



در شماره گذشته با بخش اول مطلب چگونه می‌توانیم در عمل از یادگیری ماشین استفاده کنیم؟ آشنا شدیم و به شما گفتیم که دو رویکرد یادگیری تحت نظارت و بدون نظارت چه تفاوت‌هایی با یکدیگر دارند و چرا پیاده‌سازی الگوی یادگیری تحت نظارت و برجسب‌گذاری داده‌ها از اهمیت خاصی برخوردار است. در این شماره بخش دوم این مطلب را ادامه دهیم.

در **بخش اول مقاله** «[چگونه می‌توانیم در عمل از یادگیری ماشین استفاده کنیم؟](#)» دیدید که در ابتدا یکسری داده‌ها را جمع‌آوری کردیم، داده‌های به‌دست‌آمده را در سه مرحله مقدماتی و ساده به الگوریتم ژنریک خود تحویل دادیم و در نهایت تابعی ایجاد کردیم که قادر بود پیش‌بینی قیمت همه خانه‌های موجود در یک منطقه را ارائه کند. اما سه مرحله این الگوریتم را به شرح زیر پیاده‌سازی می‌کنیم:

مرحله اول

در این مرحله همه وزن‌ها را برابر با مقدار یک تعیین می‌کنیم.

```
def estimate_house_sales_price(num_of_bedrooms, sqft, neighborhood):
    price = 0
    # a little pinch of this
    price += num_of_bedrooms * 1.0
    # and a big pinch of that
    price += sqft * 1.0
    # maybe a handful of this
    price += neighborhood * 1.0
    # and finally, just a little extra salt for good measure
    price += 1.0
    return price
```

## مرحله دوم

در این مرحله داده‌های مربوط به خانه‌های به فروش رفته را در اختیار تابع قرار می‌دهیم و پیش‌بینی انجام شده را با قیمت واقعی خانه‌ها ارزیابی می‌کنیم تا از صحت درستی تابع و انحراف معیار تابع و قیمتی که به‌عنوان پیش‌بینی ارائه کرده، اطلاع پیدا کنیم. برای مثال، اگر قیمت یک‌خانه برابر 250 هزار دلار است، اما مدل قیمت آن را 178 هزار دلار برآورد کرده، شما در اینجا خطایی به میزان 72 هزار دلار دارید. اکنون باید مربع انحراف را برای همه خانه‌های مجموعه آموزشی جمع کنید. (به عبارت ساده‌تر باید میزان خطای به وجود آمده را برای هر خانه‌ای که در

مجموعه داده‌ای وجود دارد به توان 2 رسانده و مقادیر را با یکدیگر جمع کنید.) اگر 500 خانه در مجموعه داده‌ای شما وجود دارد، در این حالت مربع خطا برای همه خانه‌ها برابر با 86,123,373 دلار خواهد بود. این مقدار بیانگر کارکرد اشتباه تابع شما در این الگوریتم است. (شکل 1)

حالا باید مجموع به دست آمده را بر 500 تقسیم کنید تا میانگین مربع‌های خطا برای هر خانه به دست آید. مقدار میانگین به دست آمده هزینه (Cost) تابع شما خواهد بود. حال اگر بتوانیم وزن‌ها را به گونه‌ای تغییر دهیم که این هزینه‌ها به صفر متمایل شود، عملکرد تابع دقیق خواهد بود. به عبارت دقیق‌تر، تابع در زمینه محاسبه قیمت خانه در شرایط مختلف بر مبنای داده‌های ورودی عملکردی درست خواهد داشت. پس باید وزن‌ها را به شکلی تغییر دهیم تا هزینه به کمترین حد خود برسد.

## مرحله سوم

اکنون باید یکبار دیگر مرحله دوم را با ترکیبی از وزن‌های مختلف تکرار کرده و در ادامه سعی کنیم ترکیبی را انتخاب کنیم تا مقدار هزینه تابع را به صفر نزدیک کند. اگر در انجام این کار موفق شویم، به معنای آن خواهد بود که برای این مشکل توانسته‌ایم راه‌حلی پیدا کنیم.

## چند نکته کلیدی در ارتباط با روشی که به آن اشاره کردیم

برای آن‌که با توانایی الگوریتم‌های یادگیری ماشین بهتر آشنا شوید، اجازه دهید به چند نکته کلیدی اشاره کنیم:

- پژوهش‌هایی که در چهار دهه گذشته در ارتباط با هوش مصنوعی و در حوزه‌هایی شبیه به زبان‌شناسی/ ترجمه انجام شده، به وضوح نشان داده که الگوریتم‌های یادگیری ماشین ژنریک که در تعامل با آزمایش ترکیب‌های مختلفی از ارقام و اعداد هستند، زمانی که در ارتباط با یکسری دستورالعمل‌های خاص به کار گرفته شوند، کارایی خیلی بهتری نسبت به عامل انسانی دارند. حتی الگوریتم‌هایی که از دستورالعمل خاصی تبعیت نمی‌کنند نیز در عمل نشان داده‌اند که بهتر از یک کارشناس حرفه‌ای قادر به پیش‌بینی هستند.
- تابعی که در بالا آن را پیاده‌سازی کردید، در حالت کلی غیرهوشمند (Dumb) است. تابعی که در عمل ممکن است فاکتورهای همچون مساحت خانه و تعداد اتاق‌خواب‌ها را برای ارائه یک تخمین به کار ببرند، هیچ‌گاه نمی‌دانند که معنای تعداد اتاق‌ها یا فوت مربع چیست. تنها مزیتی که چنین تابعی دارند در این است که قادر هستند ترکیب‌های مختلفی از ارقام را به عنوان وزن‌های مدل آزمایش کنند تا بتوانند پاسخ درستی را ارائه کنند.
- تجربه نشان دادن داده که حتی توسعه‌دهندگان چنین تابعی نیز به درستی نمی‌دانند که چرا یک مجموعه خاص از وزن‌ها قادر هستند نتایج درستی را ارائه کنند. به عبارت دقیق‌تر، توسعه‌دهنده به شما اعلام می‌دارد تابعی طراحی کرده که نحوه کار آن را به درستی نمی‌تواند تشریح کند، اما در عمل می‌تواند ثابت کند که این تابع به درستی کار خواهد کرد. اتفاقی که نزدیک به دو سال پیش برای متخصصان الگوریتم هوشمند گوگل رخ داد. تابع آن‌ها در آزمایش نتایج درستی را ارائه می‌کرد اما پژوهشگران این شرکت از درک عملکرد و توضیح روشی که الگوریتم به کار گرفته عاجز بودند. (البته این مسئله در ارتباط با شبکه عمیق عصبی بود.)

## یک فرضیه دیگر

تصور کنید که ما به جای آن‌که پارامترهای همچون مساحت خانه و تعداد اتاق‌خواب‌ها را به تابع تحویل دهیم، تنها یکسری ارقام را به عنوان ورودی در اختیار تابع قرار دهیم. اعدادی که تنها بیانگر پیکسل‌هایی هستند که درون یک تصویر قرار دارند. تصویری که یک دوربین نصب شده مدار بسته بالای یک ماشین تهیه کرده است. در این حالت تابع فوق باید در خروجی خود به جای آن‌که به برآورد پیش‌بینی قیمت بپردازد، برآوردی در ارتباط با میزان زاویه چرخش فرمان ماشین ارائه کند. به عبارت دقیق‌تر، ما به تابعی نیاز داریم که از طریق آن یک ماشین خودران قادر باشد به تهای هدایت را بر عهده بگیرد. اگر سعی کنیم بر مبنای سناریویی که در پاراگراف قبل به آن اشاره داشتیم، ترکیب‌های موجود در گام سوم را آزمایش کنیم، به امید آن‌که به ترکیبی برسیم که نتیجه مدنظر ما را ارائه کند، شانس کمی برای موفقیت خواهیم داشت. به دلیل این‌که در عمل با یک پروسه زمان‌بر و طاقت‌فرسا روبه‌رو خواهیم شد. در چنین شرایطی پژوهشگران برای حل مسائلی این‌چنینی سعی می‌کنند از روش‌هایی که هوشمندی بیشتری دارند استفاده کنند. رویکردی که به آن‌ها اجازه می‌دهد به فرایند پیدا کردن مقادیر صحیحی که بتوان برای وزن‌ها از آن‌ها استفاده کرد شتاب بیشتری ببخشد. به کارگیری چنین راهکاری به آن‌ها اجازه می‌دهد دیگر نیازی نداشته باشند

ترکیب‌های مختلفی از وزن‌ها را آزمایش کنند. اما چطور می‌توان به چنین راهکاری دست پیدا کرد؟

## راهکاری هوشمندانه برای پیدا کردن وزن مناسب

اجازه دهید در نخستین مرحله معادله ساده‌ای را بنویسیم. معادله‌ای که به بیان ساده در نظر دارد تابع هزینه را که در مرحله دوم طراحی کردیم، تشریح کند. (شکل 2)

شکل 2. معادله هزینه

$$\text{Cost} = \frac{\sum_{i=1}^{500} (\text{MyGuess}(i) - \text{RealAnswer}(i))^2}{500 \cdot 2}$$

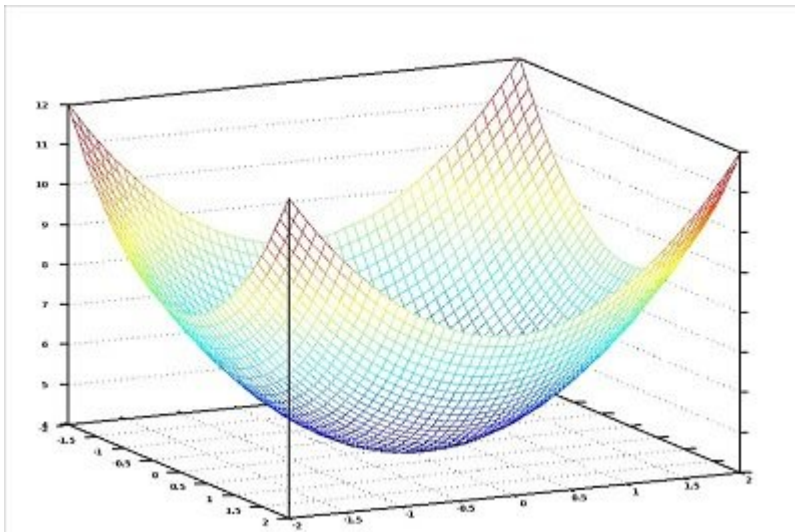
اکنون باید معادله نوشته‌شده را از طریق یکسری المان‌های ریاضی که در اصلاح به آن Jargon می‌گویند، بازنویسی کرده تا صورت **یادگیری ماشین** به خود پیدا کند (شکل 3).

شکل 3. معادله هزینه در فرم یادگیری ماشین

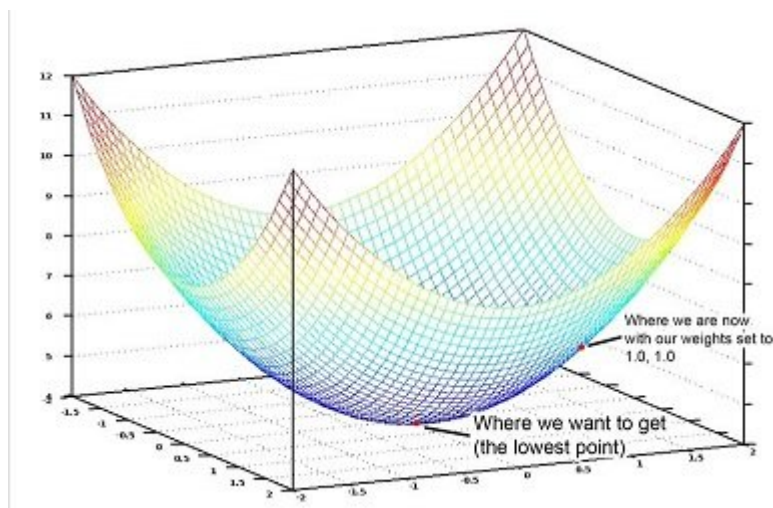
$$J(\theta) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2$$

در رابطه فوق تنها نشان‌دهنده وزن‌های فعالی است که در الگوریتم به کار گرفته‌شده و جی‌تتا تابع هزینه است که به ازای آن ترکیبی از وزن‌ها به دست می‌آید. (شکل 4)

شکل 4. نمایش سه‌بعدی معادله هزینه



معادله فوق میزان خطای تابع تخمین قیمت را به ازای مجموعه وزن‌هایی که برای مدل در نظر گرفته‌شده نشان می‌دهد. برای دقیق‌تر شدن معادله به رسم نموداری نیاز داریم تا هزینه را برای همه مقادیر ممکن از وزن‌ها و با توجه به در نظر گرفتن فاکتورهای همچون تعداد اتاق‌خواب‌های خانه و اندازه خانه به ما نشان دهد. در نموداری که به دست خواهیم آورد، محور عمودی مقدار Cost را نشان می‌دهد. (شکل 5)



نقطه آبی که در انتها قرار دارد، بیانگر کمترین هزینه است. حال اگر تابع هزینه را با مقدار وزن‌های یک ارزیابی کنیم، از روی نمودار می‌توان این‌گونه برداشت کرد که تابع هزینه ما توانسته به مقدار اشتباه کمینه دست پیدا کند. در نتیجه هر چه مقدار هزینه به دست آمده از حل معادله فوق در مکان‌های بالاتری از این نمودار قرار گیرد، نشان می‌دهد تابع خطای بیشتری دارد و در نتیجه باید به دنبال پیدا کردن وزن‌هایی باشیم که مقدار تابع هزینه را برای هر یک از وزن‌ها در پایین‌ترین نقطه نمودار نشان دهد. در چنین شرایطی می‌توانیم ادعا کنیم جواب درست را به دست آورده‌ایم. پس باید به دنبال وزن‌های مدل به شکلی باشیم که نقاط در پایین‌ترین سطح از نمودار قرار گرفته و هر زمان تغییری در وزن‌ها به وجود آمد، حرکت پیوسته تابع هزینه به سمت پایین نمودار باشد. با این راهکار به دست آوردن مقدار کمینه درست برای تابع هزینه دیگر به آزمایش ترکیب‌های مختلفی از وزن‌ها نیازی نخواهد داشت و در نتیجه تنها از طریق اعمال چند تغییر کوچک در وزن‌ها می‌توانیم به ترکیب موردنظر دست پیدا کنیم. فارغ‌التحصیلان رشته ریاضی به خوبی می‌دانند که اگر از تابعی مشتق گرفته شود، شیب نمودار آن تابع در نقاط دلخواه به دست خواهد آمد. به زبان ساده، مشتق یک تابع به ما نشان می‌دهد از هر نقطه از نمودار چطور می‌توان به پایین‌ترین نقطه آن دسترسی پیدا کرد. در نتیجه اگر بتوانیم مشتق جزئی تابع هزینه را برای هر یک از وزن‌های مدل ارزیابی کنیم و نتیجه به دست آمده را از هر یک از وزن‌ها کم کنیم، قادر خواهیم بود تابع موردنیاز خود را یک گام به کمینه شدن نزدیک کنیم. کمینه شدن به معنای رسیدن به پایین نقطه نمودار است. ما قادر هستیم این فرایند را تا زمانی که مقدار کمینه تابع و بهترین ترکیب از وزن‌ها را برای مدل خود به دست آوریم، تکرار کنیم. روشی که به آن اشاره کردیم، ساده‌ترین تشریح از رویکرد Batch Gradient Descent است. رویکردی که برای به دست آوردن بهترین ترکیب از وزن‌های توابع امروزه استفاده می‌شود. توجه داشته باشید، کتابخانه‌های یادگیری ماشینی که امروزه در اختیار شما قرار دارند بر مبنای الگویی که در بالا به آن اشاره شد، کار می‌کنند. در نتیجه توابع موجود در این کتابخانه‌ها می‌توانند همه این کارها را به شکل خودکار انجام دهند، بدون آن‌که شما نیازی داشته باشید خود را درگیر پیاده‌سازی جزئیات بسیار ریز کنید.

## مطلب پیشنهادی



برنامه‌نویسی هوش مصنوعی  
دانلود کنید: کتاب هوش مصنوعی همراه با مثال‌های عملی

## توابعی که در فرایندهای یادگیری ماشینی به کار گرفته می‌شوند

فارغ‌التحصیلان رشته هوش مصنوعی به خوبی می‌دانند الگوریتم سه مرحله‌ای ذکر شده با عنوان رگرسیون خطی چند متغیره (Multivariate Linear Regression) شناخته شده است. الگوریتمی که با اتکا بر آن می‌توان معادله خطی محاسبه کرده، معادله فوق را روی همه داده‌های خانه‌های موجود تنظیم کرد و در نهایت معادله را آموزش داد. در نهایت، می‌توان از معادله به دست آمده به منظور برآورد قیمت فروش خانه‌هایی که جدید هستند و الگوریتم هیچ

شناختی از آن‌ها ندارد، استفاده کرد. همان‌گونه که ممکن است حدس زده باشید از این الگوریتم می‌توان برای حل مسئله بدیهی‌تر نیز استفاده کرد. به عبارت ساده‌تر، نمونه ما واقعا جنبه کاربردی دارد. هرچند از چنین رویکردی برای حل مسائل ساده می‌توان بهره برد و شاید برای حل مسائل پیچیده کارآیی لازم را نداشته باشد. به دلیل این‌که قیمت خانه‌ها را نمی‌توان تنها بر مبنای چند پارامتر تخمین زد و فاکتورهای مرتبط دیگری هم نیاز است. زمانی که فاکتورهای وابسته به صورت مسئله اضافه شوند، دیگر مسئله با یک معادله خطی ساده قابل حل نخواهد بود. اما نگران نباشید. روش‌های متعددی برای حل مسائل پیچیده وجود دارند که از آن جمله به شبکه‌های عصبی و SVM می‌توان اشاره کرد. الگوریتم‌هایی که برای مدیریت داده‌های غیرخطی می‌توانند استفاده شوند. هرچند این امکان وجود دارد تا الگوریتم‌های رگرسیون خطی را از طریق به‌کارگیری راهکارهای هوشمندانه‌ای برای تنظیم شدن با فاکتورهای وابسته به کار گرفت. اما در حالت کلی همه الگوریتم‌ها به دنبال آن هستند تا بهترین ترکیب از وزن‌ها را به‌منظور دستیابی به پاسخ دلخواه پیدا کنند. به‌عنوان خواننده‌ای که با دنیای هوش مصنوعی آشنایی دارد، ممکن است این پرسش را مطرح کنید که ما چرا به موضوع Overfitting اشاره‌ای نداشتیم. مفهومی که توصیف‌کننده این موضوع است که بخش عمده‌ای از الگوریتم‌های **یادگیری ماشین** روی داده‌های آموزشی تنظیم شده و فقط قادر هستند برای این داده‌ها جواب دقیقی را ارائه کرده و برای ارائه جواب برای داده‌های متفرقه ضریب دقت آن‌ها به‌شدت کاهش پیدا می‌کند. برای حل مشکل Overfitting در **مدل‌های یادگیری ماشین** ما به تکنیک‌هایی همچون Regularization و همچنین بهره‌مندی از مجموعه داده‌های Cross-Validation دسترسی داریم اما توصیف این روش‌ها و مشکلات خارج از این نوشتار است.

## کلام آخر

درحالی‌که مفهوم اصلی مدل‌های یادگیری ماشین ساده بوده، اما در مقابل برای آن‌که به نتایج دلخواه و دقیقی دست پیدا کنید به تجربه و مهارت بالایی نیاز دارید. مهارتی که با تمرین مستمر به دست می‌آید. اگر دو بخش این مطلب را مطالعه کرده باشید، به‌خوبی می‌دانید که الگوهای موجود در یادگیری ماشین این پتانسیل را دارند تا مسائل به‌ظاهر پیچیده‌ای را که جواب مشخصی برای آن‌ها وجود ندارد، به راحتی حل کنند. اما این الگوریتم‌ها تنها زمانی که داده‌های ورودی درستی را به دست آورند قادر هستند خروجی درستی را ارائه کنند. به بیان ساده، اگر مدلی برای پیش‌بینی قیمت خانه بنویسیم و در ادامه در نظر داشته باشیم بر مبنای داده‌ها نوع گل‌های موجود در خانه را پیش‌بینی کنیم، به موفقیتی دست پیدا نخواهیم کرد، به‌واسطه آن‌که هیچ‌گونه ارتباطی میان گل‌های یک‌خانه و قیمت برآورد شده خانه وجود ندارد. در نتیجه تلاش‌های مدل برای ارزیابی از آن جهت با شکست روبه‌رو خواهند شد که میان داده‌های ورودی و خروجی موردنظر ما ارتباط مشخص و مفهومی وجود ندارد تا مسئله حل شود. پس باید به دنبال مدل‌سازی روابطی باشیم که به معنای واقعی کلمه وجود دارند. به بیان ساده، اگر به‌عنوان یک انسان موفق نشوید میان داده‌های ورودی و خروجی ارتباطی پیدا کنید، مدل‌های یادگیری ماشین نیز موفق نخواهند بود. پس نگاه شما باید روی مسائلی باشد که انسان‌ها قادر هستند آن‌ها را حل کنند، مسائلی که کامپیوترها به شکل سریع‌تر و با دقت بیشتری قادر به حل آن‌ها هستند. برآورد قیمت سهام یک شرکت از جمله این موارد است.

## تاریخ انتشار:

16 بهمن 1397

### نشانی منبع:

<https://www.shabakeh-mag.com/workshop/14279/%DA%86%DA%AF%D9%88%D9%86%D9%87-%D9%85%DB%8C%E2%80%8C%D8%AA%D9%88%D8%A7%D9%86%DB%8C%D9%85-%D8%AF%D8%B1-%D8%B9%D9%85%D9%84-%D8%A7%D8%B2-%DB%8C%D8%A7%D8%AF%DA%AF%DB%8C%D8%B1%DB%8C-%D9%85%D8%A7%D8%B4%DB%8C%D9%86-%D8%A7%D8%B3%D8%AA%D9%81%D8%A7%D8%AF%D9%87-%DA%A9%D9%86%DB%8C%D9%85%D8%9F-%D8%A8%D8%AE%D8%B4-%D8%AF%D9%88%D9%85>