طول بکشد

5 تکرار (Iteration): مراحل 3 و 4 را زمانی که به یک شرط توقف برسیم به سیم تکرار می کنیم  
شرط توقف می تواند مانند نایندارد شدن مقدار گرادیان یا پایان یافتن تعداد تکرار باشد  
تغییر در تابع هزینه از یک مقدار مشخص کمتر باشد به یک زمان عین برسیم

6 به پایان ناهشی می الکوریتم قدرتی می رسیم در یادگیری ماشین است که ما گام های زیادی را طی می کنیم تا به یک مدل مناسب برسیم و تغییرات نرخ یادگیری را در حین یادگیری می توانیم به کار ببریم تا به یک مدل مناسب برسیم



Subject:

Date:

لچھا Deep Learning برای پیچیده ترین مسائل استفاده می شود.  
یادگیری عمیق (Deep Learning) به چندین دلیل مورد برای حل پیچیده و مورد استفاده فراوان است.

1 قابلیت یادگیری ویژگی های پیچیده (Learning Feature):

• یادگیری عمیق برخلاف الگوریتم های سنتی یادگیری ماشین قادر است به طور خودکار ویژگی های ذاتی حجم و پیچیده را از داده ها استخراج کند که کار برای انسان بسیار دشوار است.  
• شبکه های عمیق عمیق یادگیری لایه های متعددی توانایی ویژگی های انتزاعی و پیچیده را برای رایانگی دارند لایه های اولیه ویژگی های ساده را استخراج می کنند و لایه های بعدی با ترکیب این ویژگی ها ویژگی های پیچیده تری را می سازند.

2 مقایسه پذیری با داده های زیاد:

بعلکد الگوریتم های یادگیری عمیق با افزایش حجم داده ها به طور قابل توجهی بهبود می یابند و نتایجی

که بسیاری از الگوریتم های سنتی با افزایش داده ها به دست می آورند به دست می آورند و بهبود

قابل توجهی نشان می دهند شبکه های عمیق برای یادگیری الگو های پیچیده و استفاده کنند

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

استفاده از این داده ها است

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

شبکه های عمیق می تواند برای حل انواع مختلفی از مسائل مورد استفاده قرار گیرد و این یک ویژگی

Subject:

Date:

تشکیل عصبی (مانند شبکه های کانولوشن یا RNN) می تواند با تغییراتی جزئی برای مسائل

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

تشکیل های عصبی عموماً می توانند الگوهای عصبی را از داده ها یاد بگیرند و به خوبی به داده

های جدید تعمیم یافت می شود که مدل های یادگیری عصبی در دنیای واقعی بر داده های

معمولاً در حال تغییر هستند و عملکرد خوبی دارند

4- قدرت مدل سازی روابط غیر خطی:

بسیاری از مسائل پیچیده در دنیای واقعی دارای روابط غیر خطی بین متغیرها هستند که مدل های

عصبی عموماً قادرند از توانمندی های غیر خطی در لایه های مختلف کار کنند این روابط غیر

خطی باعث می شود که یادگیری عصبی برای حل مسائل مانند تشخیص بینامتنی

که دارای پیچیدگی های زیاد هستند مناسب باشد

مثال ها

خود رهای خود را از یادگیری عصبی برای تشخیص اشیاء مانند عابر پیاده خود در ویدیو  
دیگر علائم راهنمایی و ترافیکی و تعمیم گیری در صورت غده رانندگی استفاده می شود

تشخیص پزشکی: یادگیری عصبی برای تشخیص بیماری ها از روی تصاویر پزشکی مانند



Subject:

Date:

رادیولوژی و MRI و پیش بینی خطر ابتلا به بیماری های استفاده می شود

موجوده ماشین های اسکینینگ عمیق برای ترجمه متن از یک زبان به زبان دیگر با دقت بالا

استفاده می شود

پد دانش زبان طبیعی (NLP): یادگیری عمیق برای تحلیل متن در یک زبان طبیعی می تواند

حق استفاده می شود