

Soft computing



K.N. Toosi
University of
Technology


محاسبات نرم



Hasan Ghasemzadeh
<http://wp.kntu.ac.ir/ghasemzadeh>

Soft Computing

Soft computing



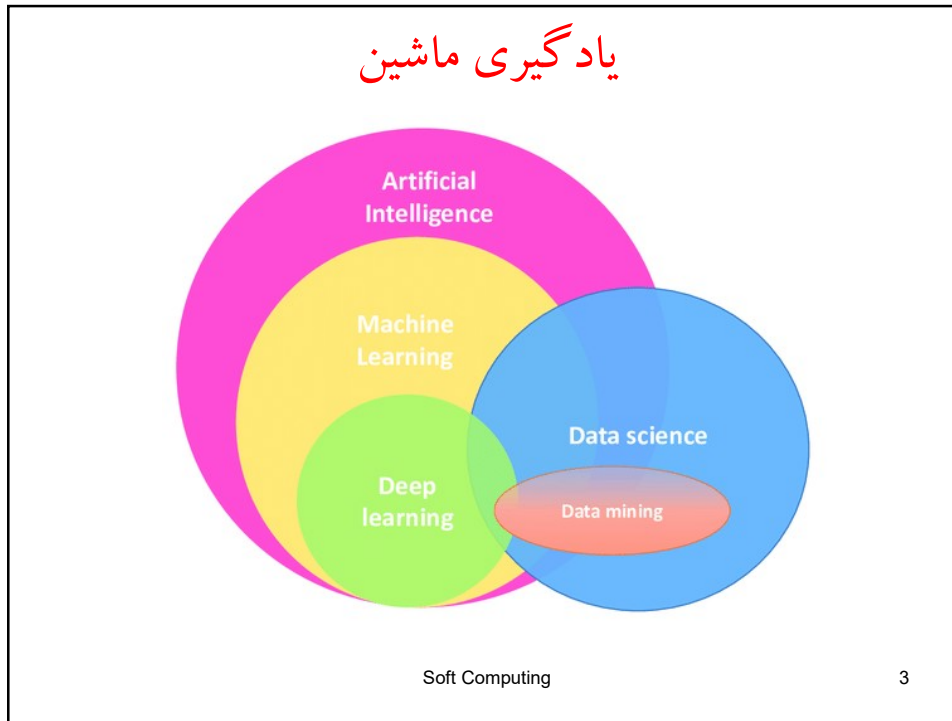
K.N. TOOSI
University of
Technology

یادگیری ماشین



Soft Computing

2



یادگیری ماشین

تعریف آرتور ساموئل ۱۹۵۹

- حوزه مطالعاتی که به رایانه قابلیت یادگیری می دهد بدون آنکه برنامه نویسی صریح شود

تعریف جدید تام میشل ۱۹۹۸

- یک برنامه رایانه ای از تجربه E نسبت به کلاس وظایف T و بازدهی P یاد می گیرد اگر بازدهی P وظایف T با تجربه E بهبود یابد

برای مثال در بازی شطرنج توسط رایانه

E: تجربه بازی های متعدد بازیکنان

T: وظیفه بازی بازیکنان

P: احتمال اینکه برنامه بازی بعدی را ببرد

Soft Computing 4

یادگیری ماشین

الگوریتمهای یادگیری ماشین به سه دسته تقسیم می شوند

- یادگیری نظارت شده
- یادگیری بدون نظارت
- یادگیری تقویتی



یادگیری نظارت شده: مجموعه ای از داده ها داریم و میدانیم نتیجه خروجی مناسب چگونه است (استاد برای تصحیح وجود دارد).

یادگیری نظارت شده به دو دسته اصلی تقسیم می شود

- رگرسیون
- طبقه بندی

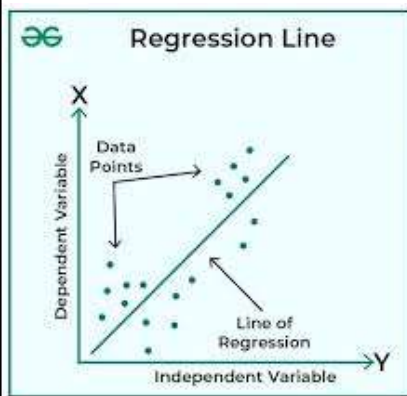
Soft Computing

5

یادگیری نظارت شده - رگرسیون

رگرسیون

رگرسیون یک نوع از یادگیری نظارت شده است که در آن الگوریتم یاد می گیرد بر اساس ویژگی های ورودی، مقادیری پیوسته را پیش بینی کند. مثال هایی از مسائل رگرسیون شامل پیش بینی قیمت سهام و قیمت مسکن می شوند.



الگوریتم های معروف:

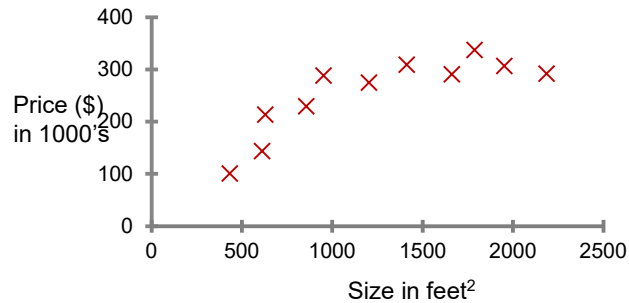
- رگرسیون خطی
- رگرسیون چندجمله ای
- رگرسیون ریج
- رگرسیون درخت تصمیم
- رگرسیون جنگل تصادفی
- رگرسیون ماشین بردار پشتیبان

Soft Computing

6

یادگیری نظارت شده - رگرسیون

Housing price prediction.



Supervised Learning
"right answers" given

Regression: Predict
continuous valued output
(price)

Soft Computing

7

یادگیری نظارت شده - نمونه ژئومکانیکی رگرسیون

• پیش بینی فشار منفذی:

با استفاده از ویژگی‌های مختلف نظیر عمق، سرعت امواج لرزه‌ای، مقاومت الکتریکی، چگالی سنگ

• پیش بینی تنش‌های درون سنگ:

با داشتن داده‌های مختلف از ویژگی‌های عمق، خواص مکانیکی سنگ (مانند مدول یانگ و نسبت پواسون) و داده‌های لرزه‌ای

• پیش بینی نفوذپذیری سنگ:

با تحلیل داده‌های مربوط به تخلخل، اندازه دانه‌ها، داده‌های چاه‌نگاری

• پیش بینی مقاومت فشاری سنگ:

با داشتن داده‌های سرعت امواج لرزه‌ای، چگالی سنگ، داده‌های چاه‌نگاری

• پیش بینی رفتار سنگ در شرایط فشار و دمای بالا:

با تحلیل پارامترهای مختلف مانند دما، فشار، خواص مکانیکی سنگ

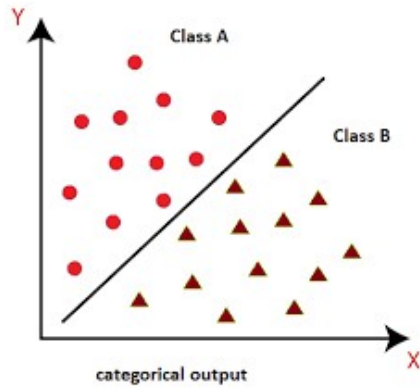
Soft Computing

8

یادگیری نظارت شده - طبقه بندی

طبقه بندی

الگوریتم یاد می گیرد داده های ورودی را بر اساس ویژگی های ورودی به یک دسته یا کلاس خاص تخصیص دهد. در طبقه بندی، برچسب های خروجی مقادیر گسسته هستند.



الگوریتم های معروف:
 رگرسیون لجستیک
 نایو بیس
 درخت تصمیم
 ماشین بردار پشتیبان (SVM)
 همسایگان نزدیک (KNN)

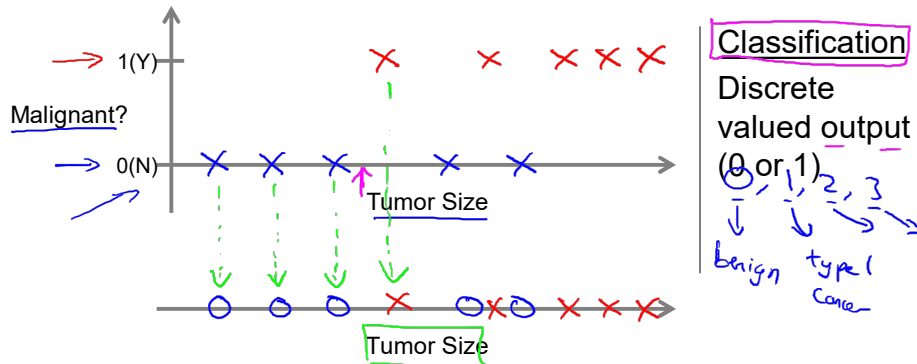
Soft Computing

9

یادگیری نظارت شده - طبقه بندی

طبقه بندی فرایندی است که در آن مدل یا تابعی کشف می شود که به جدا کردن داده ها به چندین کلاس دسته بندی، یعنی مقادیر گسسته، کمک می کند.

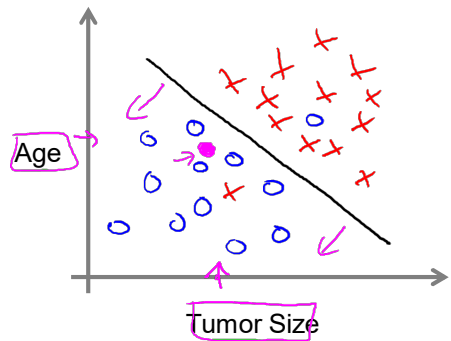
Breast cancer (malignant, benign)



Soft Computing

10

یادگیری نظارت شده - طبقه بندی



- Clump Thickness
- Uniformity of Cell Size
- Uniformity of Cell Shape

...

Soft Computing

11

یادگیری نظارت شده - مثال

You're running a company, and you want to develop learning algorithms to address each of two problems.

Problem 1: You have a large inventory of identical items. You want to predict how many of these items will sell over the next 3 months.

Problem 2: You'd like software to examine individual customer accounts, and for each account decide if it has been hacked/compromised.

Should you treat these as classification or as regression problems?

Treat both as classification problems.

Treat problem 1 as a classification problem, problem 2 as a regression problem.

Treat problem 1 as a regression problem, problem 2 as a classification problem.

Treat both as regression problems.

یادگیری نظارت شده - نمونه عمرانی طبقه بندی

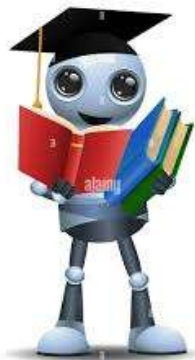
- **طبقه بندی نوع سنگ:**
با تحلیل داده‌های چاه‌نگاری مانند مقاومت الکتریکی، تخلخل، چگالی و داده‌های لرزه‌ای مانند سرعت امواج لرزه‌ای.
- **تشخیص شکستگی‌های طبیعی:**
با استفاده از داده‌های لرزه‌ای و چاه‌نگاری.
- **طبقه بندی پایداری چاه:**
با استفاده از اطلاعات تنش‌های درون‌سنگ، خواص مکانیکی سنگ و فشار منفذی.
- **تشخیص نواحی با خطر فوران چاه:**
با استفاده از اطلاعات فشار منفذی، فشار هیدرواستاتیک، داده‌های حین حفاری.
- **تشخیص طبقات سنگ‌های معدنی در حفاری‌های عمیق:**
با استفاده از داده‌های حاصل از دستگاه‌های حفاری و اطلاعات ژئوفیزیکی، مدل‌های طبقه‌بندی می‌توانند طبقات سنگ‌های مختلف (مانند سنگ‌های آذرین، رسوبی و دگرگونی) را شناسایی و دسته‌بندی کنند.

Soft Computing

13

یادگیری بدون نظارت

یادگیری بدون نظارت: ایده‌ای از نتیجه خروجی نداریم. در این روش ساختار یا توزیع مخفی داده‌ها بدون آنکه تاثیر متغیرها را بدانیم بدست می‌آید. در این حالت بازخوردی برای نتایج صحیح وجود ندارد (استاد برای تصحیح وجود ندارد).



در یادگیری بدون نظارت دو دسته اصلی وجود دارد:

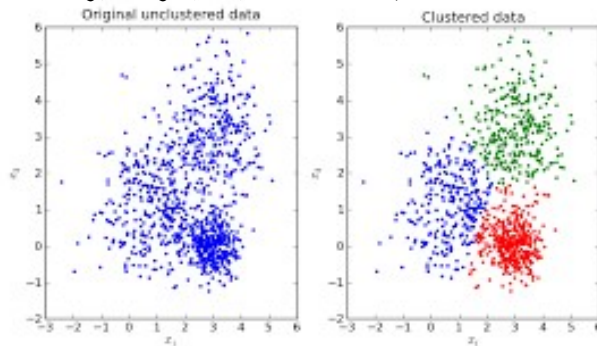
- خوشه‌بندی (Clustering)
- کاهش ابعاد (Dimensionality Reduction)

Soft Computing

14

یادگیری بدون نظارت

- خوشه‌بندی (Clustering): الگوریتم‌های خوشه‌بندی، داده‌های مشابه را بر اساس ویژگی‌های آنها در گروه‌هایی متفاوت تقسیم می‌کنند. هدف اینجا شناسایی گروه‌ها یا خوشه‌های داده‌های مشابه است که از یکدیگر متمایز هستند.



الگوریتم‌های معروف:

- K-means
- خوشه‌بندی سلسله مراتبی (Hierarchical clustering)
- DBSCAN

Soft Computing

15

یادگیری بدون نظارت

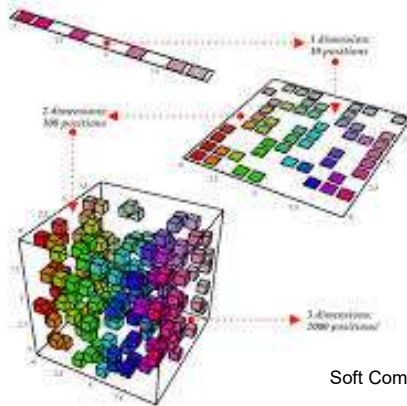
- خوشه‌بندی خاک‌ها بر اساس ویژگی‌های ژئوتکنیکی:
- با استفاده از داده‌های آزمایش‌های ژئوتکنیکی مانند درصد دانه‌بندی، تراکم، چسبندگی، و ضریب نفوذپذیری، می‌توان انواع مختلف خاک‌ها را در خوشه‌های متفاوت دسته‌بندی کرد.
- خوشه‌بندی داده‌های حین حفاری:
- با جمع‌آوری داده‌های مربوط به نرخ حفاری، فشار حفاری و لرزش حین حفاری
- خوشه‌بندی نواحی با پتانسیل تولید بالا:
- با تحلیل داده‌های مربوط به تخلخل، نفوذپذیری و درجه اشباع سیال.
- خوشه‌بندی داده‌های لرزه‌ای برای شناسایی ساختارهای زمین‌شناسی:
- گروه‌بندی داده‌های لرزه‌ای برای شناسایی ساختارهای زمین‌شناسی مانند گسل‌ها، چین‌خوردگی‌ها و مخازن
- خوشه‌بندی شکستگی‌های طبیعی:
- گروه‌بندی شکستگی‌های طبیعی بر اساس ویژگی‌هایی مانند جهت، اندازه و تراکم

Soft Computing

16

یادگیری بدون نظارت

- کاهش ابعاد (Dimensionality Reduction): الگوریتم‌های کاهش ابعاد با حفظ حداکثر اطلاعات اصلی از داده‌ها تعداد متغیرهای ورودی در مجموعه داده را کاهش می‌دهند. این کار برای کاهش پیچیدگی مجموعه داده و تسهیل در تصویرسازی و تجزیه و تحلیل آن است.



الگوریتم‌های معروف:

- تجزیه اصلی مؤلفه‌ها (PCA)
- t-SNE
- Autoencoders

Soft Computing

17

یادگیری بدون نظارت - نمونه کاهش ابعاد

- تحلیل داده‌های سنسورهای لرزه‌نگاری:**
- داده‌های سنسورهای لرزه‌نگاری بسیار حجیم و با ابعاد بالا هستند. با استفاده از کاهش ابعاد، می‌توان این داده‌ها را به مولفه‌های اصلی تقسیم کرد تا تنها اطلاعات مهم برای تحلیل و پیش‌بینی رفتار زمین لرزه‌ها استخراج شود.
- کاهش ابعاد داده‌های چاه‌نگاری:**
- داده‌های چاه‌نگاری شامل چندین نوع اندازه‌گیری (مانند مقاومت الکتریکی، تخلخل، چگالی) هستند. کاهش ابعاد این داده‌ها می‌تواند برای تحلیل‌های بعدی مانند طبقه‌بندی نوع سنگ یا پیش‌بینی خواص مخزن مفید باشد.
- کاهش ابعاد داده‌های تصویربرداری چاه:** داده‌های تصویربرداری چاه معمولاً ابعاد بسیار بالایی دارند. کاهش ابعاد این داده‌ها می‌تواند برای شناسایی شکستگی‌ها و ساختارهای زمین‌شناسی مفید باشد.
- تحلیل داده‌های جوی برای پیش‌بینی فرسایش خاک:**
- داده‌های هواشناسی، مانند بارش، دما و باد، برای ارزیابی فرسایش خاک مهم هستند. با کاهش ابعاد، می‌توان متغیرهای مهم را شناسایی کرد و از آن‌ها برای پیش‌بینی فرسایش خاک در مناطق مستعد استفاده کرد.

Soft Computing

18

یادگیری بدون نظارت - مثال

Of the following examples, which would you address using an unsupervised learning algorithm? (Check all that apply.)

- Given email labeled as spam/not spam, learn a spam filter.
- Given a set of news articles found on the web, group them into set of articles about the same story.
- Given a database of customer data, automatically discover market segments and group customers into different market segments.
- Given a dataset of patients diagnosed as either having diabetes or not, learn to classify new patients as having diabetes or not.

Soft Computing

19

یادگیری تقویتی

یادگیری تقویتی: در آن یک عامل، با انجام اعمالی در سیستم مشخص شده، و دریافت پاداش یا مجازات بر اساس اعمال خود، یاد می‌گیرد که چگونه با محیط تعامل کند. هدف اصلی یادگیری ماشینی تقویتی این است که یک سیاست (که یک نگاهت از وضعیت‌ها به اعمال است) را یاد بگیرد و میزان پاداش تجمعی انتظاری را در طول زمان به حداکثر برساند.

یادگیری تقویت شده به دو دسته اصلی تقسیم می‌شود:

- یادگیری تقویتی مبتنی بر مدل

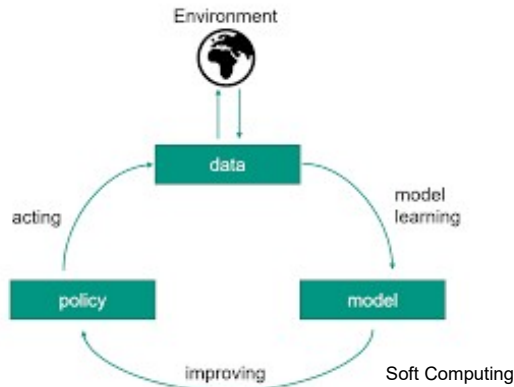
- یادگیری تقویتی بدون مدل

Soft Computing

20

یادگیری تقویتی

- یادگیری تقویتی مبتنی بر مدل: عامل یک مدل از محیط را یاد می‌گیرد که شامل احتمالات انتقال بین وضعیت‌ها و پاداش‌های مرتبط با هر جفت وضعیت-عمل است. سپس عامل از این مدل برای برنامه‌ریزی اعمال خود به منظور به حداکثر رساندن پاداش انتظاری استفاده می‌کند.



الگوریتم‌های معروف:

- Value Iteration
- Policy Iteration

21

یادگیری تقویتی - نمونه‌ها

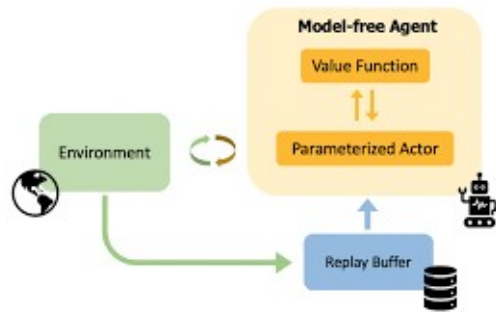
- **مدیریت هوشمند مصرف انرژی در ساختمان‌ها:**
در ساختمان‌های هوشمند، یادگیری تقویتی می‌تواند بهینه‌سازی مصرف انرژی را از طریق تنظیم سیستم‌های گرمایش، سرمایش و نورپردازی انجام دهد.
- **بهینه‌سازی زمان‌بندی و توالی ساخت در پروژه‌های ساختمانی:**
در پروژه‌های پیچیده ساختمانی، مدیریت زمان و توالی کارها از اهمیت بالایی برخوردار است. یادگیری تقویتی می‌تواند برای پیدا کردن بهینه‌ترین زمان‌بندی کارها با توجه به محدودیت‌های منابع، هزینه‌ها و زمان اجرا به کار رود، به طوری که بتواند تأخیرها و هزینه‌های اضافی را به حداقل برساند.
- **طراحی سیستم‌های مدیریت آب و کنترل سیلاب:**
برای پیش‌بینی و کنترل سیلاب‌ها، یادگیری تقویتی می‌تواند در طراحی و مدیریت بهینه سیستم‌های تخلیه و مدیریت آب به کار رود. مدل می‌تواند بر اساس داده‌های بارش و سطح آب رودخانه‌ها، تصمیم‌گیری‌های بهینه‌ای برای تخلیه آب و جلوگیری از سیلاب انجام دهد.
- **بهینه‌سازی الگوهای بهره‌برداری از منابع طبیعی (مانند معادن):**
در مدیریت و بهره‌برداری از معادن، یادگیری تقویتی می‌تواند بهینه‌سازی برداشت منابع را انجام دهد. مدل با توجه به اطلاعات مربوط به منابع، هزینه‌ها، و شرایط بازار، بهترین تصمیمات را برای استخراج بهینه منابع ارائه می‌دهد و از هدررفت منابع جلوگیری می‌کند.

Soft Computing

22

یادگیری تقویتی بدون مدل

- یادگیری تقویتی بدون مدل: عامل بدون ایجاد مدل دقیقی از محیط به صورت مستقیم از تجربیات خود سیاست خود را یاد می گیرد. عامل با تعامل با محیط و دریافت پاداش ها سیاست خود را به روزرسانی می کند.



الگوریتم های معروف:

- Q-Learnin
- SARSA
- یادگیری تقویتی عمیق (Deep Reinforcement Learning)

Soft Computing

23

یادگیری تقویتی بدون مدل نمونه ژئومکانیکی

- **کنترل و هدایت ماشین آلات سنگین در سایت های ساخت و ساز:**
یادگیری تقویتی بدون مدل می تواند برای هدایت ماشین آلات خود کار (مانند بولدوزرها یا بیل مکانیکی) استفاده شود. این مدل ها به مرور زمان از تعاملات خود با محیط یاد می گیرند که چگونه با بهره وری بیشتری عمل کنند و مسیرها یا الگوهای بهینه برای حرکت و حفاری را پیدا کنند. مثلاً یک بیل مکانیکی می تواند یاد بگیرد که با کمترین انرژی و در سریع ترین زمان، خاک را بردارد.
- **بهینه سازی روش های حفاری در پروژه ها:**
در حفاری های پیچیده و عمیق، یادگیری تقویتی بدون مدل می تواند روش های بهینه حفاری را بر اساس داده های فوری از شرایط زمین و رفتار دستگاه یاد بگیرد. این روش به دستگاه کمک می کند که با بهره وری بالاتری کار کند و با توجه به تجربیات خود، بهترین پارامترهای حفاری مانند سرعت و زاویه را تنظیم کند...
- **کنترل عملیات رباتیک در محیط پروژه های خطرناک:**
یادگیری تقویتی بدون مدل می تواند برای کنترل ربات های خود کار در محیط های ساخت و ساز خطرناک (مانند مناطق با خطر ریزش، اشتعال و فوران چاه) استفاده شود. این ربات ها از تعاملات خود با محیط یاد می گیرند که چگونه به صورت ایمن و بهینه کار کنند و به تدریج استراتژی های خود را برای اجرای بهتر عملیات تنظیم می کنند.

Soft Computing

24

چالش‌ها و محدودیت‌های یادگیری ماشین

- ✓ چالش اصلی در یادگیری ماشین، کمبود داده یا تنوع کم در مجموعه داده‌ها است.
- ✓ اگر داده‌های کافی در دسترس نباشد یک ماشین نمی‌تواند یاد بگیرد. علاوه بر این، یک مجموعه داده با تنوع کم، باعث ایجاد مشکل برای ماشین می‌شود.
- ✓ برای یادگیری معنادار، یک ماشین نیاز به تنوع در داده‌ها دارد.
- ✓ وقتی تنوع در داده‌ها کم باشد یا اصلاً وجود نداشته باشد، بسیار نادر است که یک الگوریتم بتواند اطلاعات معنی‌داری استخراج کند.
- ✓ توصیه می‌شود حداقل ۲۰ مشاهده در هر گروه وجود داشته باشد تا به ماشین در یادگیری کمک کند. این محدودیت منجر به ارزیابی و پیش‌بینی نادرست می‌شود.

Soft Computing

25

توزیع نرمال

Normal Distribution

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

import numpy as np

import matplotlib.pyplot as plt

from scipy.stats import chi2square, norm

• $f(x)$ مقدار تابع توزیع نرمال برای مقدار x است.• μ میانگین (mean) توزیع است.• σ انحراف معیار (standard deviation) توزیع است.

class norm1:

def __init__(self, mean, sd, x):

self.mean = mean

self.sd = sd

self.x = x

def dist_curve(self):

plt.plot(self.x, 1/(self.sd * np.sqrt(2 * np.pi)) *

np.exp(- (self.x - self.mean)**2 / (2 * self.sd**2)),

linewidth=2, color='y')

Soft Computing

26

توزیع نرمال

```
# پارامترهای مربوط به میانگین و انحراف معیار
mean1 = 5
sd1 = 2

# تولید داده‌های تصادفی نرمال
c = np.random.normal(mean1, sd1, 3000)

# رسم هیستوگرام
count, bins, _ = plt.hist(c, bins=100, density=True, alpha=0.5,
color='b') # هیستوگرام با نرمال‌سازی و شفافیت

# برای رسم منحنی نرمال #bins محاسبه نقاط میانی بین
bin_centers = 0.5 * (bins[1:] + bins[:-1])

# رسم منحنی نرمال
hist1 = norm1(mean1, sd1, bin_centers)  $f(x) = \frac{1}{2\sqrt{2\pi}} \exp\left(-\frac{(x-5)^2}{2 \times 2^2}\right)$ 
hist1.dist_curve()

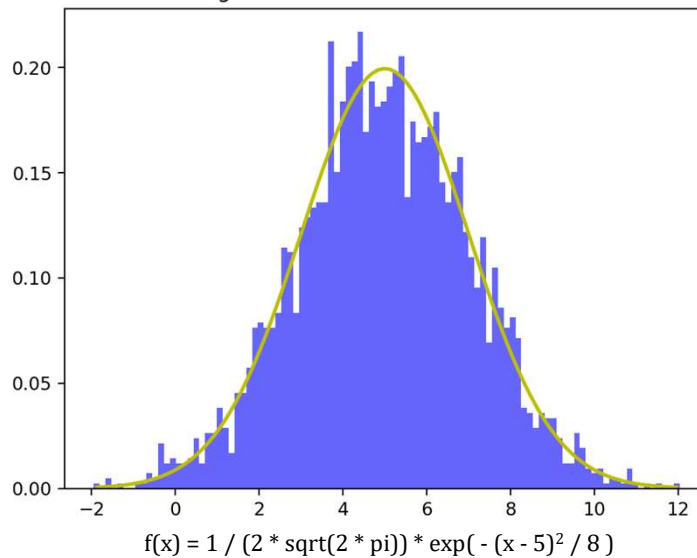
# نمایش نمودار
plt.title('Histogram and Normal Distribution Curve')
plt.show()
```

Soft Computing

27

توزیع نرمال

Histogram and Normal Distribution Curve



28

آزمون توزیع

$$\chi^2 = \sum \frac{(F_o - F_e)^2}{F_e}$$

χ^2 : شاخص کای اسکویئر

F_o : فراوانی مشاهده شده

F_e : فراوانی مورد انتظار می باشد

محاسبه مقادیر پیش‌بینی شده برای توزیع نرمال

```
from scipy.stats import chisquare
expected_counts = np.array([
    3000 * (norm.cdf(bins[i + 1], mean1, sd1) - norm.cdf(bins[i], mean1, sd1))
    for i in range(len(bins) - 1) ])
# نرمال‌سازی مقادیر پیش‌بینی‌شده برای تطابق با مجموع مقادیر مشاهده‌شده
expected_counts *= count.sum() / expected_counts.sum()
# آزمون کای-دو برای بررسی تطابق داده‌ها با توزیع نرمال
chi_statistic, p_value = chisquare(f_obs=count, f_exp=expected_counts)
# نمایش نتایج آزمون کای-دو
print("Chi square test:", chi_statistic)
print("p value:", p_value)

# نتیجه‌گیری
alpha = 0.05 # سطح معناداری
if p_value > alpha:
    print("The data fit a normal distribution. P value>.05")
else:
    print("The data does not fit a normal distribution. P value<.05 ")
Chi square test: 0.23046991808461853
p value: 1.0
The data fit a normal distribution. P value>.05
```

29

آزمون توزیع

لیست توزیع‌های مختلف برای برازش

```
from scipy.stats import chisquare, norm, expon, lognorm, gamma, beta, chi2
distributions = [norm, expon, lognorm, gamma, beta, chi2]
# تابع برای محاسبه معیار برازش (کای-دو) برای هر توزیع
def fit_distribution(data, bins):
    best_p_value = 0
    best_distribution = None
    best_params = None

    for distribution in distributions:
        params = distribution.fit(data)
        expected_counts = np.array([
            3000 * (distribution.cdf(bins[i + 1], *params) -
            distribution.cdf(bins[i], *params))
            for i in range(len(bins) - 1) ])
        expected_counts *= count.sum() / expected_counts.sum() # نرمال‌سازی
        chi_stat, p_value = chisquare(f_obs=count, f_exp=expected_counts)
        chi_stat, p_value = chisquare(f_obs=count, f_exp=expected_counts)
        print(f"{distribution.name} - p_value: {p_value}")
        if p_value > best_p_value:
            best_p_value = p_value
            best_distribution = distribution
            best_params = params

    return best_distribution, best_params, best_p_value
```

Soft Computing

30

آزمون توزیع

```
#برازش توزیع های مختلف و انتخاب بهترین توزیع
best_distribution, best_params, best_p_value = fit_distribution(c,
bins)
```

```
#نمایش بهترین توزیع
print(f"\nBest fitting distribution: {best_distribution.name}")
print(f"Best parameters: {best_params}")
print(f"Best p-value: {best_p_value}")
```

```
norm - p_value: 1.0
expon - p_value: 1.0
C:\Users\ASUS\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.12_qbz5n
istns.py:6617: RuntimeWarning: invalid value encountered in log
  lndata = np.log(data - loc)
lognorm - p_value: 1.0
gamma - p_value: 1.0
beta - p_value: 1.0
chi2 - p_value: 1.0

Best fitting distribution: norm
Best parameters: (np.float64(4.960499899240569), np.float64(2.018605804698402))
Best p-value: 1.0
```

تمرین برنامه نویسی

تمرین چهارم: یک برنامه به زبان پایتون بنویسید که n داده را بگیرد و توزیع آنها را تعیین کند

۱- خواندن داده ها از فایل اکسل

۲- تعیین پارامترهای تابع توزیع احتمال داده ها به روشهای ذیل

- نرمال
- نمایی
- لگاریتم نرمال
- گاما
- بتا
- کای دو

۳- با رسم هیستوگرام داده ها و توابع توزیع، تابع مناسبتر کدام است.

۴- با استفاده از کتابخانه scipy در برنامه، بهترین تابع توزیع احتمال را تعیین کنید