

یادگیری ماشین



دانشگاه شهید بهشتی
پژوهشکده فضای مجازی
پاییز ۱۳۹۸
احمد محمودی ازناوه

فهرست مطالب

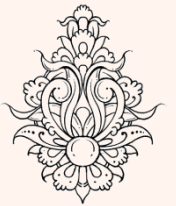
• روش‌های ناپارامتری

– تخمین چگالی

– دسته‌بندی

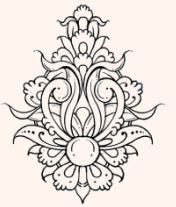
– رگرسیون

– تشخیص داده‌های برون‌هسته

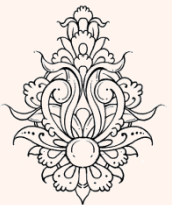


پیش‌گفتار

- «روش‌های پارامتری»: یک مدل برای تمام داده‌های ورودی در نظر گرفته می‌شود.
 - در رگرسیون خطی فرض می‌شود برای تمام ورودی‌ها، خروجی از یک تابع خطی یکسان تبعیت می‌کند.
 - هر چند تقلیل مسأله به یافتن چند پارامتر محدود، مطلوب است، اما ممکن است این فرض صحیح نباشد و باعث ایجاد خطا شود.
- «روش‌های نیمه‌پارامتری»: داده‌های هر دسته ترکیبی از مدل‌های مختلف تلقی می‌شوند.
- «روش‌های ناپارامتری»: ورودی‌های مشابه، خروجی‌های مشابه دارند.
 - «نمونه‌های مشابه» به معنای «چیزهای مشابه» هستند.
 - توابع، هموار هستند، تخیرات آن نرم است.



- الگوریتم‌های ناپارامتری، شامل یافتن نمونه‌های **نزدیک** (بر اساس یک تابع فاصله‌ی مناسب) و سپس **درون‌یابی** برای یافتن خروجی درست است.
- تفاوت‌های روش‌های ناپارامتری به انتخاب تابع فاصله و روش درون‌یابی بستگی دارد.
- در مدل‌های پارامتری کل داده‌های آموزشی بر مدل نهایی اثرگذار هستند، در حالی که در روش‌های ناپارامتری یک مدل محلی بر اساس نمونه‌های همسایه تخمین زده می‌شود.
- نیاز به حافظه و محاسبات بالا از عیب‌های این دسته از روش‌هاست.



lazy/memory-based/case-based/instance-based learning

• داده‌های آموزشی $X = \{x^t\}_{t=1}^N$ به صورت مستقل از یک توزیع تصادفی یکسان $p(x)$ استخراج شده‌اند.

• تخمین ناپارامتری «تابع توزیع تجمعی» به صورت زیر محاسبه می‌شود:

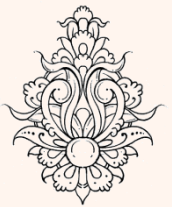
$$\hat{F}(x) = \frac{\#\{x^t \leq x\}}{N}$$

• و به همین ترتیب برای تابع چگالی احتمال

$$\hat{p}(x) = \frac{1}{h} \left[\frac{\#\{x^t \leq x+h\} - \#\{x^t \leq x\}}{N} \right]$$

• h طول بازه‌ای است که داده‌های آن به اندازه‌ی کافی به هم نزدیک (شبییه) هستند.

$$x \leq x^t \leq x+h$$

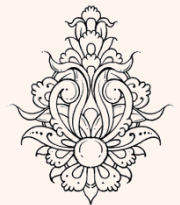


تخمین هیستوگرام (بافت نگار)

- تخمین هیستوگرام یکی از قدیمی‌ترین و متداول‌ترین روش‌هاست، داده‌ها به یک سری نوار (bin) با عرض h تقسیم می‌شوند:

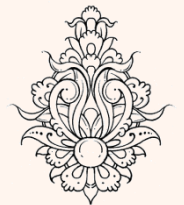
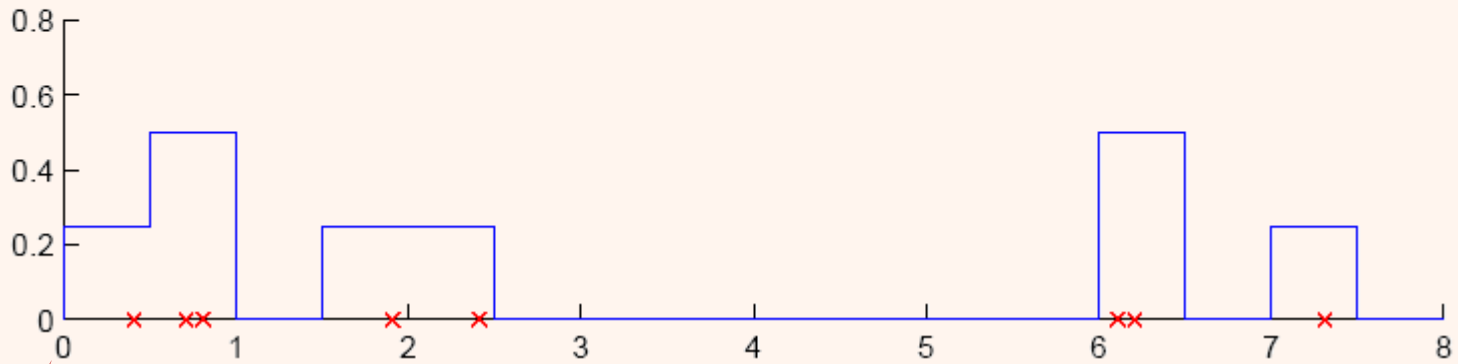
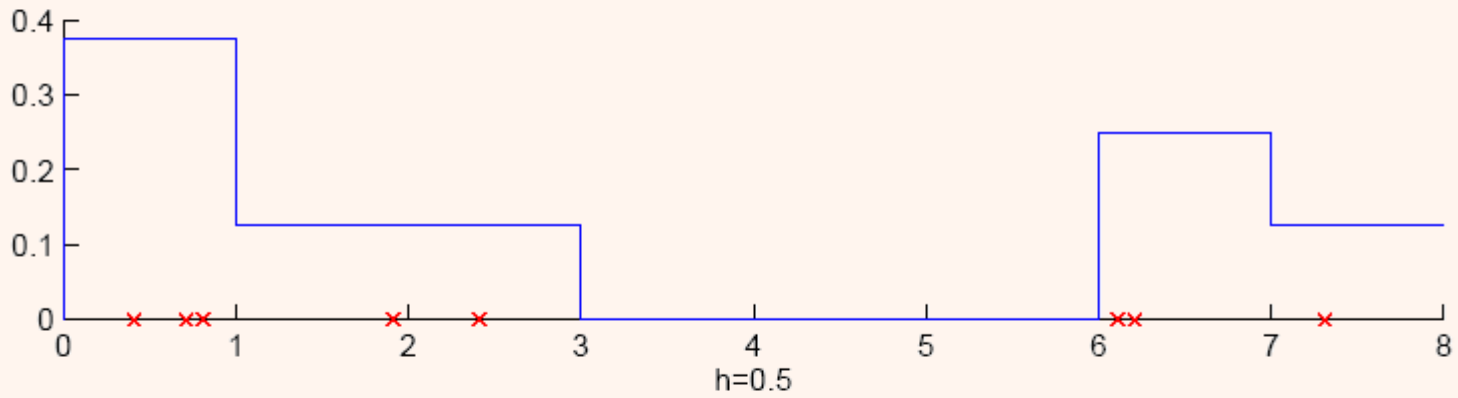
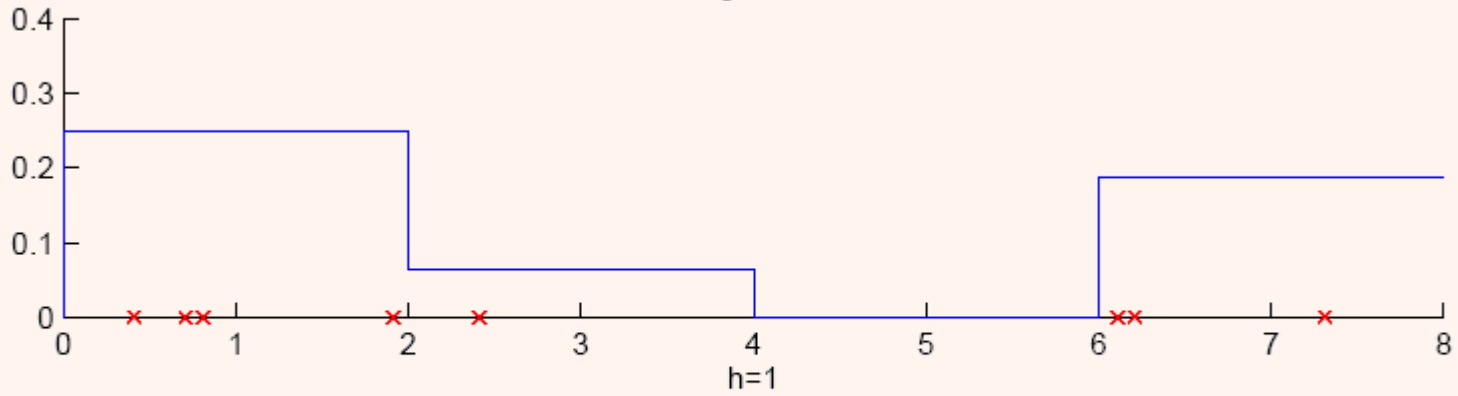
$$\hat{p}(x) = \frac{\#\{x^t \text{ in the same bin as } x\}}{Nh}$$

- تغییر عرض نوارها و مبدأ آنها بر روی تخمین به دست آمده اثرگذار است.
- در مرز نوارها گسستگی دیده می‌شود.
- نیازی به ذخیره کردن نمونه‌ها نیست.



مثال

Histogram: $h=2$



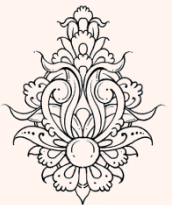
- در این روش نیازی به مشخص کردن مبدأ نوارها وجود ندارد:

$$\hat{p}(x) = \frac{\#\{x-h < x^t \leq x+h\}}{2Nh}$$

- یا با تعریف یک تابع وزن دهی:

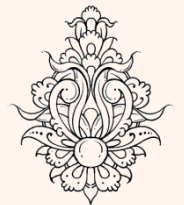
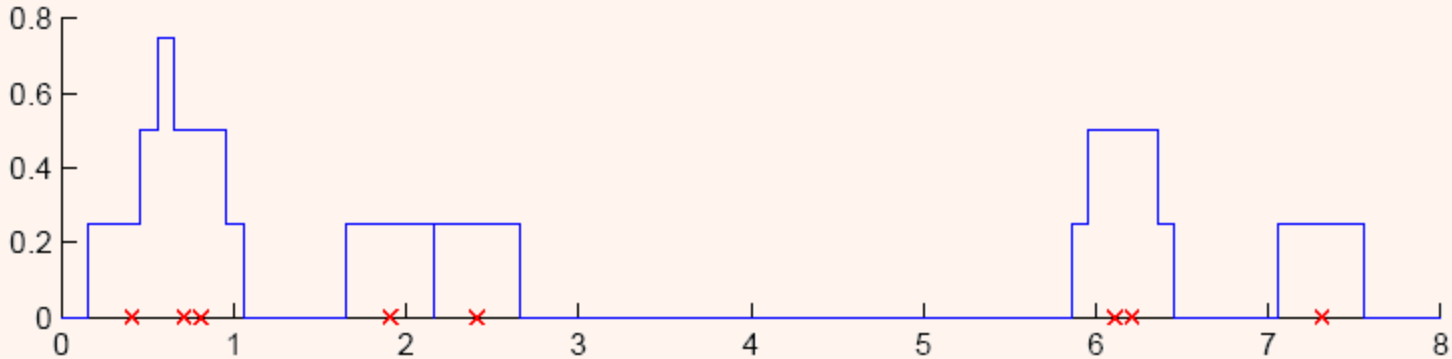
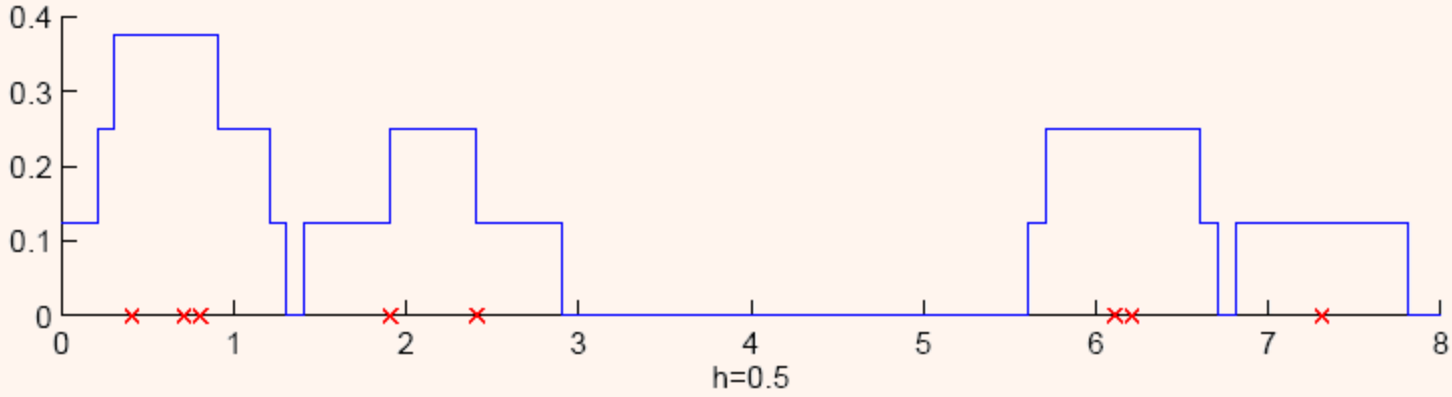
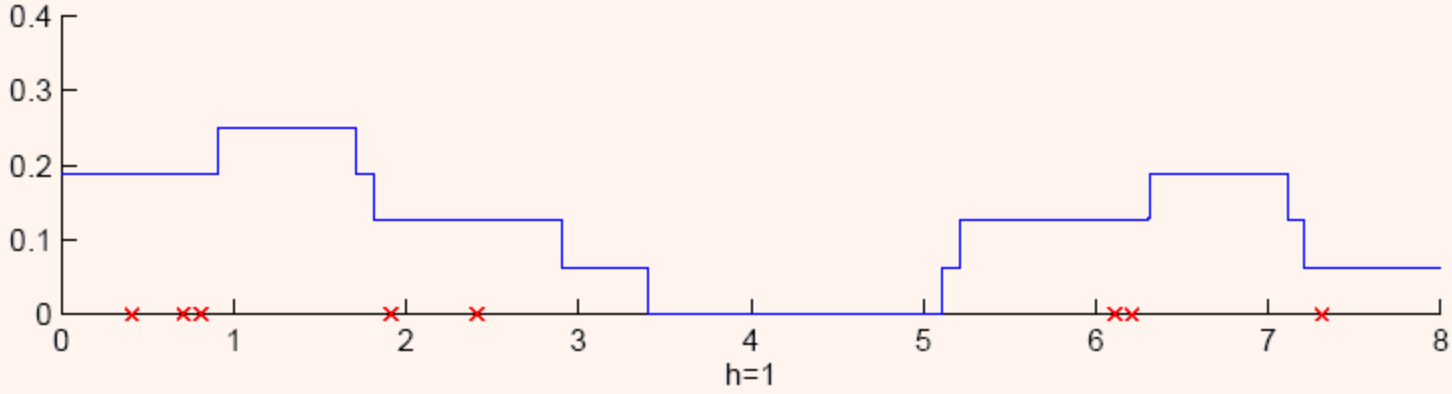
$$\hat{p}(x) = \frac{1}{Nh} \sum_{t=1}^N w\left(\frac{x-x^t}{h}\right) \quad w(u) = \begin{cases} 1/2 & \text{if } |u| < 1 \\ 0 & \text{otherwise} \end{cases}$$

- با توجه به نامیهی تحت تأثیر (hard)، تخمین به دست آمده، در نواحی مرزی دارای پرش می باشد.



مثال

Naive estimator: $h=2$

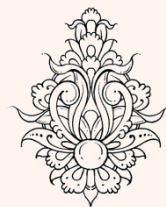


- برای به دست آوردن تخمینی هموارتر می‌توان از یک تابع وزن‌دهی هموار (کرنل) بهره برد، یکی از معروف‌ترین کرنل‌ها، تابع گاوسی است.

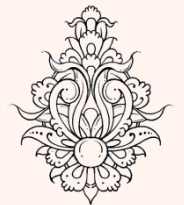
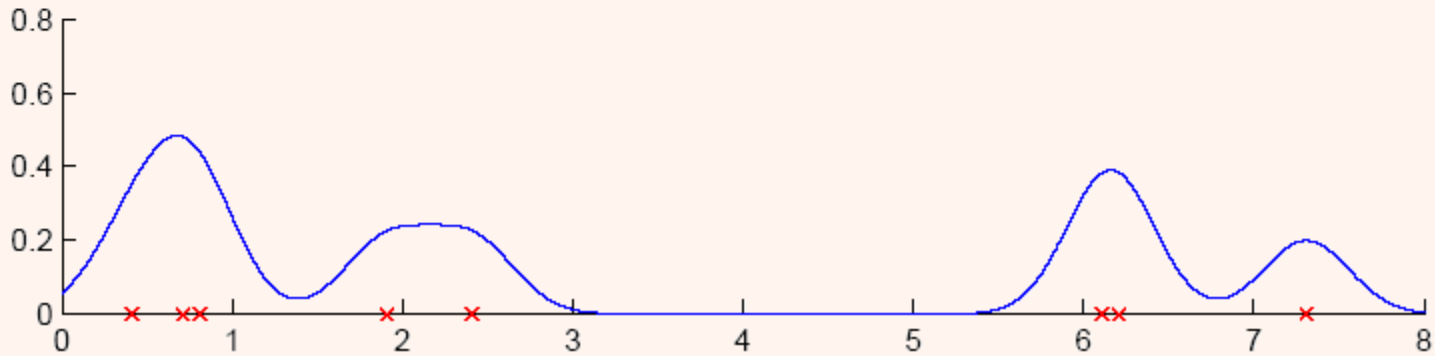
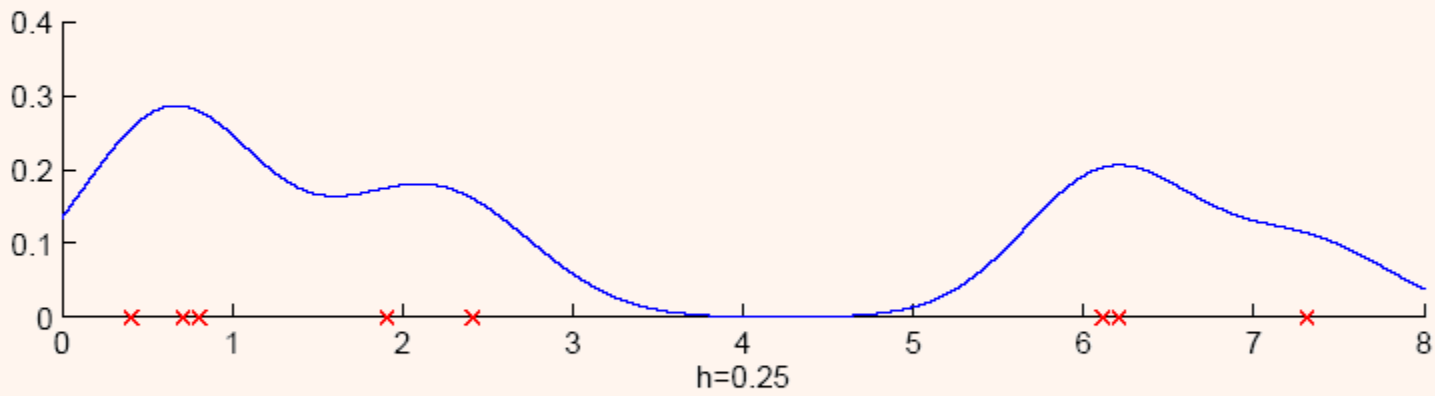
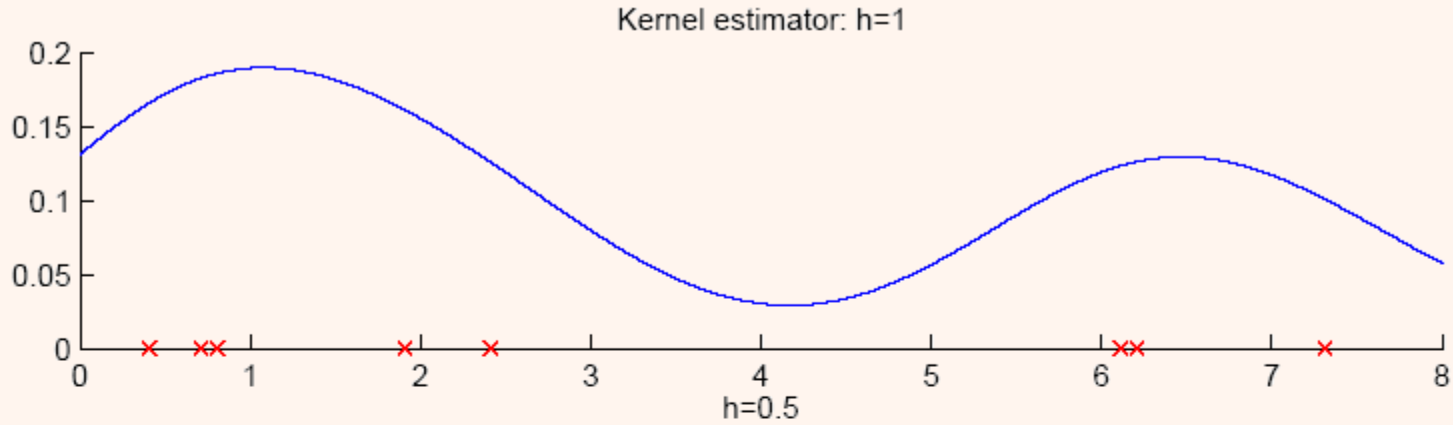
$$K(u) = \frac{1}{\sqrt{2\pi}} \exp\left[-\frac{u^2}{2}\right]$$

$$\hat{p}(x) = \frac{1}{Nh} \sum_{t=1}^N K\left(\frac{x - x^t}{h}\right)$$

- هر تابع نامنفی و متقارن حول صفر که دارای سطح یک باشد، را می‌توان به عنوان کرنل استفاده کرد.



مثال



k-Nearest Neighbor Estimator

- در این روش بازه به صورت افقی در نظر گرفته می‌شود. بازه‌ای که k همسایه‌ی نزدیک در آن واقع هستند. در واقع به جای ثابت در نظر گرفتن نوار، تعداد نمونه‌های واقع در نوار ثابت در نظر گرفته می‌شود.

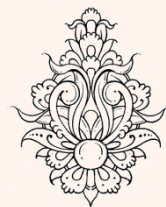
$$\hat{p}(x) = \frac{k}{2Nd_k(x)}$$

$$\hat{p}(x) = \frac{\#\{x-h < x^t \leq x+h\}}{2Nh}$$

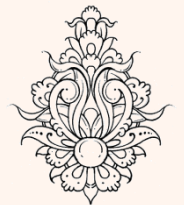
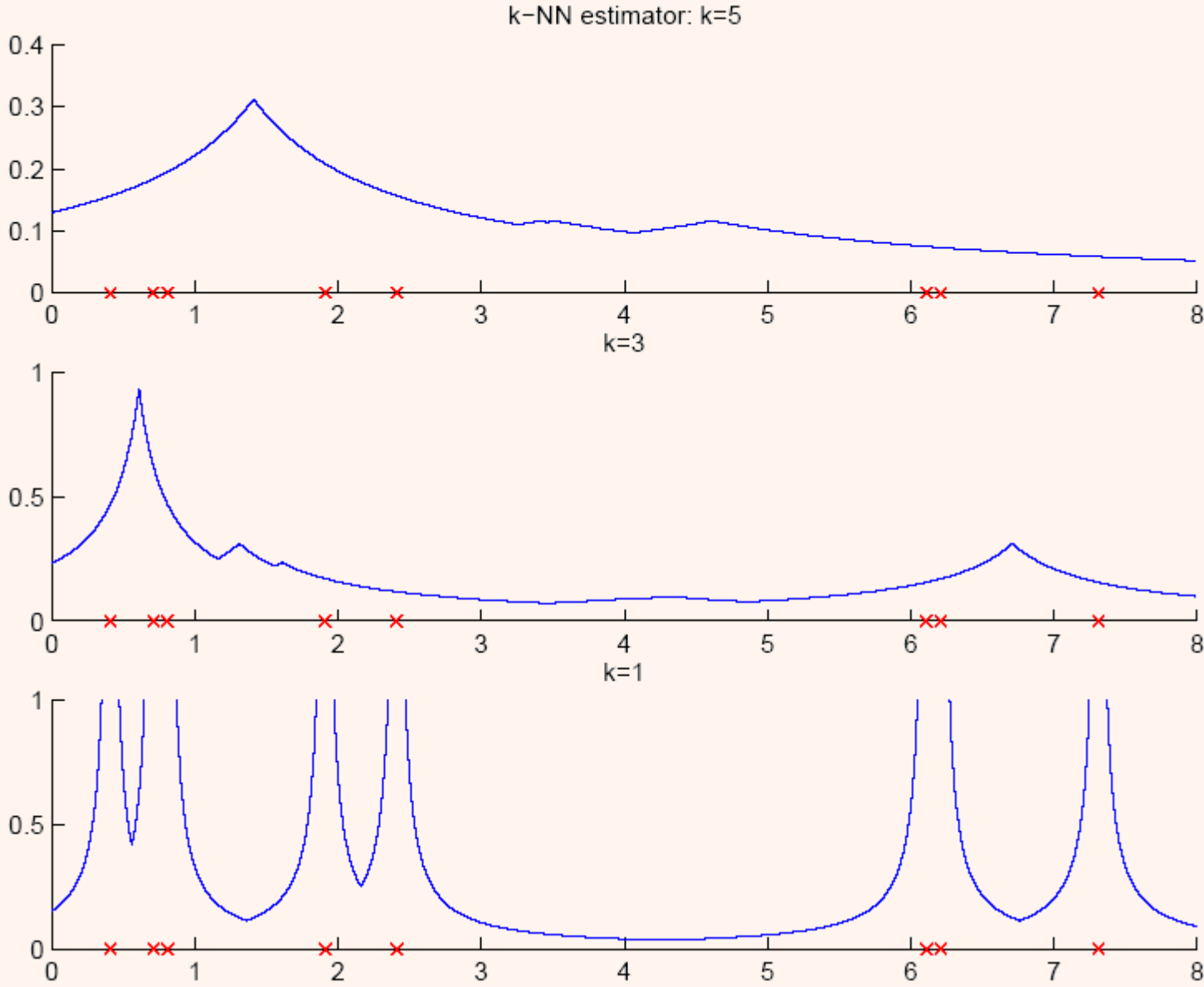
- $d_k(x)$ فاصله‌ی k -امین همسایه‌ی نزدیک است.
- در چگالی تخمین زده شده، شکستگی (گسستگی) در مشتق وجود دارد، برای به دست آوردن تقریب هموارتر می‌توان از کرنل استفاده کرد:

$$\hat{p}(x) = \frac{1}{Nd_k(x)} \sum_{t=1}^N K\left(\frac{x-x^t}{d_k(x)}\right)$$

$$\hat{p}(x) = \frac{1}{Nh} \sum_{t=1}^N K\left(\frac{x-x^t}{h}\right)$$



مثال



تخمین به داده‌های چندبعدی

- تخمین چگالی به صورت زیر انجام می‌شود:

$$\hat{p}(\mathbf{x}) = \frac{1}{Nh^d} \sum_{t=1}^N K\left(\frac{\mathbf{x} - \mathbf{x}^t}{h}\right) \quad \int_{R^d} K(x) dx = 1$$

مشروط به

- در این حالت تابع کرنل به صورت‌های زیر خواهد بود:

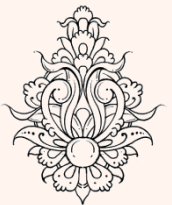
$$K(\mathbf{u}) = \left(\frac{1}{\sqrt{2\pi}}\right)^d \exp\left[-\frac{\|\mathbf{u}\|^2}{2}\right]$$

spheric

$$K(\mathbf{u}) = \frac{1}{(2\pi)^{d/2} |\mathbf{S}|^{1/2}} \exp\left[-\frac{1}{2} \mathbf{u}^T \mathbf{S}^{-1} \mathbf{u}\right]$$

ellipsoid

استفاده از فاصله‌ی Mahalanobis

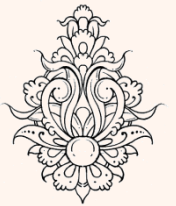


تخمین به داده‌های چندبعدی (ادامه...)

- برای تخمین هیستوگرام یک مجموعه‌ی هشت‌بعدی در صورتی که برای هر بعد تنها ده نوار در نظر گرفته شود:

Curse of dimensionality

– به 10^8 بخش نیاز خواهیم داشت.



دسته‌بندی ناپارامتری

- برای دسته‌بندی ابتدا چگالی احتمال هر کلاس $p(\mathbf{x} | C_i)$ محاسبه می‌شود:

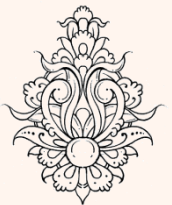
$$\hat{p}(\mathbf{x} | C_i) = \frac{1}{N_i h^d} \sum_{t=1}^N K\left(\frac{\mathbf{x} - \mathbf{x}^t}{h}\right) r_i^t$$

- و جداساز به ترتیب زیر به دست می‌آید:

$$g_i(\mathbf{x}) = \hat{p}(\mathbf{x} | C_i) \hat{P}(C_i) \quad \hat{P}(C_i) = \frac{N_i}{N}$$

$$g_i(\mathbf{x}) = \hat{p}(\mathbf{x} | C_i) \hat{P}(C_i) = \frac{1}{N h^d} \sum_{t=1}^N K\left(\frac{\mathbf{x} - \mathbf{x}^t}{h}\right) r_i^t$$

- \mathbf{x} متعلق به کلاسی است که جداساز آن بیشترین مقدار را داشته باشد، از بخش مشترک جداساز می‌توان چشم‌پوشید.



دسته‌بندی ناپارامتری (ادامه...)

$$g_i(\mathbf{x}) = \sum_{t=1}^N K\left(\frac{\mathbf{x} - \mathbf{x}^t}{h}\right) r_i^t$$

- هر نمونه‌ی آموزشی متعلق به کلاس i -ام به ورودی یک «رأی» می‌دهد. رأی توسط تابع کرنل مشخص می‌شود.

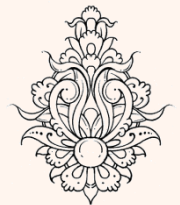
- در حالتی که از k -nn استفاده کنیم:

$$\hat{p}(\mathbf{x}|C_i) = \frac{k_i}{N_i V^k(\mathbf{x})}$$

- k_i تعداد بخشی از k همسایه است که به کلاس i -ام تعلق دارند و V ابرکره‌ای است به مرکز \mathbf{x} و با شعاع نزدیک‌ترین همسایه‌ی k -ام $\|\mathbf{x} - \mathbf{x}_{(k)}\|$

9 -

$$V^k = r^d c_d, \quad c_1 = 2, c_2 = \pi, c_3 = 4\pi/3, \dots$$



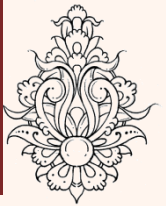
k-nn classifier (دسته‌بندی ناپارامتری (ادامه...)

$$\hat{P}(C_i|\mathbf{x}) = \frac{\hat{p}(\mathbf{x}|C_i)\hat{P}(C_i)}{\hat{p}(\mathbf{x})}$$

$$\hat{p}(\mathbf{x}|C_i) = \frac{k_i}{N_i V^k(\mathbf{x})} \quad \hat{P}(C_i) = \frac{N_i}{N} \quad \hat{p}(\mathbf{x}) = \frac{k}{V^k(\mathbf{x})N}$$

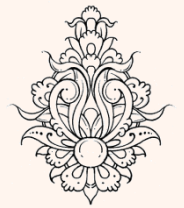
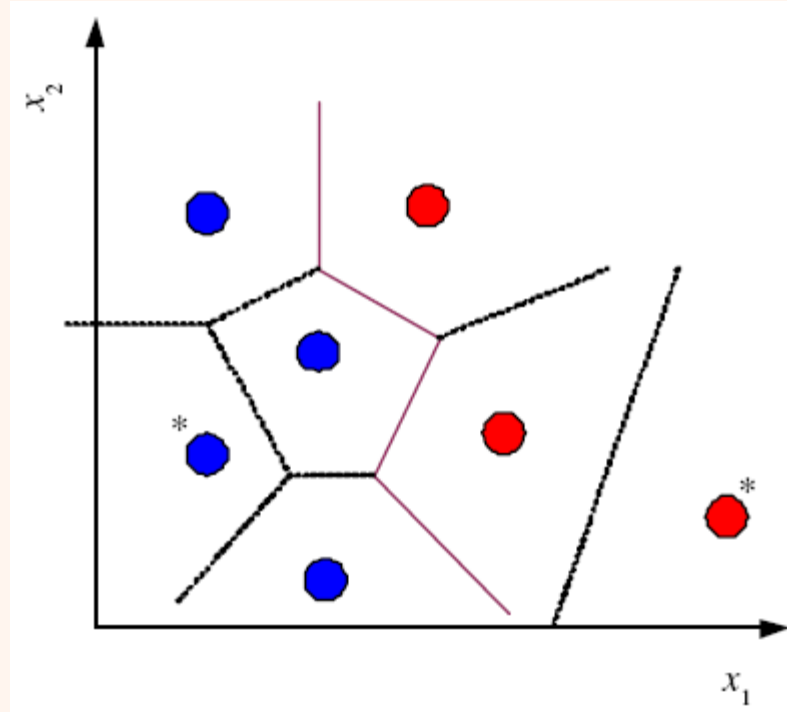
$$\hat{P}(C_i|\mathbf{x}) = \frac{\hat{p}(\mathbf{x}|C_i)\hat{P}(C_i)}{\hat{p}(\mathbf{x})} = \frac{k_i}{k}$$

در دسته‌بندی k-nn ورودی به کلاسی نسبت داده می‌شود که در بین k همسایه‌ی نزدیک بیشترین عضو را داشته باشد.



Nearest neighbor classifier

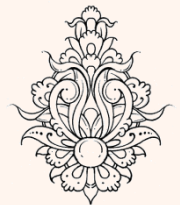
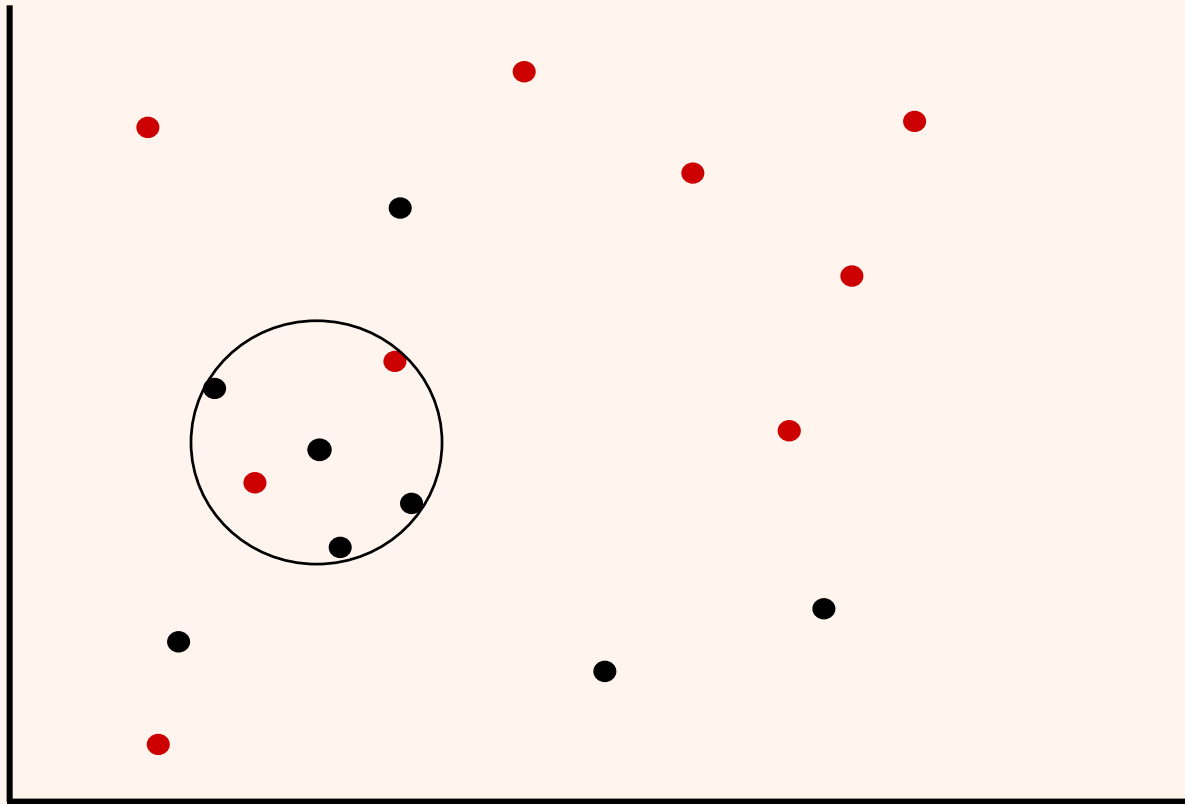
- در حالت خاص که $k=1$ در نظر گرفته شود، ورودی به کلاسی نسبت داده می‌شود که نزدیک‌ترین نمونه به ورودی متعلق به آن است:



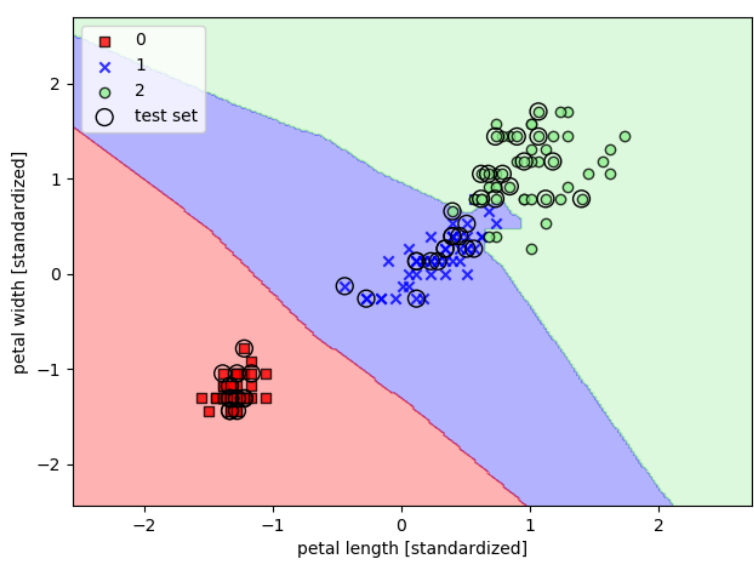
Voronoi tessellation

5-Nearest Neighbor

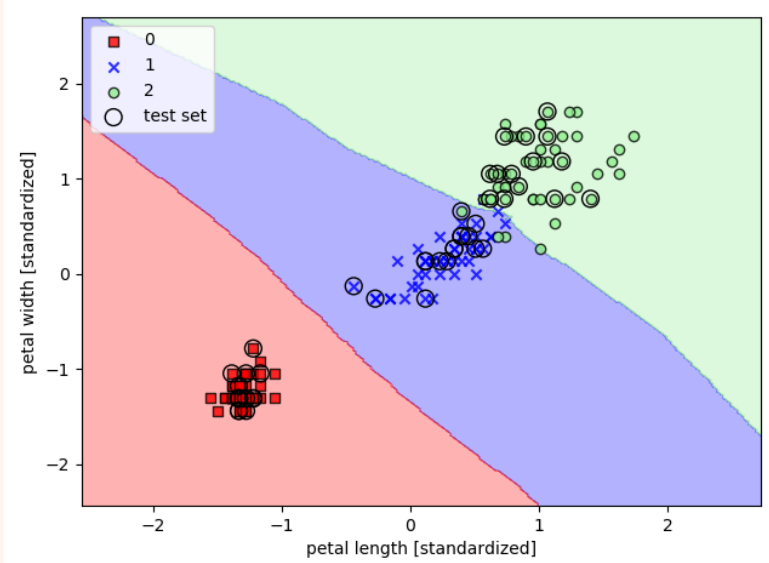
مثال



مثال



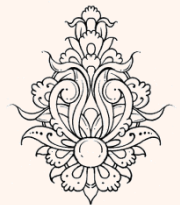
n_neighbors=1



n_neighbors=10

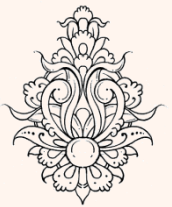
```
from sklearn.neighbors import KNeighborsClassifier

knn = KNeighborsClassifier(n_neighbors=?,
                          p=2,
                          metric='minkowski')
knn.fit(X_train_std, y_train)
```



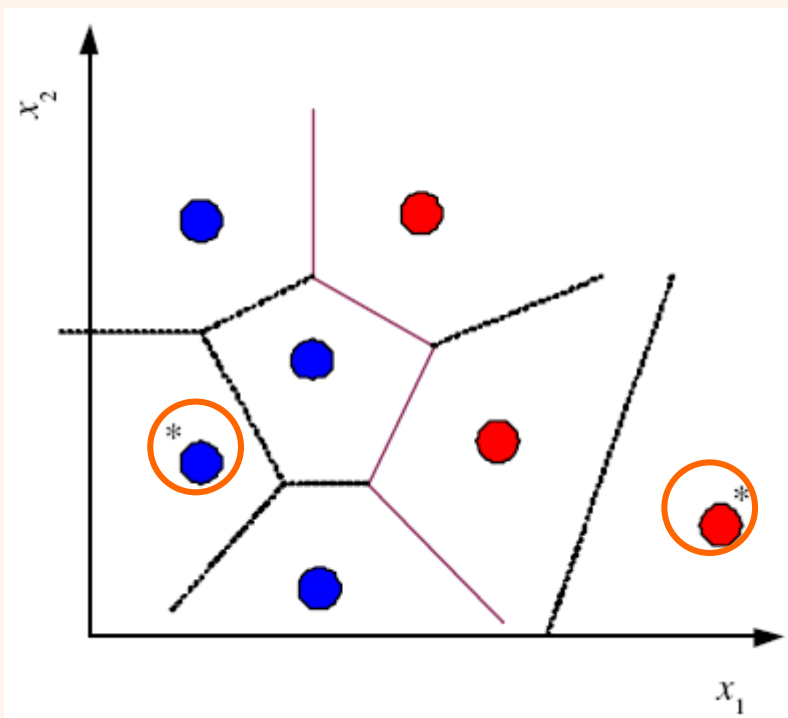
Condensed Nearest Neighbor

- پیچیدگی مکانی و زمانی روش‌های ناپارامتری به تعداد نمونه‌های آموزشی (N) بستگی دارد.
- روش‌هایی ارائه شده است که هدف آن کاهش نمونه‌های ذخیره شده بدون افت کارایی است.
- در این روش‌ها یک زیر مجموعه‌ی کوچک (Z) از نمونه‌های آموزشی (X) به نحوی انتخاب می‌شود که ضمن کاهش داده‌های ذخیره شده خطا افزایش چندانی نیابد.
- در `Condensed Nearest Neighbor` برای دسته‌بندی از $1-nn$ استفاده می‌شود.



Condensed Nearest Neighbor (C.N.N.)

- در 1-nn جداساز خطی به صورت تکه‌ای تقریب زده می‌شود و تنها نمونه‌هایی که در تشکیل جداساز نقش دارند، لازم است، نگهداری شوند. به چنین مجموعه‌ای consistent می‌گویند.



Incremental algorithm

$Z \leftarrow \emptyset$

Repeat

For all $x \in \mathcal{X}$ (in random order)

Find $x' \in Z$ s.t. $\|x - x'\| = \min_{x^j \in Z} \|x - x^j\|$

If $\text{class}(x) \neq \text{class}(x')$ add x to Z

Until Z does not change

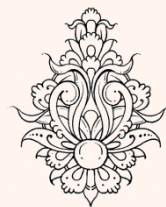
- پاسخ نهایی به ترتیب اعمال ورودی‌ها بستگی دارد.

- با تعریف تابع هزینه‌ی زیر می‌توان بین پیچیدگی و خطای مدل بسته به مقدار λ نوعی مصالحه برقرار کرد:

$$E'(Z | \mathcal{X}) = E(\mathcal{X} | Z) + \lambda |Z|$$



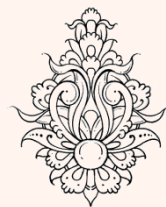
- داده‌ی پرت (ناهنجار)، داده‌ای است که با سایر نمونه‌ها تفاوت زیادی دارد.
- چنین نمونه‌ای بیانگر یک رفتار غیرعادی است:
 - تراکنش‌های کارت اعتباری، سوءاستفاده از کارت
 - در تصاویر پزشکی، وجود تومور
 - در ترافیک شبکه، نفوذ در شبکه
 - در پرونده‌ی پزشکی، بروز بیماری
 - بروز خطا در یک سیستم، فراب شدن یک سنسور
- معمولاً به عنوان یک مسأله‌ی دو کلاسه با نظارت تلقی نمی‌شوند.
 - داده‌های پرت، نسبت به داده‌های نرمال بسیار اندک هستند و معمولاً بدون برچسب هستند.



تشخیص داده‌های پرت (ادامه...)

one-class classification problem

- در واقع، مسأله یک دسته‌بندی کننده‌ی تک‌کلاسه است.
- در این حالت داده‌های نرمال مدل می‌شوند؛ توزیع داده‌های نرمال به دست می‌آید.
- در این حالت هرچند می‌توان از شیوه‌های پارامتری و نیمه‌پارامتری استفاده کرد، اما با توجه به حساسیت روش‌های پارامتری به داده‌های پرت، روش‌های ناپارامتری ترجیح داده می‌شود.
- یک داده‌ی پرت مدل را تحت تأثیر قرار می‌دهد.

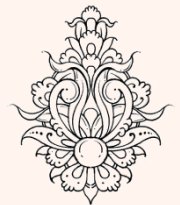
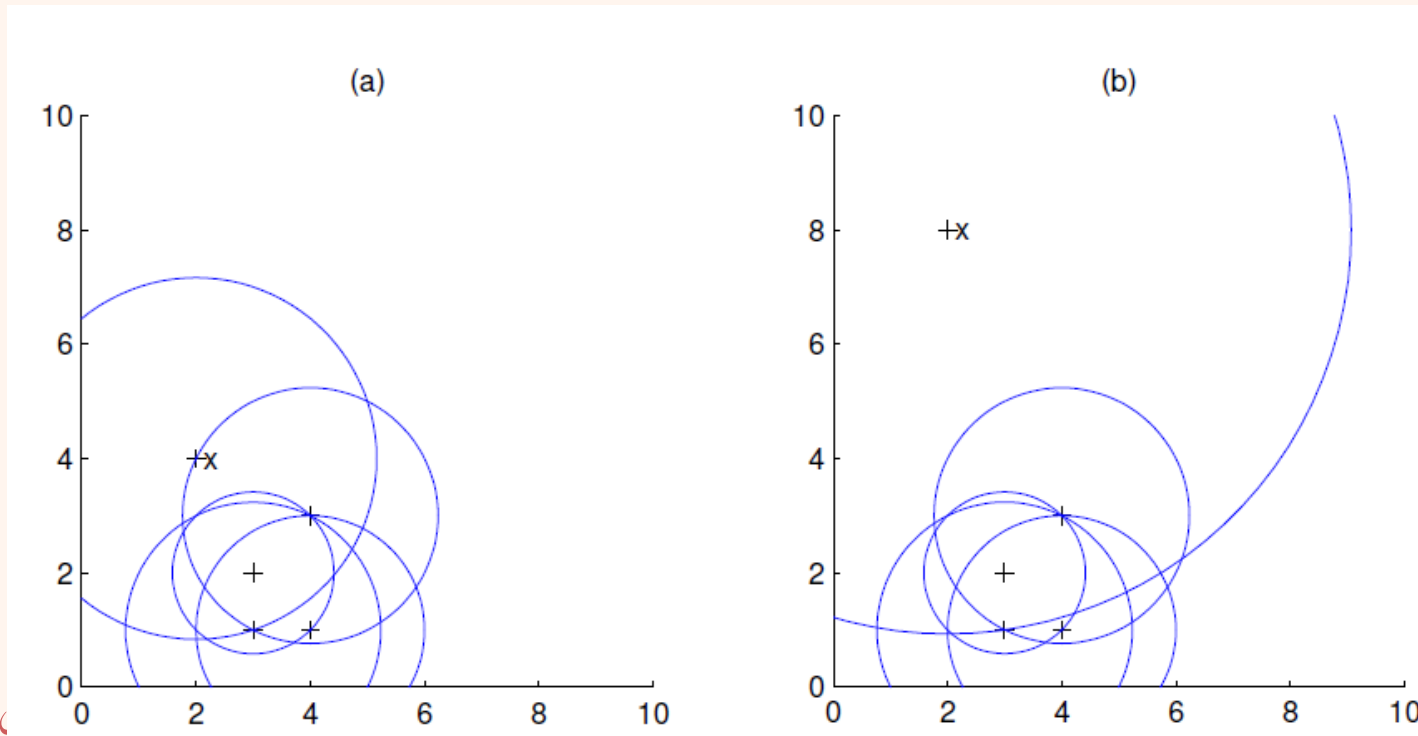


تشخیص داده‌های پرت (ادامه...)

Local Outlier Factor

- در روش‌های ناپارامتری در صورتی که یک نمونه از سایر نمونه‌ها دور باشد، پرت تشخیص داده می‌شود.

$$\text{LOF}(x) = \frac{d_k(x)}{\sum_{s \in \mathcal{N}(x)} d_k(s) / |\mathcal{N}(x)|}$$



رگرسیون ناپارامتری (تک متغیره)

Nonparametric Regression

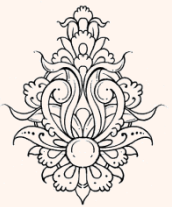
smoothing models(smoother)

$$\mathcal{X} = \{\mathbf{x}^t, r^t\}, \quad r^t = g(\mathbf{x}^t) + \varepsilon$$

- رگرسیون ناپارامتری زمانی مورد استفاده قرار می‌گیرد که نتوان یک مدل کلی برای داده‌ها در نظر گرفت.

- در این حالت فرض می‌شود که داده‌های نزدیک به x مقادیر نزدیک به $g(x)$ خواهند داشت.

- در این حالت رویکرد یافتن همسایه‌های x و میانگین گرفتن از مقادیر r مربوط به آن‌هاست.

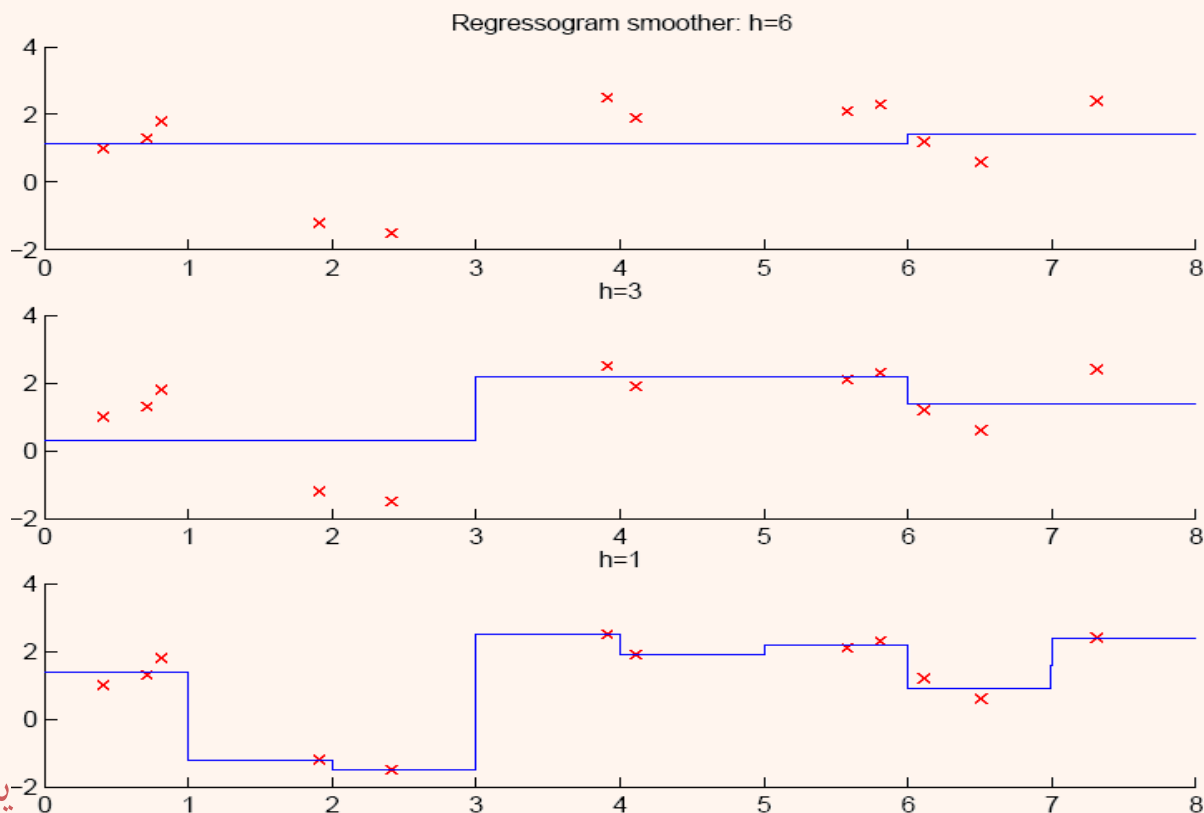


Regressogram

این واژه در سال ۱۹۶۱ توسط شخصی به نام Tukey مطرح شد تا شباهت آن را به هیستوگرام نشان دهد.

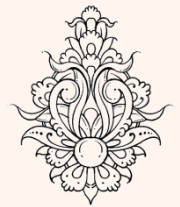
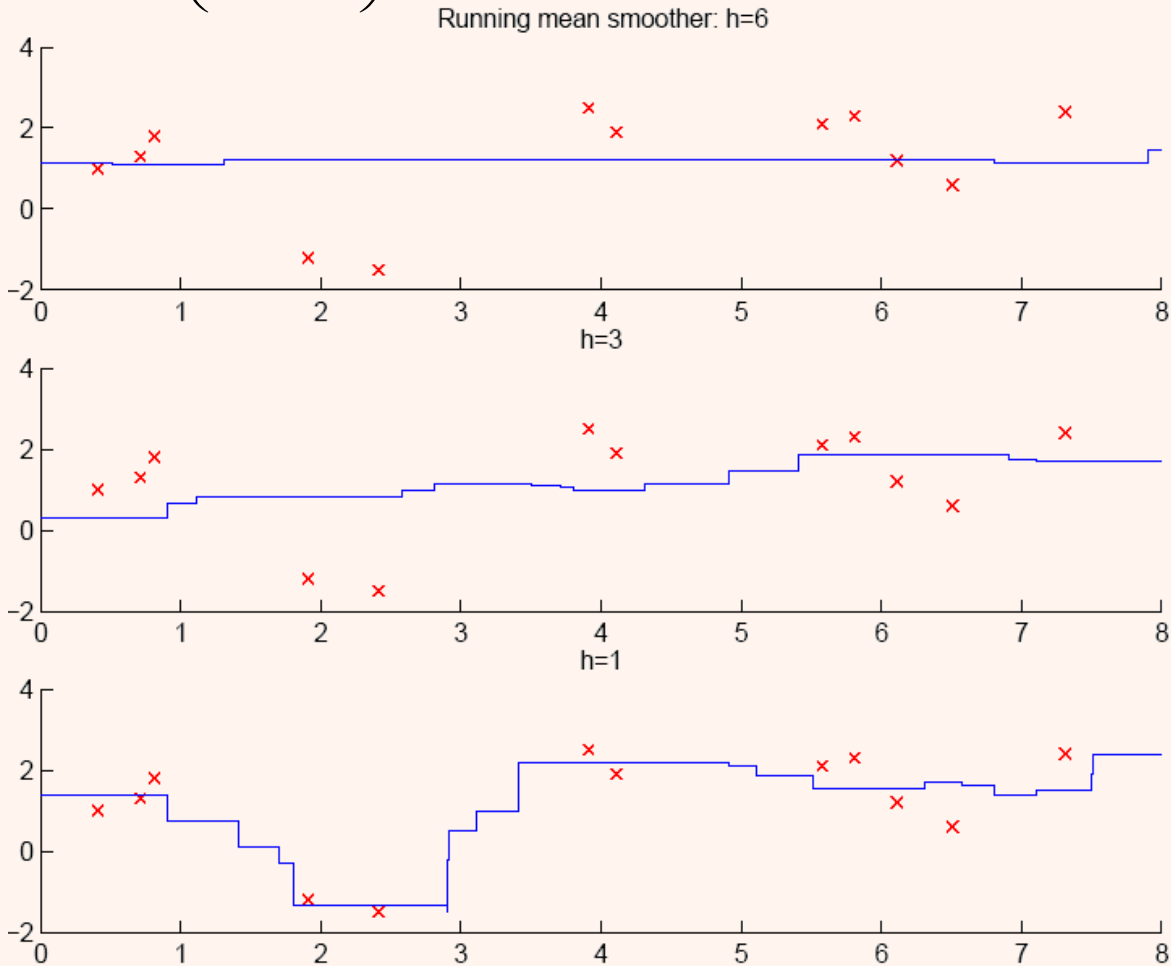
$$\hat{g}(x) = \frac{\sum_{t=1}^N b(x, x^t) r^t}{\sum_{t=1}^N b(x, x^t)}$$

where $b(x, x^t) = \begin{cases} 1 & \text{if } x^t \text{ is in the same bin with } x \\ 0 & \text{otherwise} \end{cases}$



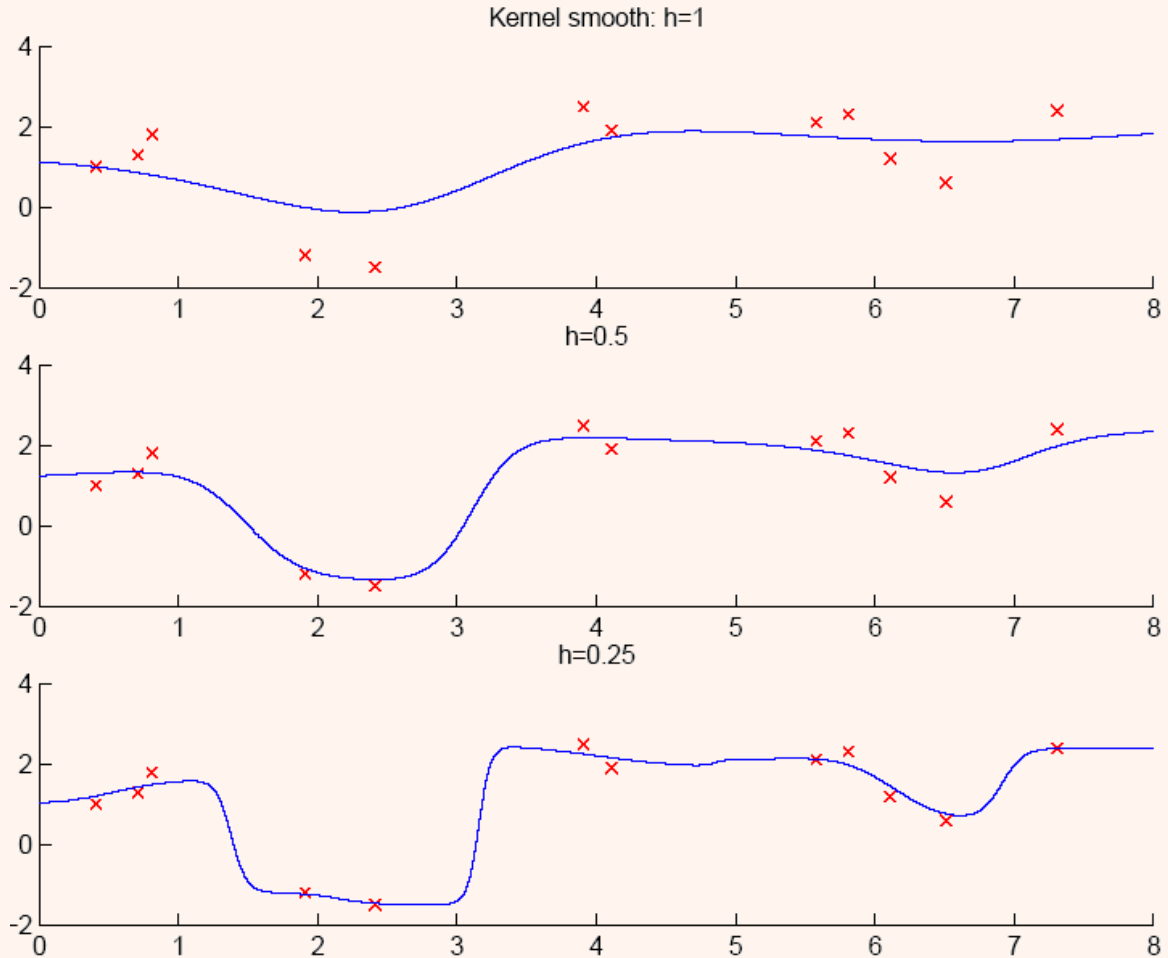
Running Mean Smoother

$$\hat{g}(x) = \frac{\sum_{t=1}^N w\left(\frac{x-x^t}{h}\right) r^t}{\sum_{t=1}^N w\left(\frac{x-x^t}{h}\right)} \quad \text{where } w(u) = \begin{cases} 1 & \text{if } |u| < 1 \\ 0 & \text{otherwise} \end{cases}$$

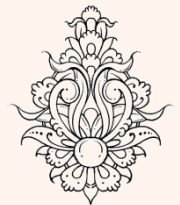


Kernel Smoother

$$\hat{g}(x) = \frac{\sum_{t=1}^N K\left(\frac{x - x^t}{h}\right) r^t}{\sum_{t=1}^N K\left(\frac{x - x^t}{h}\right)}$$

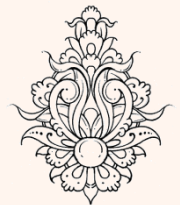
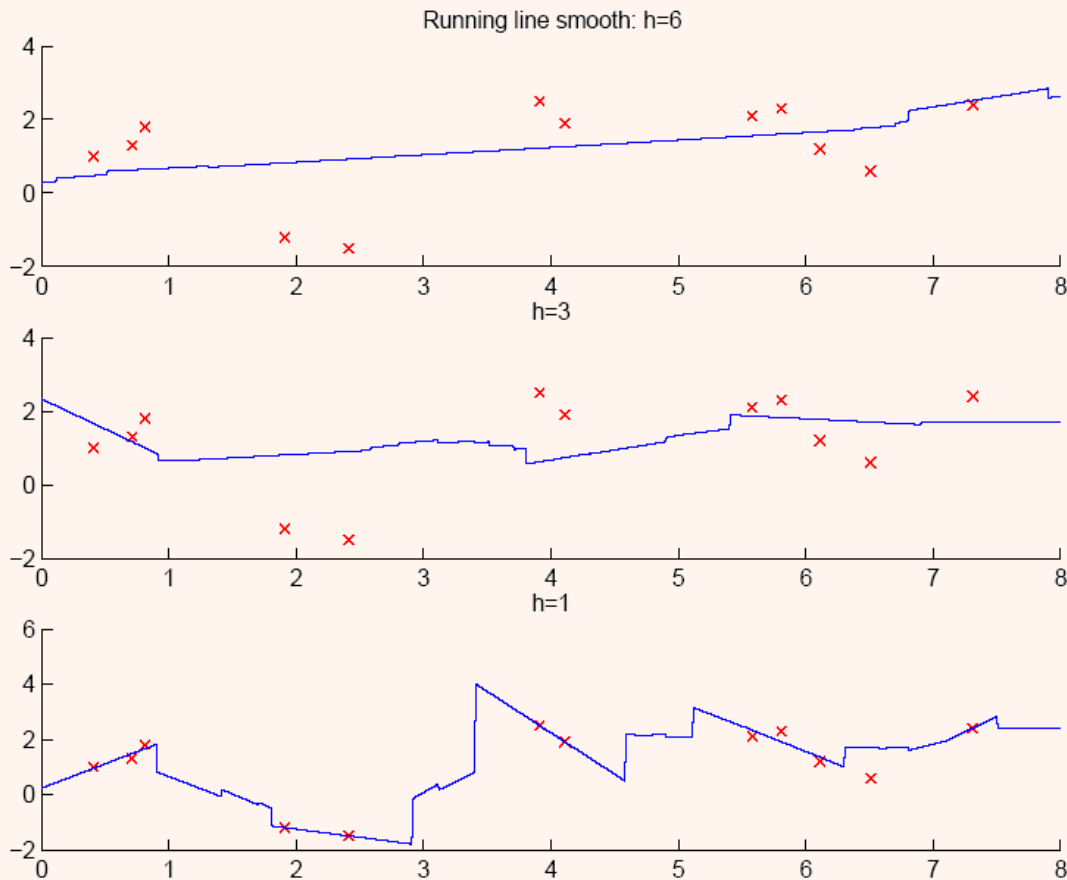


K() is Gaussian



Running line smoother

- در این حالت برای هر همسایگی، یک رگرسیون خطی به صورت محلی در نظر گرفته می‌شود.



تعیین پارامتر هموارسازی

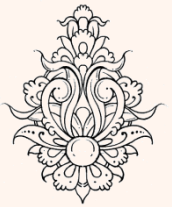
- در صورتی که k یا h کوچک در نظر گرفته شوند (به عنوان مثال زمانی که تنها خود نمونه در نظر گرفته شود)، میزان بایاس کم است، اما واریانس بالا خواهد بود (پیچیدگی زیاد).

undersmoothing

- در صورت افزایش دامنه‌ی هموارسازی، واریانس کاهش یافته، اما بایاس افزایش می‌یابد (پیچیدگی کم).

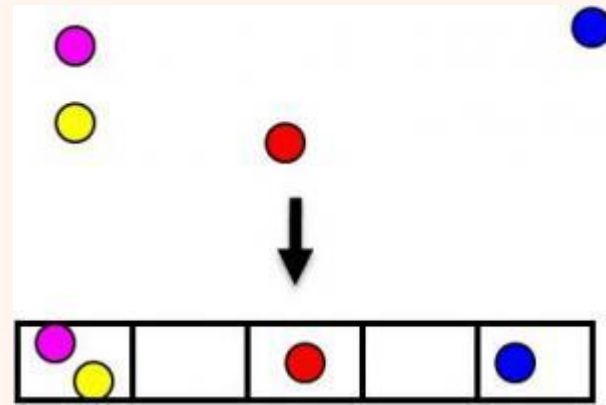
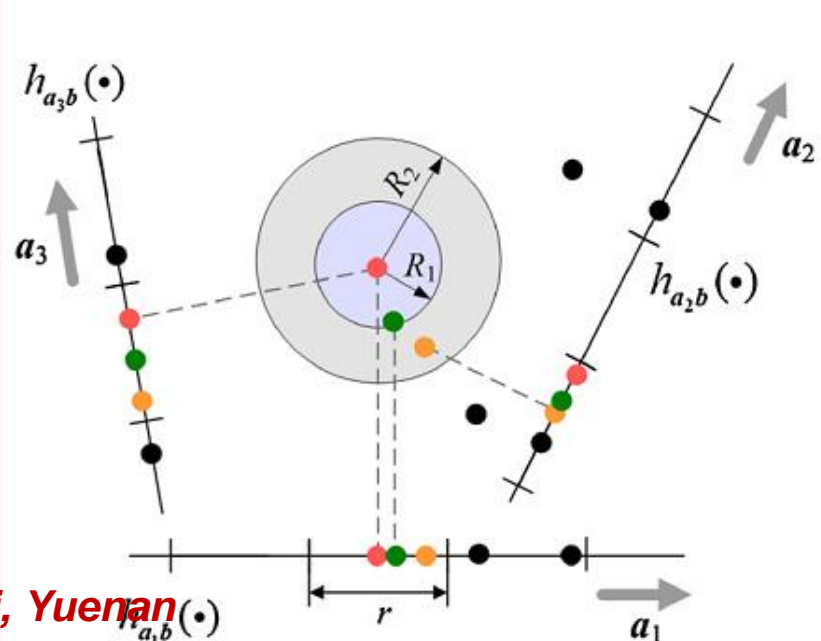
oversmoothing

- می‌توان از cross validation نیز برای تنظیم پارامتر هموارسازی استفاده کرد.

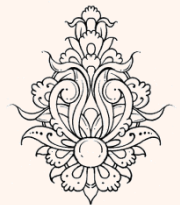


روش‌های جستجو

- یافتن «نزدیک‌ترین همسایه» به صورت جستجوی خطی به ویژه زمانی که تعداد نمونه‌های آموزشی بالاست، به صرفه نیست.
- در این شرایط عموماً از درخت kd استفاده می‌شود.
- همچنین روش‌های درهم‌سازی نظیر (LSH) استفاده می‌شود.



<http://bigdata.csail.mit.edu/node/17>



Li, Yuenan

یادگیری ماشین

Locality Sensitive Hashing