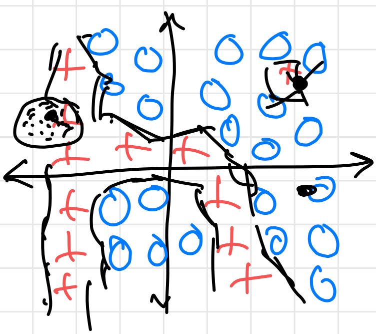
# 9/14/2023 Non-parametric methods

| | Discriminative | Generative |
|---|---|---|
| **Parametric Method** <br> Fixed number of parameters to learn. <br> After learning, training data can be ignored | Logistic Regression <br> Softmax Regression <br> Parameter: $w \in \mathbb{R}^d$ <br> Parameters $w^{(1)}, \ldots, w^{(C)} \in \mathbb{R}^d$ | Naive Bayes <br> $P(y)$ <br> $\pi$ <br> $P(x\|y)$ <br> $\tau$ |
| **Non-parametric** <br> # of paramters / size of model proportional to # of training examples <br> Usually b/c need to use training dataset to make predictions | K-Nearest Neighbors <br> Kernel methods | |



## 1-Nearest Neighbor

Idea: Similar points should have same label

1. Training step: Store training data in memory
2. Test time: Given $x$, find most similar training example (e.g. Euclidean distance), return same label as that training example

Generalization: K-Nearest Neighbors

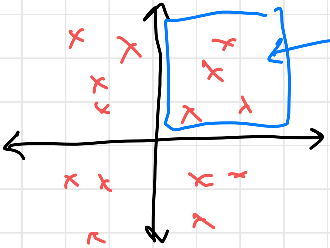
Find K closest points, return most common label among neighbors

## Potential Pitfalls:

- $\underbrace{\text{Bias}}_{\text{usually low}}$ vs $\underbrace{\text{Variance}}_{\text{can be very significant}}$

- Curse of Dimensionality

In high dimensions, you very rarely have close neighbors



1/4 of points in each quadrant in $\mathbb{R}^2$

If $d = 1000$, then only $\frac{1}{2^{1000}}$ points in each quadrant of $\mathbb{R}^{1000}$

No close neighbors in high-dim space

$\Rightarrow$ Even closest neighbor's label may not be same

---

| K-NN | Logistic Regression |
|---|---|
| Intuition: Similar points have similar labels | - Only learn linear decision boundary |
| - No parameters to learn to boost performance | - Parameters get learned from data |
| - No good ways to regularize | - Regularization ($L_2$) |

## Kernel Methods

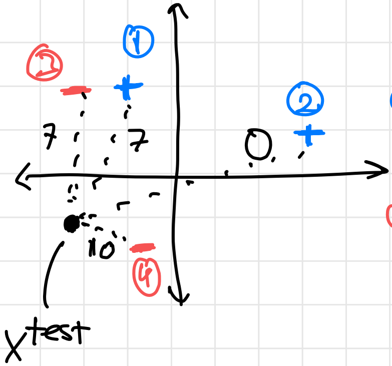
Make a prediction on example $x^{test}$ by computing:

$$\sum_{i=1}^{n} \underbrace{\alpha_i}_{\text{parameter}} \underbrace{K(x^{(i)}, x^{test})}_{\text{"kernel function"}}$$

"kernel function" ← measures similarity between 2 points

If $>0$, predict $y=1$

If $<0$, predict $y=-1$




$\alpha_1 = +1$
$\alpha_2 = +1$
$\alpha_3 = -1$
$\alpha_4 = -1$

$1 \cdot 7 + 1 \cdot 0 + -1 \cdot 7 + -1 \cdot 10$

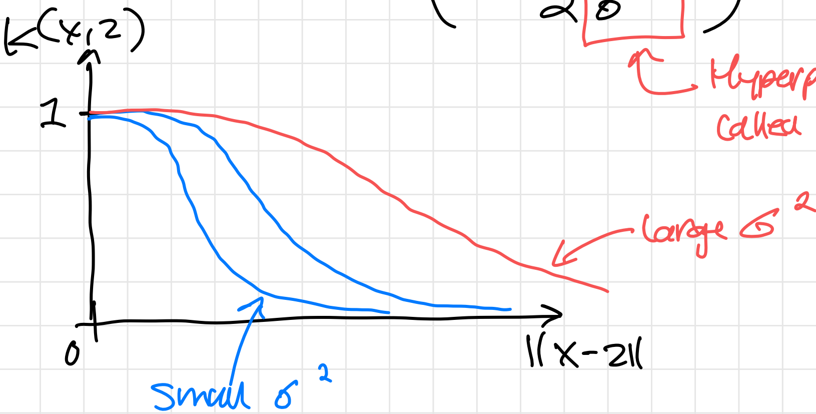
$= -10$

$\Rightarrow$ predict $y = -1$

Popular kernel function: Radial Basis Function kernel (RBF)

$$K(x,z) = \exp\left(-\frac{\|x-z\|^2}{2\boxed{\sigma^2}}\right)$$

Hyperparameter called "bandwidth"




Recap: Logistic Regression

Training time:

$$w^{(t+1)} \leftarrow w^{(t)} + \eta \frac{1}{n} \sum_{i=1}^{n} \sigma\left(-y^{(i)} w^{(t)T} x^{(i)}\right) \cdot y^{(i)} \cdot x^{(i)}$$

Test time: Compute $w^T x^{test}$, If $>0$, predict $+1$
else, predict $-1$

Claim: I can rewrite this so that x's only appear in dot products with other x's

Let's define $K(x, z) = x^T z$

I will rewrite logistic regression to only have x's inside $K(\cdot, \cdot)$

$w^{(0)} = 0$ vector

$$w^{(t+1)} \leftarrow w^{(t)} + \eta \frac{1}{n} \sum_{i=1}^{n} \underbrace{\sigma\left(-y^{(i)} \boxed{w^{(t)T} x^{(i)}}\right) \cdot y^{(i)}}_{\text{scalar}} \cdot x^{(i)}$$

$$\sum_{j=1}^{n} \alpha_j^{(t)} K(x^{(j)}, x^{(i)})$$

Every update adds $c_1 x^{(1)} + c_2 x^{(2)} + c_3 x^{(3)} + \cdots$ to $w$

Define $\alpha_i^{(t)}$ = How many copies of $x^{(i)}$ got added to $w$ by time $t$

then: $w = \sum_{i=1}^{n} \alpha_i x^{(i)}$

At test time: compute $w^T x^{test}$

$$= \left(\sum_{i=1}^{n} \alpha_i x^{(i)}\right)^T x^{test} = \sum_{i=1}^{n} \alpha_i K(x^{(i)}, x^{test})$$