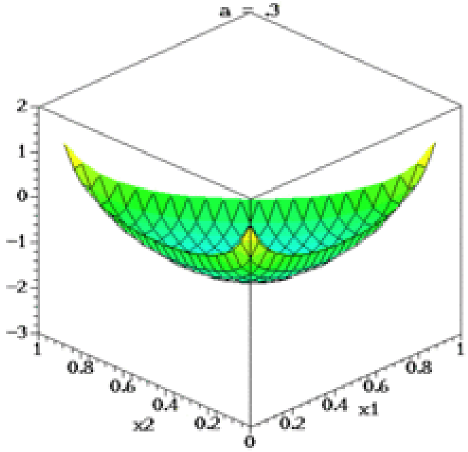
Machine Learning

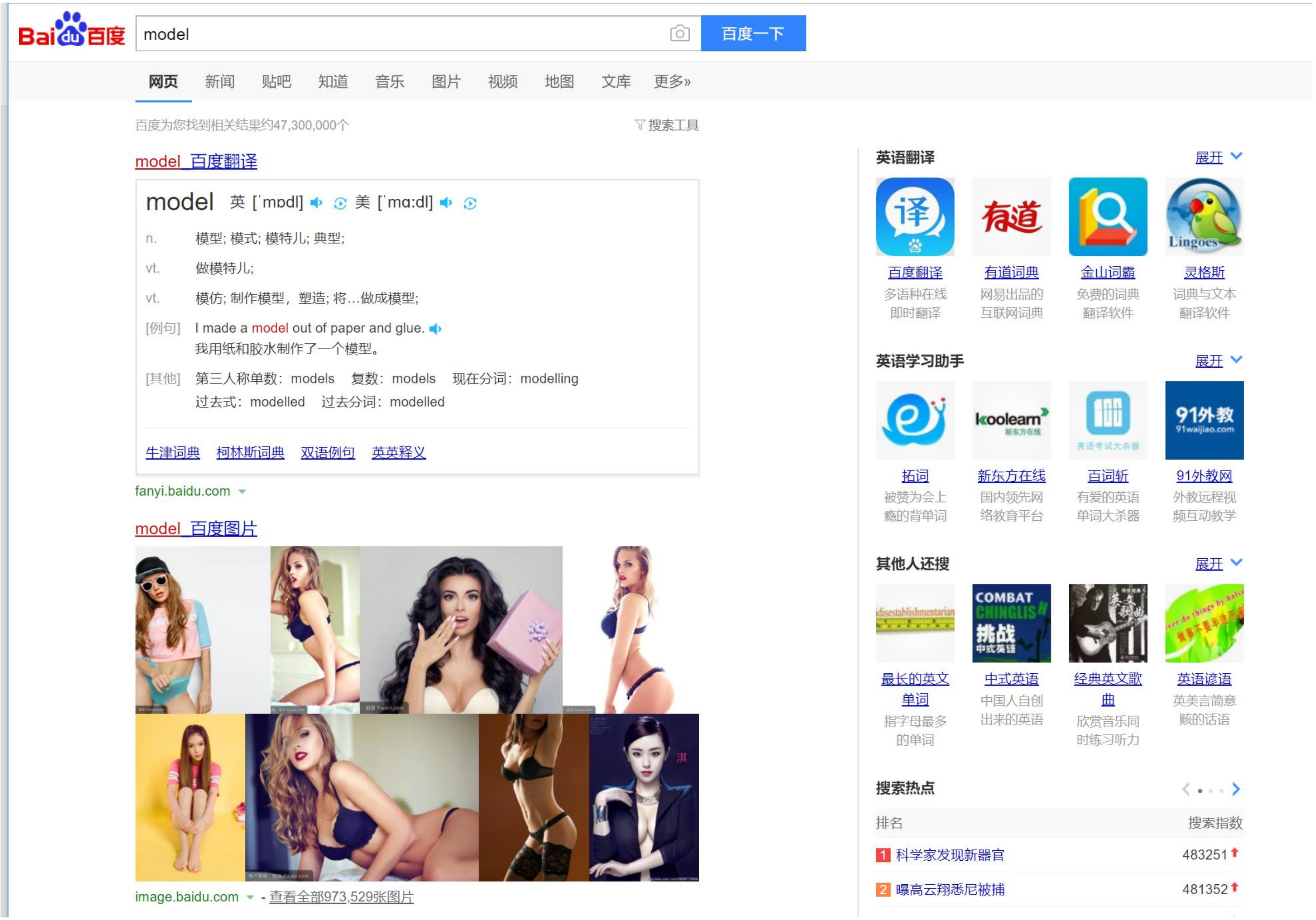
Part 4: Classical Machine Learning Model

Zengchang Qin (Ph.D.)

Case Based Reasoning

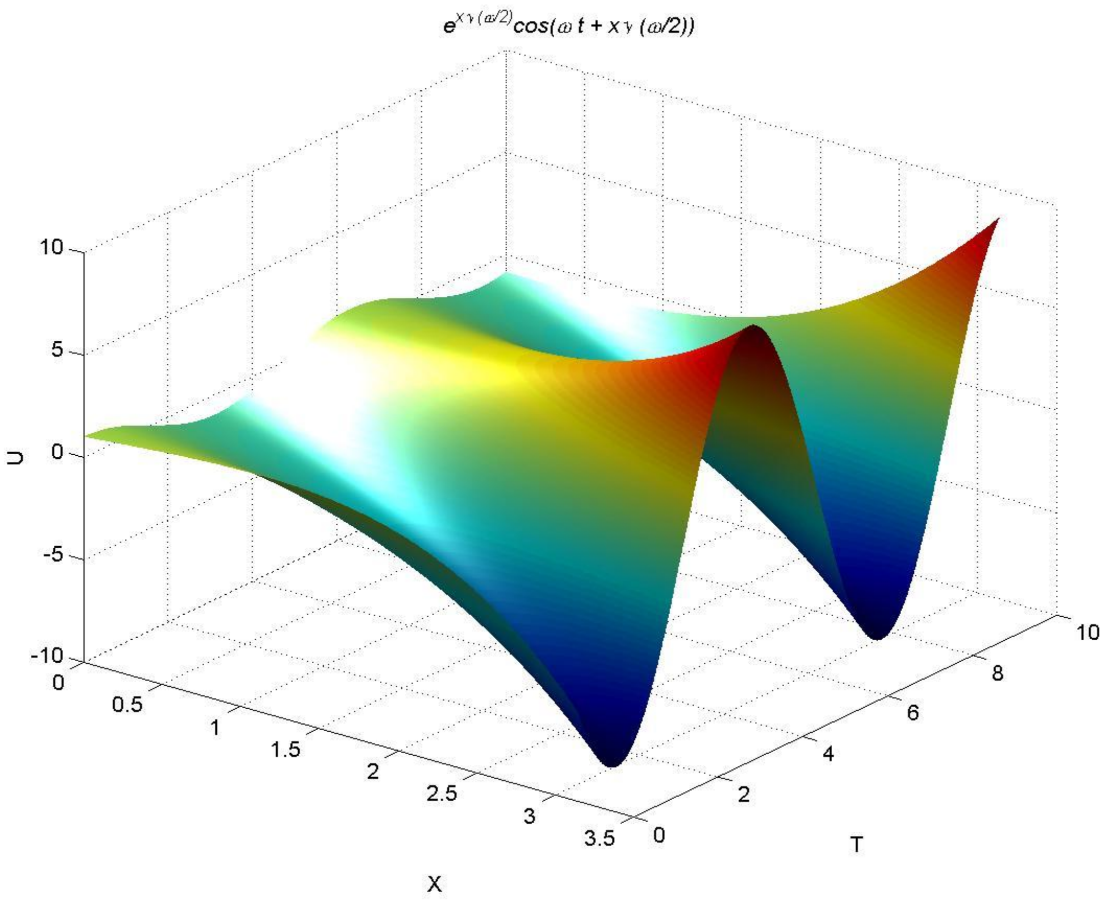


Model



百度一下
model
网页
新闻
贴吧
知道
音乐
图片
视频
地图
文库
更多»
百度为您找到相关结果约47,300,000个
搜索工具
model_百度翻译
model 英 [ˈmɒdl] 美 [ˈmɑːdl]
n. 模型; 模式; 模特儿; 典型;
vt. 做模特儿;
vt. 模仿; 制作模型，塑造; 将…做成模型;
[例句] I made a model out of paper and glue.
我用纸和胶水制作了一个模型。
[其他] 第三人称单数：models 复数：models 现在分词：modelling
过去式：modelled 过去分词：modelled
牛津词典
柯林斯词典
双语例句
英英释义
fanyi.baidu.com
model_百度图片
image.baidu.com - 查看全部973,529张图片
英语翻译
展开
百度翻译
多语种在线即时翻译
有道词典
网易出品的互联网词典
金山词霸
免费的词典翻译软件
灵格斯
词典与文本翻译软件
英语学习助手
展开
拓词
被赞为会上瘾的背单词
新东方在线
国内领先网络教育平台
百词斩
有爱的英语单词大杀器
91外教网
外教远程视频互动教学
其他人还搜
展开
最长的英文单词
指字母最多的单词
中式英语
中国人自创出来的英语
经典英文歌曲
欣赏音乐同时练习听力
英语谚语
英美言简意赅的话语
搜索热点
排名
搜索指数
1 科学家发现新器官 483251
2 曝高云翔悉尼被捕 481352

The Model We are Interested

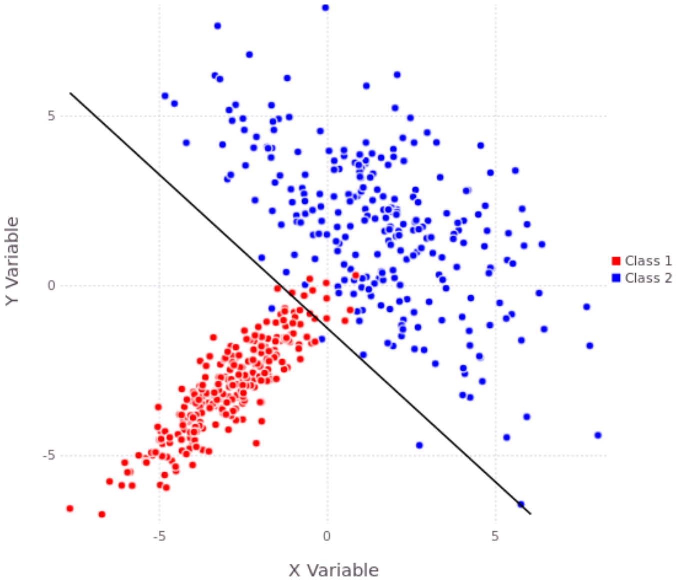


Parametric Model

Parametric models assume some finite set of parameters θ. Given the parameters, future predictions, x, are independent of the observed data, $\mathcal{D}$:

$$P(x|\theta, \mathcal{D}) = P(x|\theta)$$

therefore θ capture everything there is to know about the data.

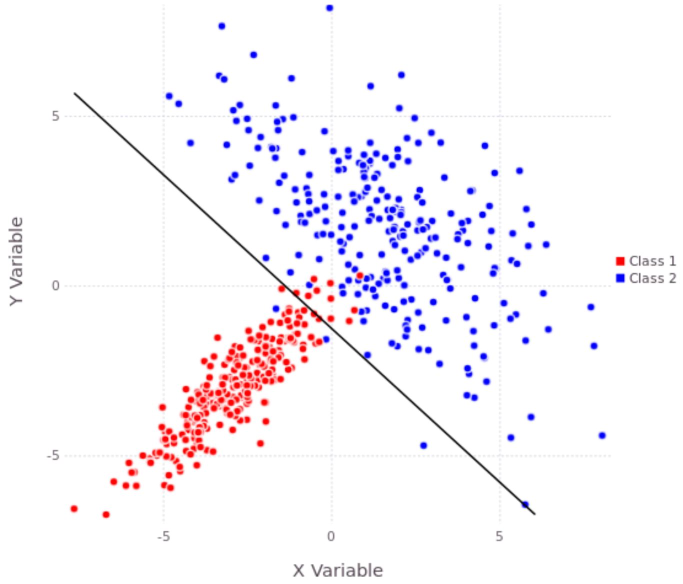


Parametric Model

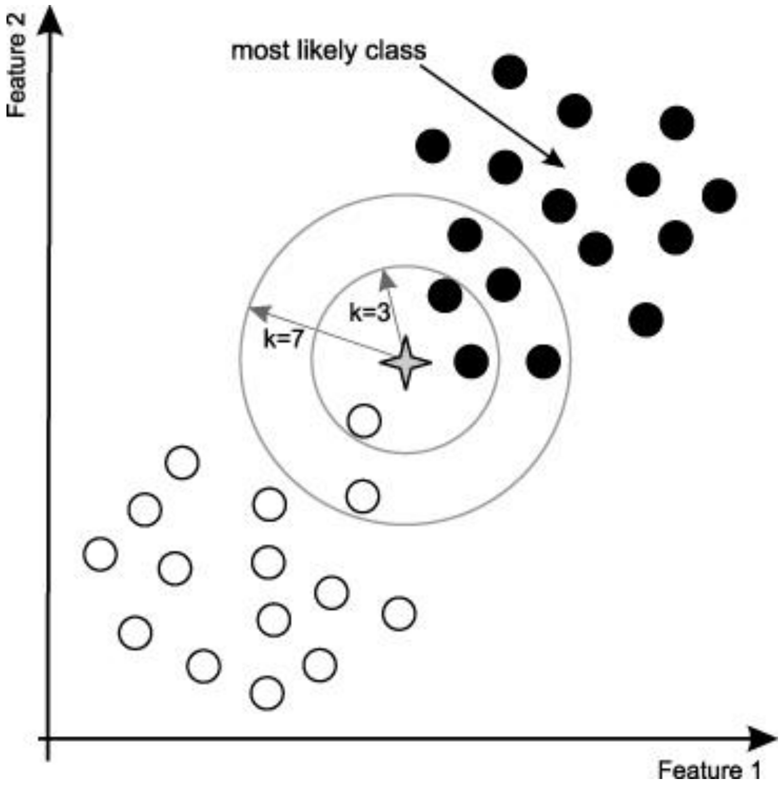
Parametric models assume some finite set of parameters θ. Given the parameters, future predictions, x, are independent of the observed data, $\mathcal{D}$:

$$P(x|\theta, \mathcal{D}) = P(x|\theta)$$

therefore θ capture everything there is to know about the data.



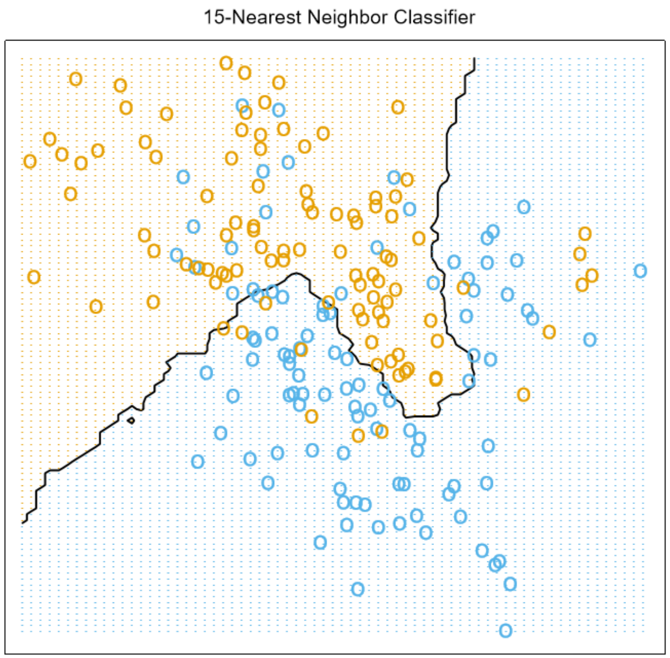
K-Nearest Neighbor (k-NN)

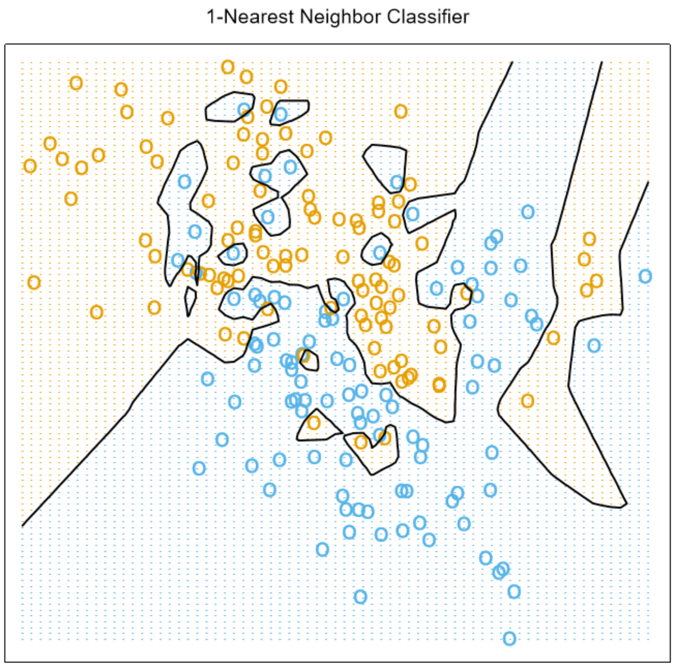


k*-nearest neighbors algorithm** (k*-NN**) is a non-parametric method used for classification and regression. The input consists of the *k* closest training examples in the feature space.

Q: *k* has to be an odd number?

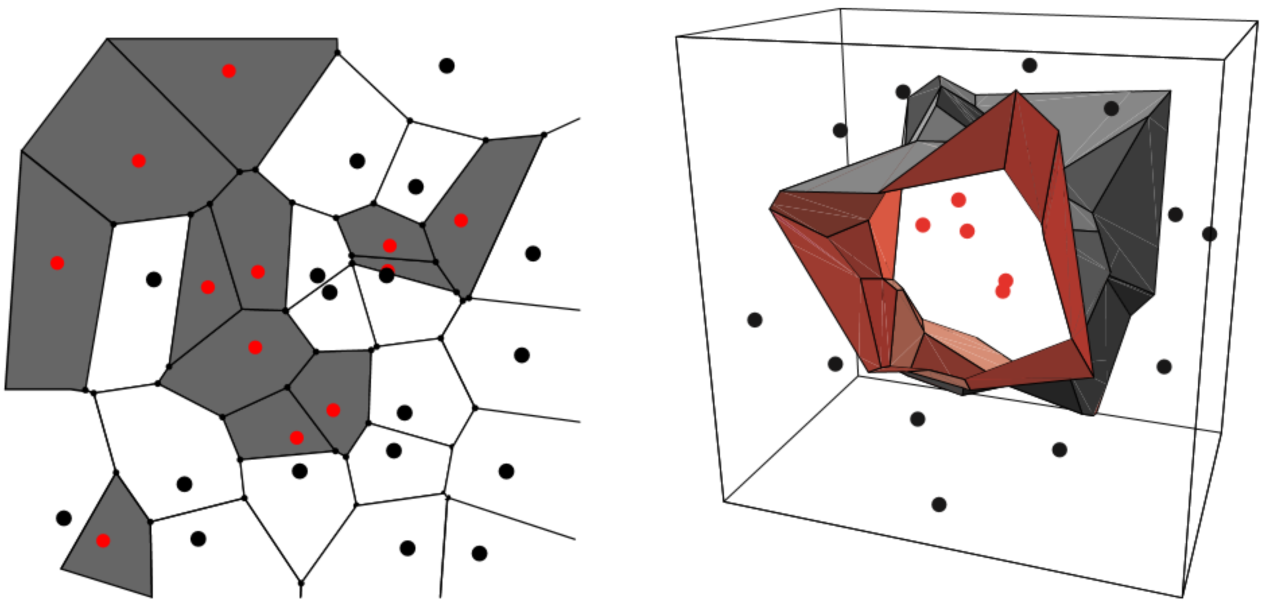
K-Nearest Neighbor (KNN)





Q: What if K becomes very large?

Parametric Model



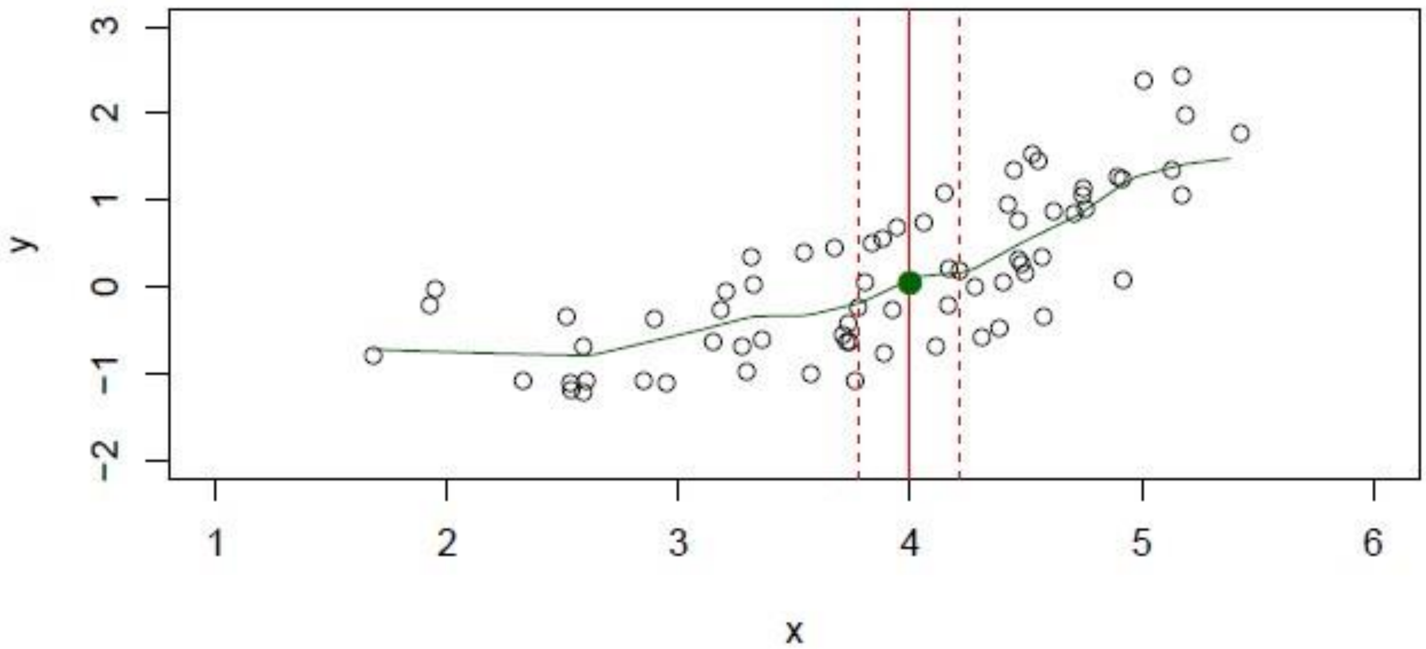
In two dimensions, the nearest-neighbor algorithm leads to a partitioning of the input space into Voronoi cells, each label led by the category of the training point it contains. In three dimensions, the cells are three-dimensional, and the decision boundary resembles the surface of a crystal.

Can k-NN Work for Prediction?

K-Nearest Neighbors for Machine Learning, Photo by [Valentin Ottone]

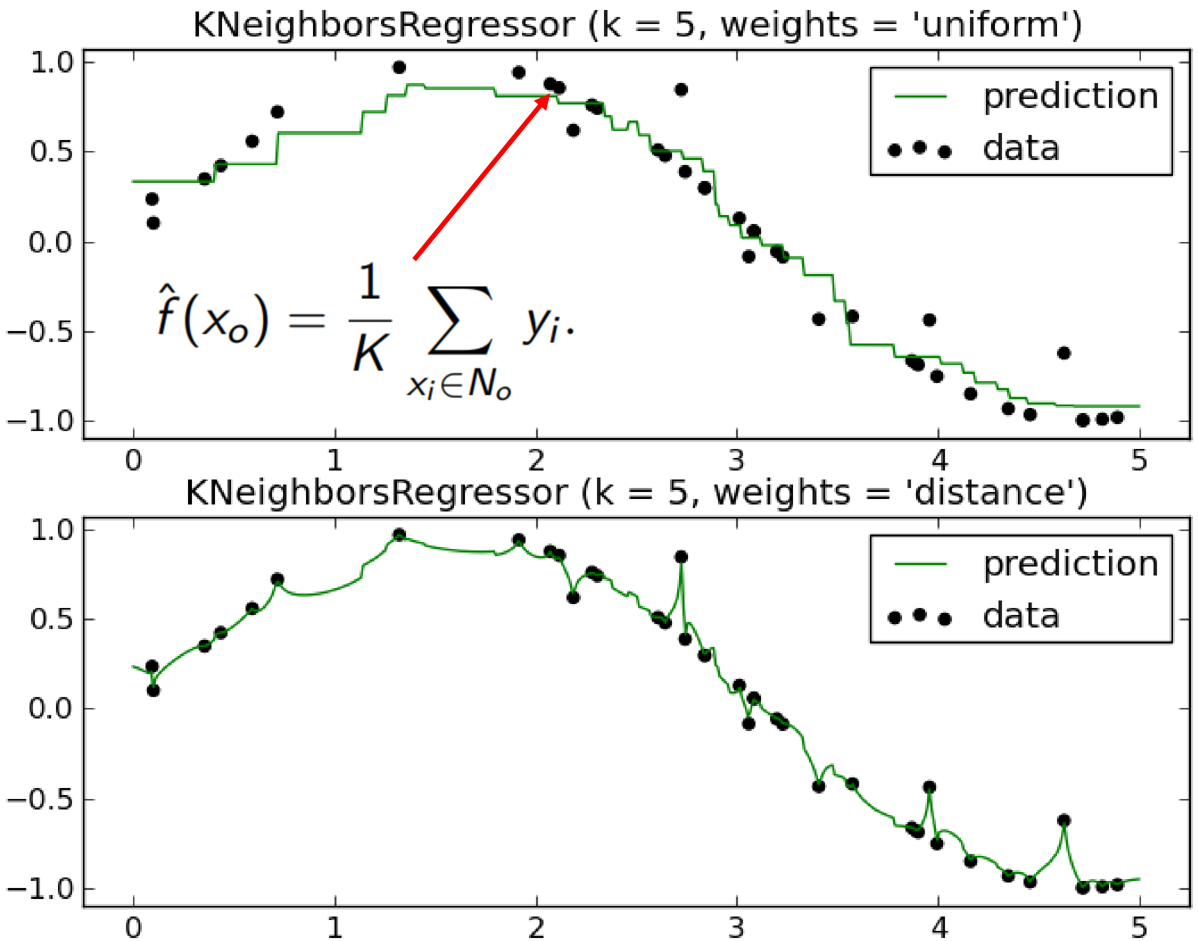
Regression Problem

KNN for Regression (Prediction)



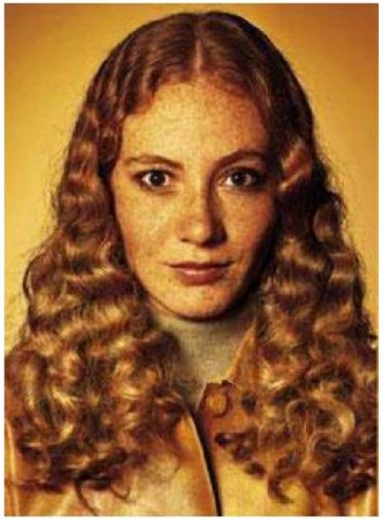
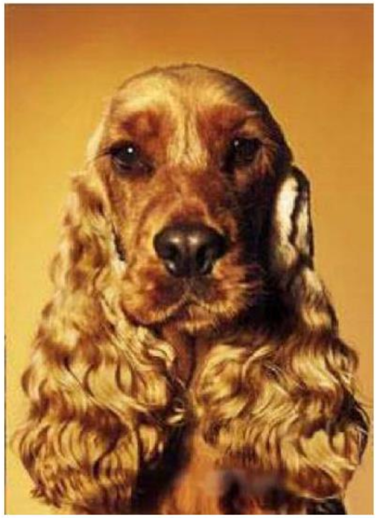
1. Assume a value for the number of nearest neighbors K and a prediction point xo.
2. KNN identifies the training observations No closest to the prediction point xo.
3. KNN estimates f (xo) using the average of all the responses in No

k-NN for Prediction



Q: Any better non-parametric model, do we need to adjust the weights?

Distance Measure

$d(A,B) = d(B,A)$ Symmetry

$d(A,A) = 0$ Constancy of Self-Similarity

$d(A,B) = 0$ iff $A=B$ Positivity Separation

$d(A,B) \leq d(A,C)+d(B,C)$ Triangular Inequality

Distance Measure (Look Familiar?)

Minkowski Distance

$$dist(\mathbf{x},\mathbf{y}) = \sqrt[r]{\sum_{i=1}^{d} |x_i - y_i|^r}$$

Euclidean distance (r=2)

$$dist(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{k=1}^{d} (x_k - y_k)^2} = \|\mathbf{x} - \mathbf{y}\|_2$$

Manhattan distance (r=1)

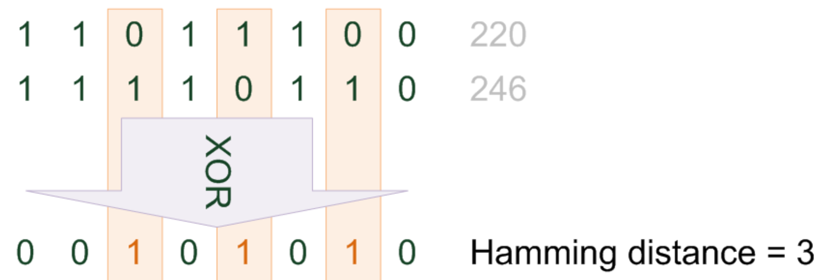
$$dist(\mathbf{x},\mathbf{y}) = \sum_{k=1}^{d} |x_k - y_k| = \|\mathbf{x} - \mathbf{y}\|_1$$

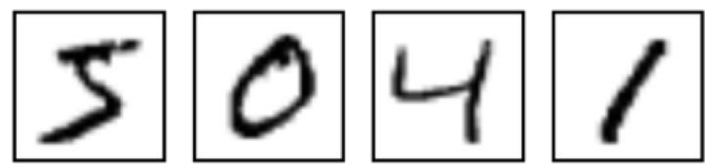
Distance Measure

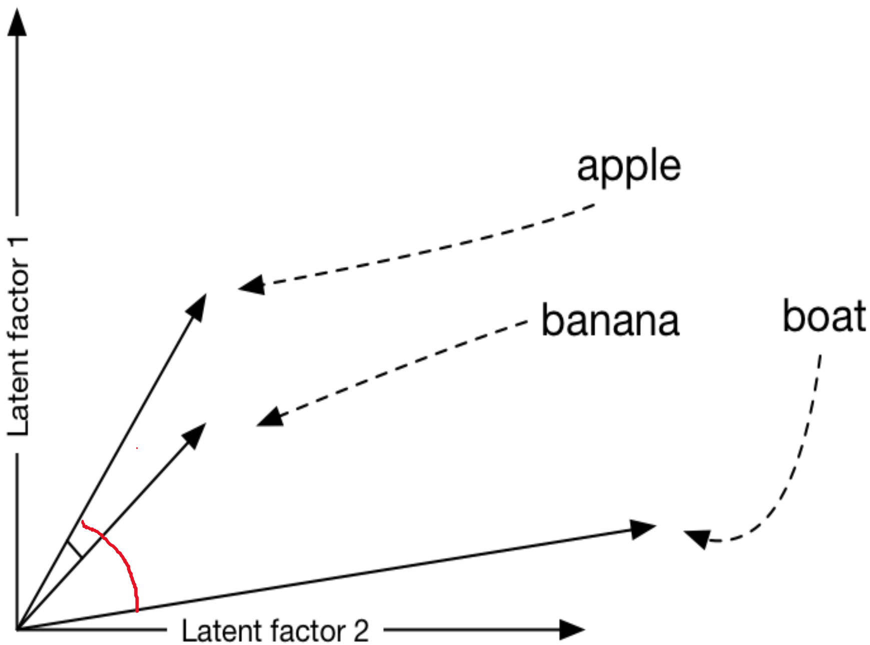
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
GeneA	0	1	1	0	0	1	0	0	1	0	0	1	1	1	0	0	1
GeneB	0	1	1	1	0	0	0	0	1	1	1	1	1	1	0	1	1

Hamming distance

when all features are binary





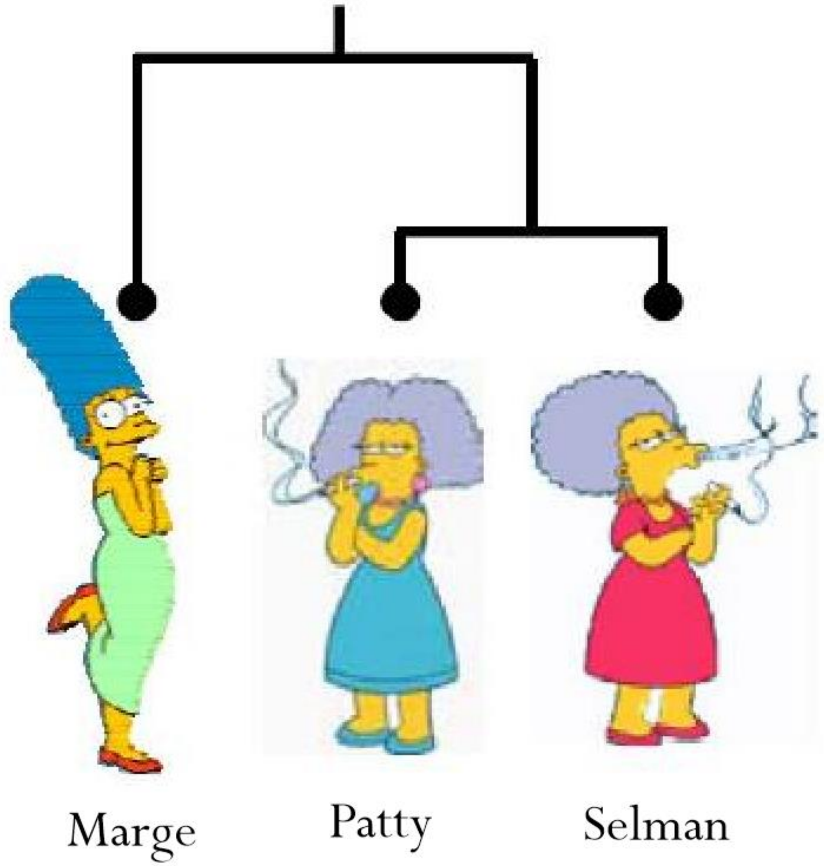


Cosine Similarity

$$\cos(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{x}^\top \mathbf{y}}{\|\mathbf{x}\|_2 \|\mathbf{y}\|_2}$$

Edited Measure

To measure the similarity between two objects, transform one into the other, and measure how much effort it took. The measure of effort becomes the distance measure.



The distance between Marge and Selma

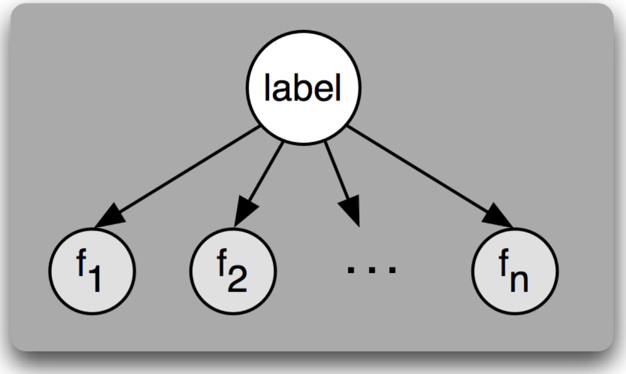
Change dress color, 1 point
Add earrings, 1 point
Decrease height, 1 point
Take up smoking, 1 point
Loss weight, 1 point

D(Marge, Selma) = 5

The distance between Patty and Selma.

Change dress color, 1 point
Change earring shape, 1 point
Change hair part, 1 point

D(Patty,Selma) = 3



Naïve Bayes

Bayes Rule

Likelihood Prior

$$P(Y|X_1,\ldots,X_n) = \frac{P(X_1,\ldots,X_n|Y)P(Y)}{P(X_1,\ldots,X_n)}$$

Normalization Constant

The Naïve Bayes Assumption: Assume that all features are independent given the class label Y:

Equationally speaking:

$$P(X_1,\ldots,X_n|Y) = \prod_{i=1}^{n} P(X_i|Y)$$

Independence assumption – Naïve but effective!

Play-Tennis Problem

PlayTennis: training examples

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

$P(\text{Play}=Yes) = 9/14$

Outlook	Play =*Yes*	Play =*No*
Sunny	2/9	3/5
Overcast	4/9	0/5
Rain	3/9	2/5

Learning

$P(\text{Play}=Yes) = 9/14$ $P(\text{Play}=No) = 5/14$

Outlook	Play=*Yes*	Play=*No*
Sunny	2/9	3/5
Overcast	4/9	0/5
Rain	3/9	2/5

Temperature	Play=*Yes*	Play=*No*
Hot	2/9	2/5
Mild	4/9	2/5
Cool	3/9	1/5

Humidity	Play=*Yes*	Play=*No*
High	3/9	4/5
Normal	6/9	1/5

Wind	Play=*Yes*	Play=*No*
Strong	3/9	3/5
Weak	6/9	2/5

For a new data:

x'=(Outlook=*Sunny*, Temperature=*Cool*, Humidity=*High*, Wind=*Strong*)

Prediction

- Given a new instance, predict its label

  x'=(Outlook=*Sunny*, Temperature=*Cool*, Humidity=*High*, Wind=*Strong*)

- Look up tables achieved in the learning phrase

P(Outlook=*Sunny*|Play=*Yes*) = 2/9

P(Temperature=*Cool*|Play=*Yes*) = 3/9

P(Huminity=*High*|Play=*Yes*) = 3/9

P(Wind=*Strong*|Play=*Yes*) = 3/9

P(Play=*Yes*) = 9/14

P(Outlook=*Sunny*|Play=*No*) = 3/5

P(Temperature=*Cool*|Play==*No*) = 1/5

P(Huminity=*High*|Play=*No*) = 4/5

P(Wind=*Strong*|Play=*No*) = 3/5

P(Play=*No*) = 5/14

- Decision making with the MAP rule

$$P(Yes|\mathbf{x}') \approx [P(Sunny|Yes)P(Cool|Yes)P(High|Yes)P(Strong|Yes)]P(Play{=}Yes) = 0.0053$$

$$P(No|\mathbf{x}') \approx [P(Sunny|No)\,P(Cool|No)P(High|No)P(Strong|No)]P(Play{=}No) = 0.0206$$

Given the fact $P(Yes|\mathbf{x}') < P(No|\mathbf{x}')$, we label $\mathbf{x}'$ to be "*No*".

NB for Spam Filtering

But if we have, say, $y \in \{0,1\}$ for being spam or non-spam email, a vocabulary of 50000 words, then $x \in \{0,1\}^{50000}$:

$$
\begin{aligned}
& p(x_1, \ldots, x_{50000} | y) \\
&\quad = p(x_1|y)p(x_2|y,x_1)p(x_3|y,x_1,x_2)\cdots p(x_{50000}|y,x_1,\ldots,x_{49999}) \\
&\quad = p(x_1|y)p(x_2|y)p(x_3|y)\cdots p(x_{50000}|y) \\
&\quad = \prod_{i=1}^{n} p(x_i|y)
\end{aligned}
$$

$$
x = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix}
\begin{array}{l} \text{a} \\ \text{aardvark} \\ \text{aardwolf} \\ \vdots \\ \text{buy} \\ \vdots \\ \text{zygmurgy} \end{array}
$$

- **Disadvantages**:

– Assumes independence of feature

NB for Spam Filtering

$$\phi_{j|y=1} = \frac{\sum_{i=1}^{m} 1\{x_j^{(i)} = 1 \wedge y^{(i)} = 1\}}{\sum_{i=1}^{m} 1\{y^{(i)} = 1\}}$$

$$\phi_{j|y=0} = \frac{\sum_{i=1}^{m} 1\{x_j^{(i)} = 1 \wedge y^{(i)} = 0\}}{\sum_{i=1}^{m} 1\{y^{(i)} = 0\}}$$

$$\phi_y = \frac{\sum_{i=1}^{m} 1\{y^{(i)} = 1\}}{m}$$

- **Advantages:**

– Fast to train (single scan).
– Fast to classify
– Not sensitive to irrelevant features
– Handles real and discrete data
– Handles streaming data well

- **Smoothing:**

– It always happens that a particular word has not appeared in the given text.
– A smoothing factor is needed, e.g. P = 1/|V|, where |V| = 50000 in this example.