



کافی است چند دقیقه از وقت خود را صرف تحقیق درباره هوش مصنوعی و به‌ویژه یادگیری ماشین کنید تا ببینید اهل فن همگی به این حقیقت اذعان دارند که آینده از آن هوش مصنوعی است. البته به نظرم ایده بدی نیست که واژه آینده را از اصطلاح هوش مصنوعی حذف کنیم. به دلیل این‌که الگوریتم‌های هوشمند از مدت‌ها پیش از سوی همه ما به کار گرفته شده‌اند، حتی زمانی که خودمان هم نمی‌دانستیم در حال استفاده از آن‌ها هستیم. اما چطور می‌توانیم به حداقل دانش لازم در زمینه یادگیری ماشین دست پیدا کنیم؟ دانشی که به ما اجازه دهد زمانی که در جمعی قرار داریم درباره هوش مصنوعی صحبت کنیم؟ برای دستیابی به چنین دانشی به سطح بالایی از مطالعه و پژوهش نیاز دارید. اما در نوشتار این شماره و شماره آینده نحوه به‌کارگیری عملی یک الگوریتم یادگیری ماشین ساده را توضیح خواهیم داد.

حس می‌زنم مقاله‌ای که در حال مطالعه آن هستید، ارزش دنبال کردن را دارد. در این سری از مقاله‌ها سعی کرده‌ایم، مباحث مربوط به **یادگیری ماشین** را به‌گونه‌ای به رشته تحریر درآوریم که فهم آن‌ها حتی برای افرادی که پیش‌زمینه‌ای در این خصوص ندارند ساده باشد. در این مقاله و مقاله آینده سعی بر این بوده، مبحث به‌گونه‌ای آماده شود تا افراد متخصص نیز یک ایده اولیه در این زمینه به دست آورند. در ابتدا اجازه دهید تعریف کوتاهی از **یادگیری ماشین** داشته باشیم.

## مطلب پیشنهادی

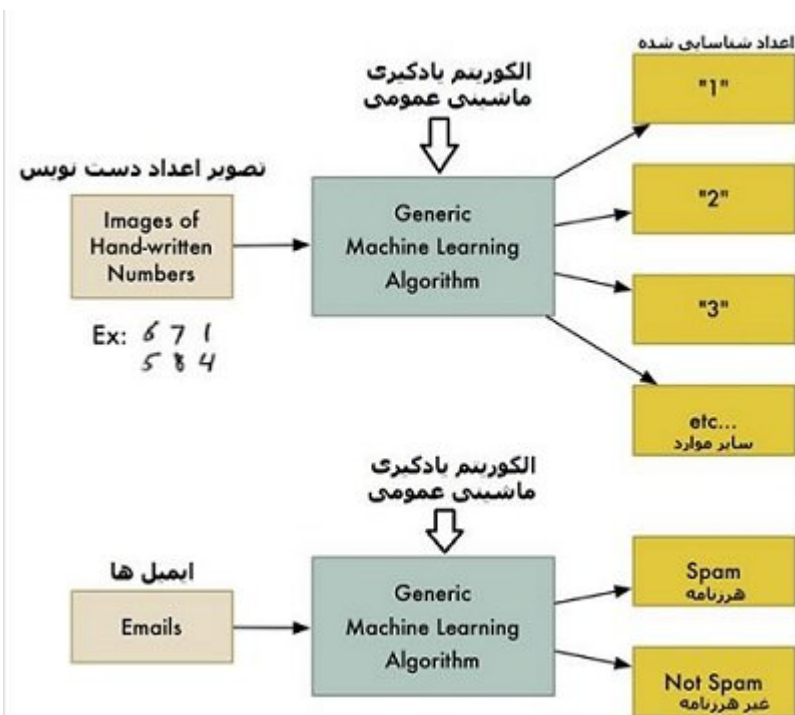


به کجا چنین شتابان  
پرونده ویژه «منزهای ماشینی؛ انسان یا برده؟» منتشر شد

## یادگیری ماشین چیست؟

**یادگیری ماشین** که در اصطلاح تخصصی از عبارت Machine Learning برای توصیف آن استفاده می‌کنند، به الگوریتم‌هایی اشاره دارد که بدون نیاز به کدنویسی اختصاصی برای گروهی از داده‌ها، می‌توانند اطلاعات مفیدی در ارتباط با داده‌ها در اختیار ما قرار دهند. در این حالت شما به‌جای کدنویسی، داده‌ها را در قالب ورودی در اختیار الگوریتم قرار می‌دهید و در ادامه این الگوریتم خواهد بود که بر مبنای داده‌های ورودی دریافت شده، منطق خودش را ایجاد می‌کند. الگوریتم طبقه‌بندی (Classification) نمونه‌ای از این موارد است که به ما اجازه می‌دهد داده‌ها را

در گروه‌های مختلفی طبقه‌بندی کنیم. الگوریتم Classification ضمن آن‌که برای شناسایی اعداد در یکدست خط انسانی به کار گرفته می‌شود، به‌منظور طبقه‌بندی ایمیل‌ها به دو گروه هرزنامه‌ها و غیر هرزنامه‌ها نیز به کار گرفته می‌شود. نکته جالب توجه آن‌که ما در این زمینه به هیچ‌گونه تغییری در کدها نیازی نداریم، به‌واسطه آن‌که در حال به‌کارگیری یک الگوریتم ژنریک هستیم. به عبارت دقیق‌تر، ما تنها نوع متفاوتی از داده‌های ورودی و مجموعه داده‌های ورودی را در قالب داده‌های آموزشی در اختیار الگوریتم قرار می‌دهیم و در ادامه این خود الگوریتم است که بر مبنای بینشی که به دست می‌آورد، یک طبقه‌بندی نهایی متفاوت را در قالب خروجی ارائه می‌کند. توجه داشته باشید که در این زمینه هرچه مقیاس داده‌های ورودی (مجموعه داده‌ها) بزرگ‌تر باشد، به همان نسبت یادگیری بهتر و دقیق‌تر خواهد بود.



## گونه‌های مختلف الگوریتم‌های یادگیری ماشین

در حالت کلی، در حوزه الگوریتم‌های یادگیری ماشین ما دو گروه یادگیری با نظارت (Supervised) و بدون نظارت (Unsupervised) را در اختیار داریم.

### Supervised Learning؛ یادگیری با نظارت

تصور کنید صاحب یک آژانس خرید و فروش مسکن و زمین هستید که برای بزرگ‌تر کردن کسب‌وکار خود تصمیم گرفته‌اید، چند نفر را به‌عنوان کارآموز استخدام کنید تا در این زمینه به شما کمک کنند. به‌عنوان مالک آژانس، شما با یک نگاه می‌توانید ارزش تقریبی یک‌خانه یا زمین را برآورد کنید، اما کارآموزان به‌واسطه فقدان مهارت، نمی‌توانند ارزش درست و منطقی را برای خانه‌های مختلف پیشنهاد کنند. در این زمان شما تصمیم می‌گیرید یک برنامه کاربردی را به‌منظور آموزش کارآموزان خود طراحی کنید. برنامه‌ای که قادر است ارزش یک‌خانه را بر مبنای پارامترهایی همچون منطقه و محله‌ای که خانه در آن قرار دارد، خانه‌هایی که در گذشته به فروش رفته، اندازه خانه و... تخمین بزند. برای نیل به این هدف شما باید داده‌هایی را در ارتباط با قیمت خانه‌های فروخته‌شده، محله، اندازه خانه، تعداد اتاق‌ها و فاکتورهای کاربردی در یک بازه زمانی جمع‌آوری کرده و ثبت کنید.

قیمت فروش	محل / منطقه	فوت مربع	انای خواب ها
\$250,000	Normaltown	2000	3
\$300,000	Hipsterton	800	2
\$150,000	Normaltown	850	2
\$78,000	Normaltown	550	1
\$150,000	Skid Row	2000	4

در ادامه باید این داده‌ها را در قالب داده‌های آموزشی (Training Data) در اختیار برنامه‌ای قرار دهید که در نظر دارد بر مبنای **هوش مصنوعی** ارزش یک‌خانه را در یک منطقه خاص برآورد کند. این مدل آموزش به نام یادگیری تحت نظارت شناخته می‌شود. در این مدل آموزش، ورودی‌هایی که در اختیار الگوریتم قرار می‌گیرند، داده‌های برجسته شده هستند و پاسخ مسئله نیز مشخص است. در مثال فوق ما درباره قیمت خانه‌های فروخته شده اطلاع داریم. به عبارت

دقیق‌تر جواب مسئله را از پیش می‌دانیم و در نتیجه می‌توانیم منطقی را که در بطن کار قرار دارد، کشف کنیم. الگوریتم نیز بر مبنای همین منطق مستتر در جواب می‌تواند در مورد مسائل آینده پاسخ مناسب را ارائه کند. در این حالت الگوریتم به دنبال پیدا کردن یک معادله ریاضی است که از طریق آن بتواند اعداد (رابطه‌ها) را پیدا کرده و آموزش خود را کامل کند.

قیمت فروش	محل / منطقه	فوت مربع	انای خواب ها
???	Hipsterton	2000	3

این رویکرد شبیه یک امتحان ریاضی است که در آن جواب تساوی‌ها در اختیارمان قرار دارد، اما عملگری که بین اعداد به کار گرفته شده در وضعیت مجهول قرار دارد. در

این وضعیت آیا می‌توانیم به پرسش امتحان پاسخ دهیم؟ جواب مثبت است. ما باید یکسری عملیات روی اعدادی انجام دهیم که در سمت چپ قرار دارند تا در ادامه به جواب سمت راست برسیم. (شکل 4)

سوال ریاضی اول- کلید پاسخ معلم	
1) $2 \ 4 \ 5 = 3$	5) $6 \ 2 \ 2 = 10$
2) $5 \ 2 \ 8 = 2$	6) $3 \ 1 \ 1 = 2$
3) $2 \ 2 \ 1 = 3$	7) $5 \ 3 \ 4 = 11$
4) $4 \ 2 \ 2 = 6$	8) $1 \ 8 \ 1 = 7$

در یادگیری تحت نظارت، ما از کامپیوتر درخواست می‌کنیم تا ارتباطات میان داده‌های ورودی را که به شکل‌گیری خروجی منجر شده‌اند، کشف کند. حال اگر منطق به کار گرفته شده در قالب داده‌های آموزشی برای حل یکسری مسائل

خاص شناسایی شوند، در ادامه این الگوریتم می‌تواند برای حل همه مسائل مشابه از چنین رویکردی استفاده کند. Un supervised learning یادگیری بدون نظارت

تصور کنید در موقعیتی قرار داریم که اطلاع نداریم خانه‌های به فروش رسیده در یک منطقه به چه قیمتی فروخته شده‌اند. در مقابل درباره اطلاعات کلی خانه‌ها، همچون ابعاد، محلی که خانه در آنجا ساخته شده و... اطلاعات لازم را داریم. در این حالت از رویکرد بدون نظارت برای آموزش مدل خود استفاده خواهیم کرد. این رویکرد، مشابه وضعیتی است که شخصی کاغذی در اختیارمان قرار دهد که تنها یکسری اعداد روی آن نوشته شده است. این شخص در ادامه به شما اعلام دارد رابطه میان این اعداد با یکدیگر را پیدا کنید. در ظاهر به نظر می‌رسد، هیچ‌گونه ایده‌ای در اختیار نداریم. اما شاید بتوانیم یک الگو برای آن‌ها پیدا کرده، آن‌ها را طبقه‌بندی کرده و مفهومی از دل آن استخراج کنیم. در چنین مواقعی ما از الگوریتم **یادگیری ماشینی** بدون ناظر یا همان بدون نظارت برای آموزش مدل خود استفاده می‌کنیم. به عبارت دقیق‌تر، به کارگیری **یادگیری ماشینی** بدون ناظر همانند این است که در نظر داشته باشید از میان فهرستی از اعدادی که مفهوم مشخصی ندارند، الگویی مشخص استخراج کرده یا آن‌ها را درون گروه‌های خاصی طبقه‌بندی کنید. اما پرسش اصلی این است که آیا می‌توان از درون چنین داده‌های خامی یک الگو استخراج کرد؟ در مثال ما، آیا به کارگیری چنین رویکردی امکان‌پذیر است؟ آیا با در اختیار داشتن داده‌های مربوط به

خانه‌ها، امکان پیاده‌سازی چنین الگوریتمی وجود دارد که قادر باشد به شکل خودکار روی داده‌های اولیه، یک طبقه‌بندی انجام داده و برآورد قیمتی در ارتباط با خانه‌های موجود در بازار ارائه کند؟ (شکل 5) پاسخ این پرسش مثبت است، اما به کمی زکات نیاز دارد.

محل / منطقه	فوت مربع	اتاق خواب ها
Normaltown	2000	3
Hipsterton	800	2
Normaltown	850	2
Normaltown	550	1
Skid Row	2000	4

در این مورد شما باید فرضیه‌هایی را برای خود مطرح کنید. به عنوان مثال، افرادی که قصد دارند خانه‌هایی در اطراف دانشگاه‌ها خریداری کنند، به دنبال خانه‌هایی کوچک با اتاق‌های زیاد هستند، در مقابل افرادی که به دنبال خرید خانه در اطراف شهرها هستند، به دنبال خانه‌های بزرگی هستند که سه خوابه باشد. پس باید در مورد مشتریان اطلاعاتی به دست آورید. اطلاعات لازم در خصوص مشتریان به شما اجازه می‌دهد داده‌ها را در گروه‌های مختلفی طبقه‌بندی کنید. این رویکرد به شما به عنوان یک فروشنده کمک می‌کند در زمینه بازاریابی به منظور فروش خانه‌ها موفق‌تر ظاهر شوید. البته راهکار دیگری نیز وجود دارد. شما می‌توانید به شکل خودکار داده‌های مربوط به خانه‌ها را جمع‌آوری کنید. به عنوان مثال، داده‌های خانه‌هایی را که وضعیت آن‌ها تفاوت‌های محسوسی با سایر خانه‌ها دارند، شناسایی کرده و آن‌ها را حذف کنید، به دلیل این‌که چنین خانه‌هایی مساحت زیادی داشته و برای افراد خاصی در نظر گرفته می‌شوند، افرادی که حاضر هستند پول بیشتری برای فروش یا خرید این خانه‌ها پرداخت کنند. با استناد به تعاریفی که برای این دو مدل ارائه شد، ما در نظر داریم در این مطلب تمرکز خود را روی یادگیری با نظارت معطوف سازیم، اما توجه داشته باشید این حرف به معنای آن نیست که فرایند کار کردن با الگوریتم‌های یادگیری بدون نظارت مفید نیست. بلکه واقعیت این است که به کارگیری الگوریتم‌های بدون ناظر به دلیل این‌که نیازی به برجسب‌گذاری داده‌های آموزشی ندارند، کمی ساده‌تر است.

## مطلب پیشنهادی



هوشمندسازی ارتباطات نسل پنجم  
یادگیری ماشینی در خدمت شبکه‌های 5G

## آیا توانایی برآورد قیمت یک‌خانه برای یک الگوریتم، نوعی از یادگیری است؟

پاسخ‌گویی به این پرسش به مثالی نیاز دارد. مغز انسان توانایی شگفت‌انگیزی دارد که قادر است با وضعیت‌های مختلف روبه‌رو شده و بی‌نیاز به دستورالعمل خاصی یاد بگیرد که چطور باید با پیشامد روبه‌رو شده برخورد کند. به عبارت دیگر، از هر وضعیتی یک نکته جدید یاد گرفته و در موقعیت‌های مشابه قادر است تصمیماتی آبی اتخاذ کند. به همین دلیل است، افرادی که در زمینه فروش خانه به فعالیت اشتغال دارند، پس از گذشت مدت‌زمان مشخصی می‌توانند به شکل غریزی قیمت یک‌خانه را حدس زده و بهترین قیمت را پیشنهاد داده و به بهترین شکل مشتریانی را که ممکن است به یک‌خانه علاقه‌مند باشند شناسایی کنند. دنیای **هوش مصنوعی** به دنبال دستیابی به چنین سطح از مهارت است. اما در مقطع فعلی الگوریتم‌های **یادگیری ماشین** هنوز این چنین مهارتی را به دست نیاورده‌اند و

تنها زمانی بهترین عملکرد را دارند که روی یکسری مسائل محدود و مشخص متمرکز شده باشند. پس باید بگویم در مقطع فعلی الگوریتم‌ها به دنبال پیدا کردن یک معادله برای حل مشکلی خاص بر مبنای یکسری داده‌های نمونه هستند. اما به شما اطمینان می‌دهم در 30 تا 50 سال آینده زمانی که این مقاله را مطالعه کنید، از خواندن این مطلب کاملاً شگفت‌زده خواهید شد که ما در سال 97 در مورد **هوش مصنوعی** چه دانشی در اختیار داشتیم!

## تخمین قیمت یک‌خانه از طریق به‌کارگیری رگرسیون خطی Multivariate

تصور کنید به دنبال نوشتن برنامه‌ای هستیم که بتواند یک برآورد کلی در یک‌خانه بر مبنای اطلاعاتی که از فروش خانه‌های قبلی در اختیار دارد ارائه کند. اما در عین حال هیچ‌گونه پیش‌زمینه‌ای درباره الگوریتم‌های **یادگیری ماشین** نداریم. پس مجبور هستیم بر مبنای یکسری دستورات عمل‌های پایه تلاش کنیم تا برنامه‌ای برای تخمین قیمت یک‌خانه بر مبنای زبان محبوب پایتون ارائه کنیم. دانش لازم در مورد زبان پایتون در این زمینه لازم است!

```
def estimate_house_sales_price(num_of_bedrooms, sqft, neighborhood):
    price = 0

    # در محدوده من، متوسط قیمت هر خانه 200 دلار در هر فوت مربع است

    price_per_sqft = 200
    if neighborhood == "hipsterton":

        # در برخی مناطق هزینه‌ها کمی بیشتر است

        price_per_sqft = 400
        elif neighborhood == "skid row":

            # و در برخی مناطق هزینه‌ها کمتر است

            price_per_sqft = 100

            # کار را با برآورد یک قیمت پایه بر اساس بزرگی خانه آغاز می‌کنیم

            price = price_per_sqft * sqft

            # اکنون برآورد خود را بر مبنای تعداد اتاق‌خواب‌ها دقیق‌تر می‌کنیم

            if num_of_bedrooms == 0:

                # آپارتمان‌های استودیویی قیمت ارزان‌تری دارند

                price = price - 20000
            else:

                # خانه‌هایی که اتاق‌های بیشتری دارند
                # به‌طور معمول با ارزش‌تر هستند

                price = price + (num_of_bedrooms * 1000)
    return price
```

فهرست 1

اگر ساعت‌های متمادی از وقت خود را صرف قطعه کدی کنید که در فهرست شماره یک مشاهده می‌کنید، این احتمال وجود دارد که بتوانید نتیجه مدنظر خود را برای برآورد تخمینی قیمت چند خانه به دست آورید، اما بدون شک برنامه فوق کامل نیست و با یک نوسان قیمت‌ها اطلاعات اشتباهی را ارائه خواهد کرد. اما اگر به کامپیوتر یا به

عبارت دقیق‌تر به برنامه خود قابلیت‌هایی بدهیم تا به وسیله آن‌ها بتواند خودش نحوه پیاده‌سازی این تابع را کشف کند، آنگاه اوضاع به گونه دیگری رقم خواهد خورد. در چنین شرایطی تا زمانی که تابع موردنظر نتیجه ایده‌آل ما را در خروجی ارائه کند، نحوه کارکرد آن برای ما اهمیت چندانی نخواهد داشت.

```
def estimate_house_sales_price(num_of_bedrooms, sqft, neighborhood):  
    price = <computer, plz do some math for me>  
    return price
```

فهرست 2

یک راهکار ایده‌آل که به کامپیوتر این توانایی را بدهد تا قابلیت تشخیص و پیاده‌سازی تابعی را داشته باشد این است که قیمت خانه را به عنوان یک معیار تصور کنیم که برای برآورد آن به یکسری المان‌ها همچون تعداد اتاق خواب‌ها، مساحت خانه و محله خانه نیاز است. همچنین باید بررسی کنیم که هر یک از این المان‌ها تا چه اندازه بر قیمت نهایی یک‌خانه اثرگذار خواهند بود، زیرا این احتمال وجود دارد که نسبت یا به عبارت دقیق‌تر، تناسبی مشخصی میان ترکیب این المان‌ها به منظور برآورد قیمت نهایی خانه وجود داشته باشد. این طرح به ما اجازه می‌دهد تابع نشان داده شده در فهرست (1) را از دست طیف بی‌شماری از دستورات شرطی if و else نجات داده و کدها را ساده‌تر کنیم.

```
def estimate_house_sales_price(num_of_bedrooms, sqft, neighborhood):  
    price = 0
```

# کمی خرج می‌کنیم

```
price += num_of_bedrooms * .841231951398213
```

# و کمی بیشتر از آن را خرج می‌کنیم

```
price += sqft * 1231.1231231
```

# شاید این تعداد انگشت‌شمار باشد

```
price += neighborhood * 2.3242341421
```

# و در نهایت کمی نمک اضافه می‌کنیم تا مقیاس اندازه‌گیری ما بهتر شود!!

```
price += 201.23432095
```

```
return price
```

فهرست 3

به ارقامی که در فهرست (3) برجسته شده‌اند، دقت کنید. این اعداد همان وزن‌های مدل هستند که از طریق ترکیب عناصری که در قیمت خانه اثرگذار هستند، به دست آمده‌اند. پس اگر در محاسبه وزن‌ها دقت زیادی به خرج دهیم، تابع قادر خواهد بود قیمت نهایی را برآورد کند. راهکار ساده‌ای برای شناسایی بهترین وزن‌ها برای مدل به شرح زیر وجود دارد. ادامه این مبحث را در شماره آینده از سر خواهیم گرفت.

**تاریخ انتشار:**

نشانی منبع:

[https://www.shabakeh-mag.com/information-feature/artificial-intelligence/14096/%DA%86%DA%A  
F%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](https://www.shabakeh-mag.com/information-feature/artificial-intelligence/14096/%DA%86%DA%A<br/>F%D9%88%D9%86%D9%87-<br/>%D9%85%DB%8C%E2%80%8C%D8%AA%D9%88%D8%A7%D9%86%DB%8C%D9%85-<br/>%D8%AF%D8%B1-%D8%B9%D9%85%D9%84-%D8%A7%D8%B2-<br/>%DB%8C%D8%A7%D8%AF%DA%AF%DB%8C%D8%B1%DB%8C-<br/>%D9%85%D8%A7%D8%B4%DB%8C%D9%86-<br/>%D8%A7%D8%B3%D8%AA%D9%81%D8%A7%D8%AF%D9%87)