مدرسه پردازش و تحلیل داده دقیقه

K-نزدیک‌ترین همسایه

K Nearest Neighbors

سعید مجیدی

یادگیری ماشین با پایتون

بهار ۱۴۰۴

FRIENDS

You Are Genetically Similar to Your Friend

Friends are as genetically related as fourth cousins.

Posted November 3, 2020 | Reviewed by Ekua Hagan

Social Proximity Effect: Your Friends' Habits Will Become Your Habits

by tyler tervooren | 4 minute read

The gist: You'll mirror the habits of the people you spend the m
people who already practice them.

الگوریتم K-نزدیک‌ترین همسایه
K-Nearest Neighbors Algorithm (KNN)

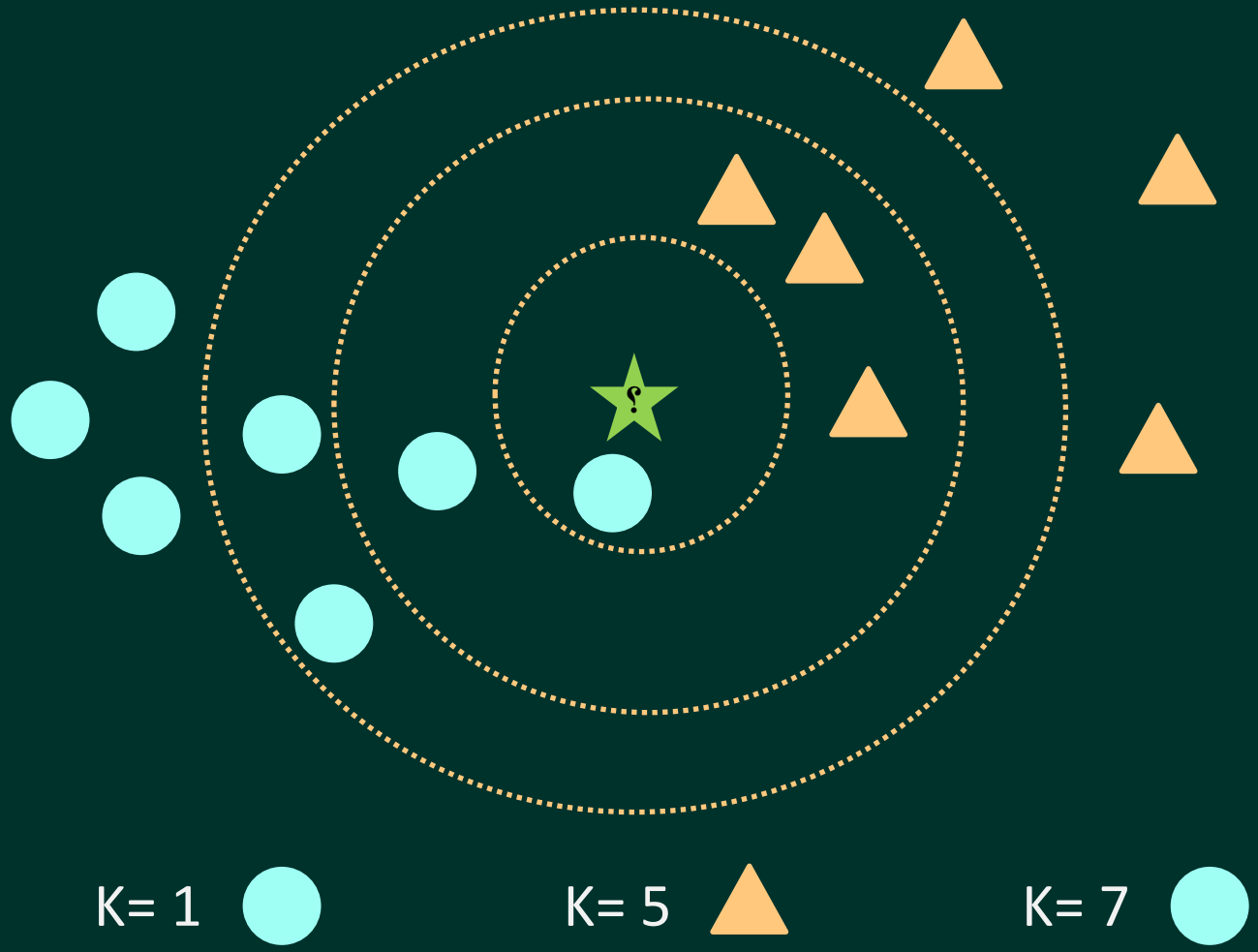
- معروف‌ترین الگوریتم نمونه‌محور، یک الگوریتم **بانظارت**

- **یادگیری**: ندارد
- **پارامتر**: K
- **پیش‌بینی** برای یک نمونه‌ی جدید:
  - K-نزدیک‌ترین همسایه‌ی نمونه‌ی جدید را در بین نمونه‌های آموزشی پیدا کن
  - مسئله دسته‌بندی: برچسبی که دربین همسایه‌ها اکثریت را دارد به نمونه جدید نسبت بده
  - مسئله رگرسیون: میانگین برچسب‌های همسایه‌ها را به نمونه جدید نسبت بده.

- نزدیک‌ترین معمولا براساس فاصله اقلیدسی محاسبه می‌شود

$$d(\mathbf{p}, \mathbf{q}) = \sqrt{\sum_{i=1}^{n} (q_i - p_i)^2}$$

مثال

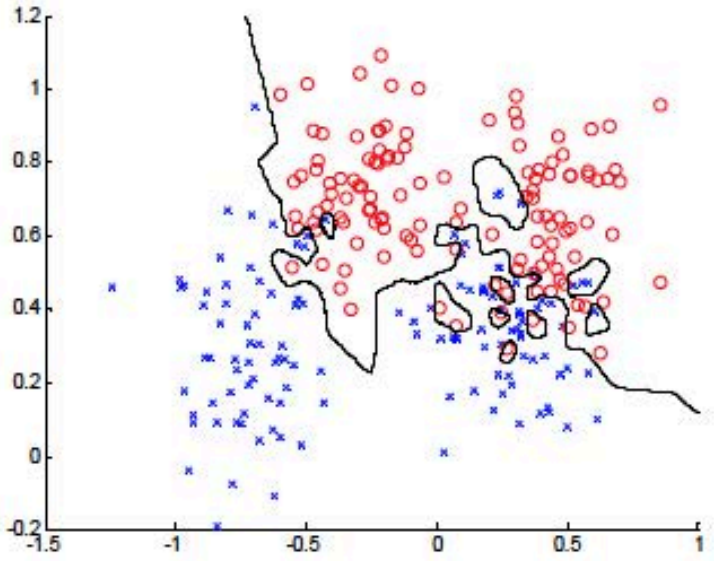


تعمیم (GENERALIZATION)

- هدف اصلی هر الگوریتم بانظارت، پیش‌بینی صحیح برای نمونه‌هایی است که تاکنون دیده نشده‌اند.
- انتخاب پارامترهایی که خطای مدل را تنها بر روی داده آموزشی کم می‌کنند، عاقلانه نیست
  - کاهش خطا = افزایش دقت (Error = 1 – accuracy)
- هدف آن است که ماشین قواعد صحیح حاکم بر داده را یاد بگیرد و به وجود نویز در داده‌ها توجه نکند.

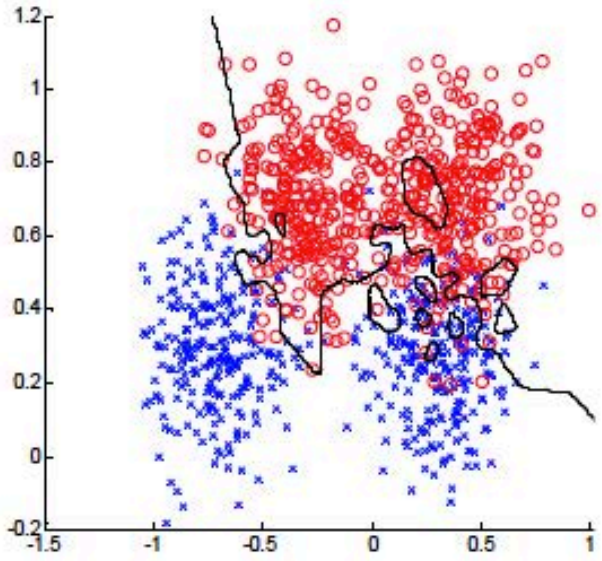
K = 1

Training data



error = 0.0

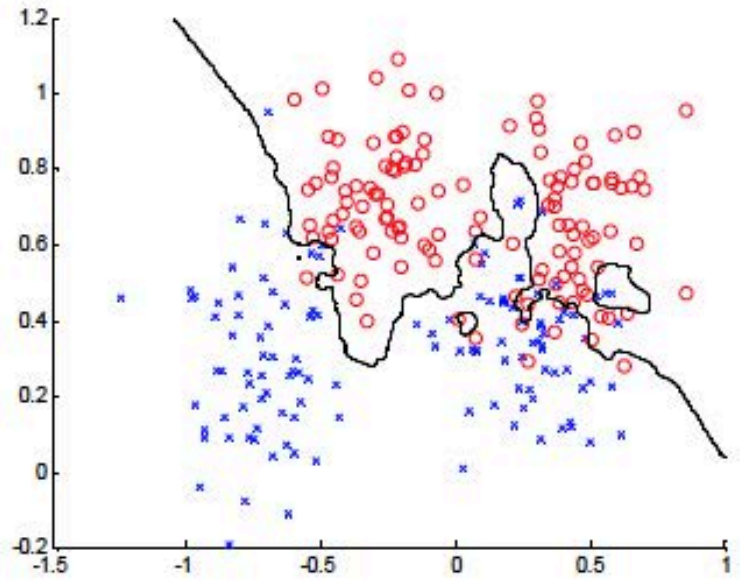
Testing data



error = 0.15

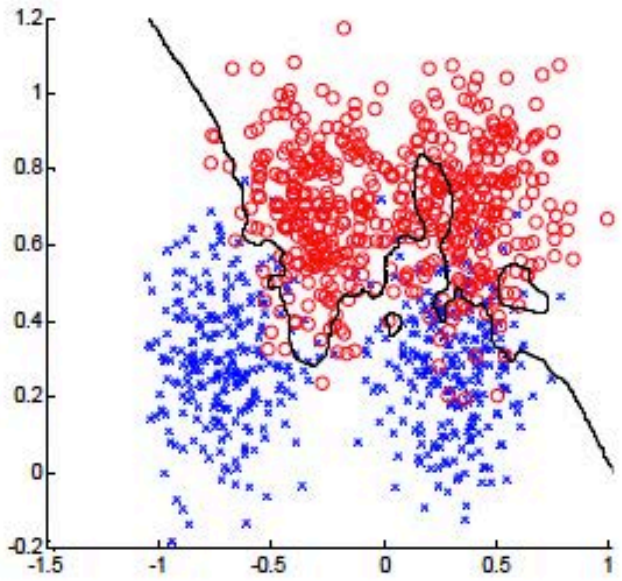
K = 3

Training data



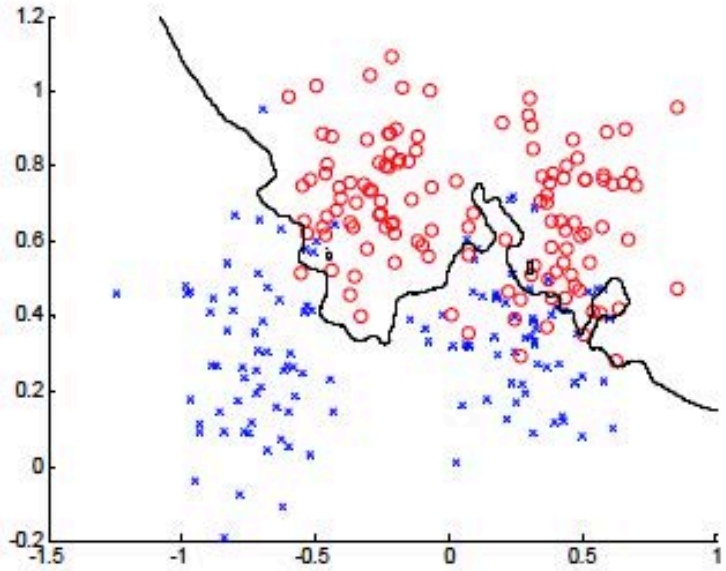
error = 0.0760

Testing data

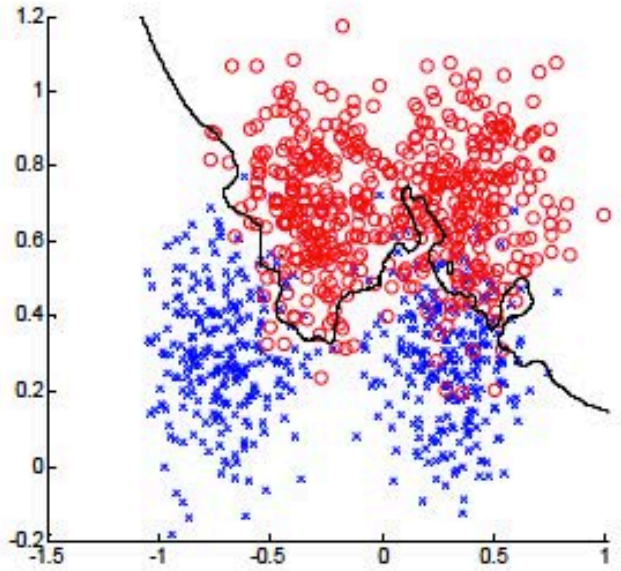


error = 0.1340

K = 7

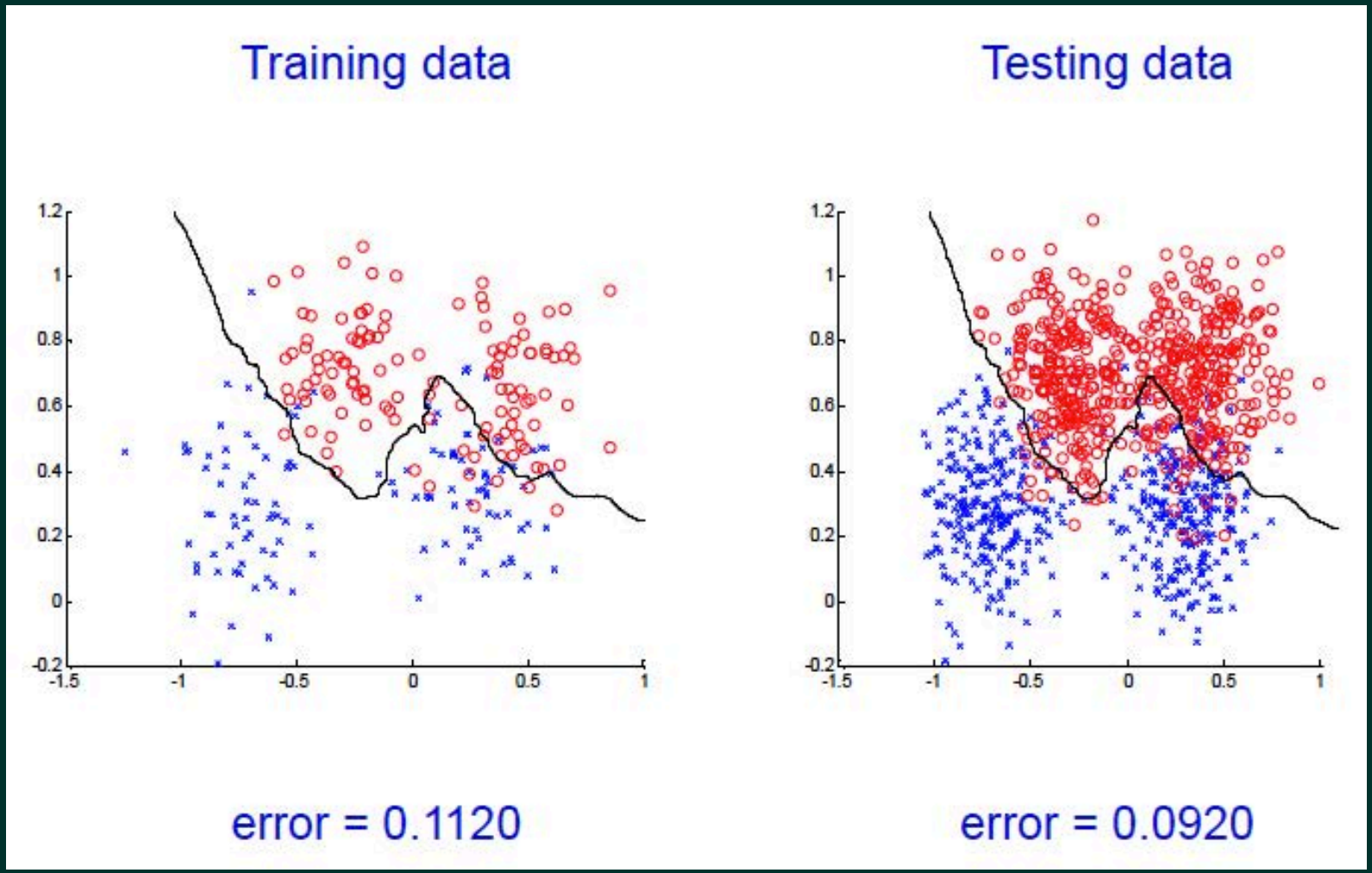


error = 0.1320



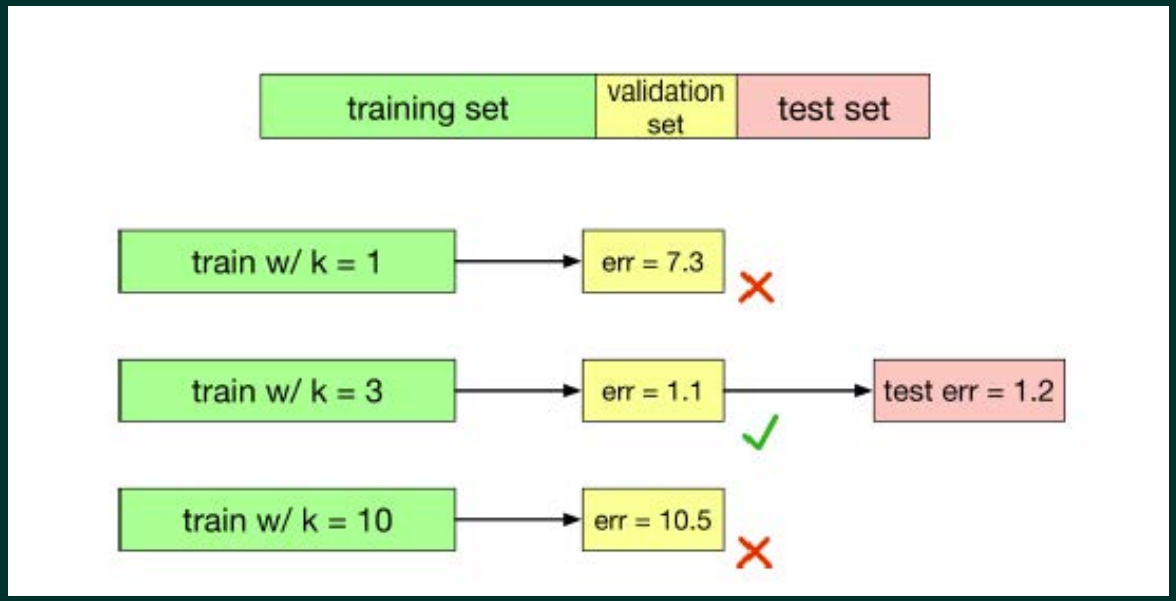
error = 0.1110

K = 21



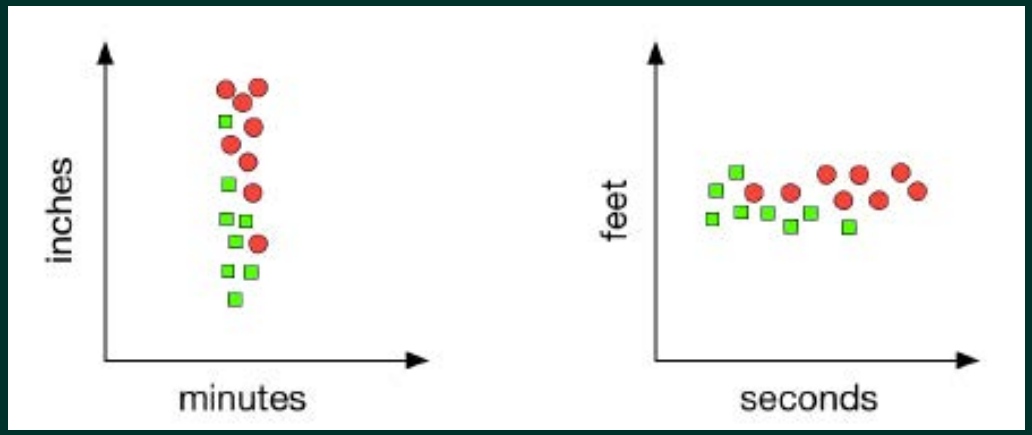
انتخاب بهترین پارامتر

- با افزایش K خطای آموزش ممکن است افزایش یابد.
- انتخاب بهترین پارامتر از طریق Cross Validation
  - به ازای هر مقدار پارامتر K
  - هر بار داده آموزشی را به دو دسته مجموعه آموزشی و مجموعه اعتبارسنجی تقسیم می‌کنیم.
  - خطای آموزش را هر بار بر روی مجموعه اعتبارسنجی محاسبه می‌کنیم.
  - مقدار پارامتر K که کمترین خطا روی مجموعه اعتبارسنجی داشت را انتخاب می‌کنیم.



نرمال‌سازی/ استانداردسازی ویژگی‌های

- **مشکل**: فاصله‌ی اقلیدسی به مقیاس و محدوده‌ی مقادیر ویژگی‌ها حساس است.
- ویژگی‌هایی که مقیاس و محدوده‌ی بزرگ‌تری دارند، اثر ویژگی‌های با محدوده کمتر را کم‌رنگ می‌کنند.



$$euclidean(A,B) = \sqrt{\underbrace{(125.000 - 100.000)^2}_{25000^2} + \underbrace{(52-45)^2}_{7^2}}$$

Contribution of bank deposit

Contribution of age

- مثال: داده بانکی مشتریان
  - موجودی حساب (از ۰ تا ۱۰۰۰ میلیارد)
  - سن (از ۱ تا ۲۰۰)

نرمال‌سازی / استانداردسازی

- راه حل: تغییر محدوده‌ی مقادیر ویژگی‌ها به‌گونه‌ای که همه محدوده‌ی یکسانی داشته باشند.
- نرمال‌سازی داده (Normalization)
  - تغییر دامنه مقادیر به بازه ۰ تا ۱ با استفاده از مقادیر ماکزیمم و مینمم هر ویژگی

$$Xnormalised = \frac{X - Xmin}{Xmax - Xmin}$$

- استانداردسازی داده (Standardization)
  - تغییر دامنه مقادیر هرویژگی به‌نحوی که مقادیر توزیع نرمال با میانگین ۰ و واریانس ۱ داشته باشند.

$$Xstandarised = \frac{X - \mu}{\sigma}$$

پیاده‌سازی

```
def knn(trainingSet, testInstance, k):
    distances = {}
    sort = {}

    length = testInstance.shape[1]

    for x in range(len(trainingSet)):
        dist = EuclideanDistance(testInstance, trainingSet.iloc[x], length)
        distances[x] = dist[0]
    sortdist = sorted(distances.items(), key=operator.itemgetter(1))
    neighbors = []
    for x in range(k):
        neighbors.append(sortdist[x][0])
    Votes = {} #to get most frequent class of rows
    for x in range(len(neighbors)):
        response = trainingSet.iloc[neighbors[x]][-1]
        if response in Votes:
            Votes[response] += 1
        else:
            Votes[response] = 1
    sortvotes = sorted(Votes.items(), key=operator.itemgetter(1), reverse=True)
    return(sortvotes[0][0], neighbors)
```


JUST REMINDING YOU:
BE A GOOD
NEIGHBORINO!