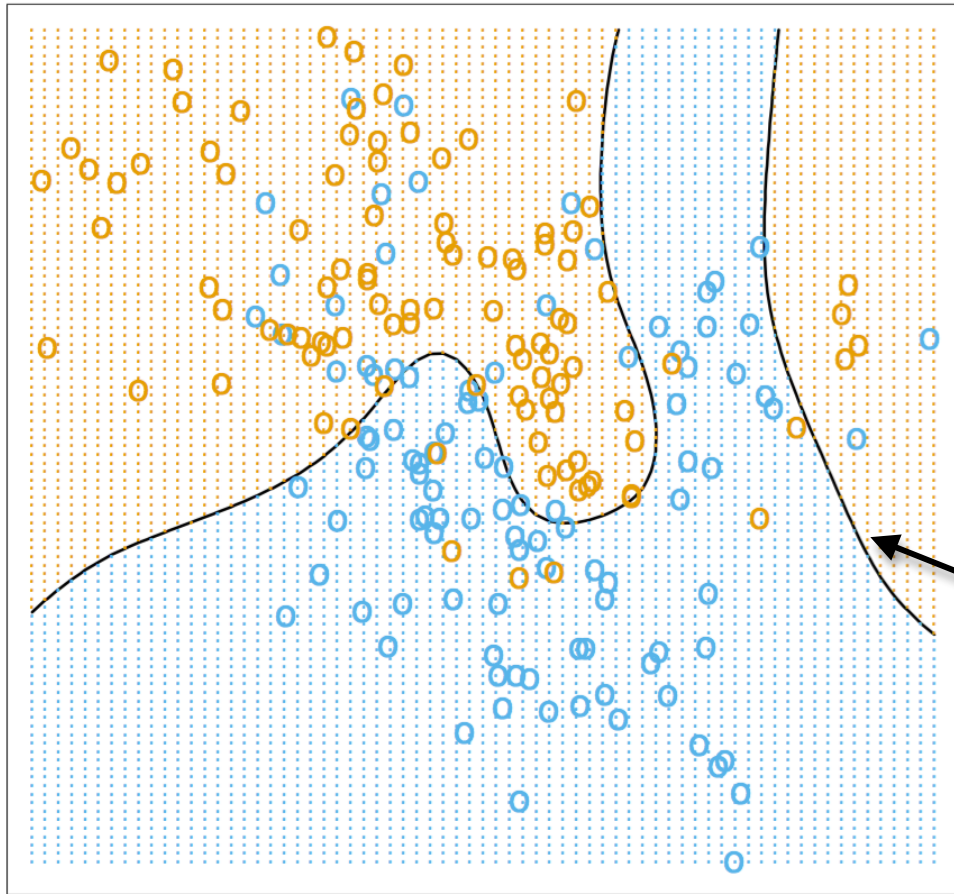


# Nearest Neighbor Methods

---

- A model is called “parametric” if the number of parameters do not depend on the number of samples
- A model is called “non-parametric” if the number of parameters increase with the number of samples (Does **not** mean absence of parameters!)

# Some data, Bayes Classifier



Training data:

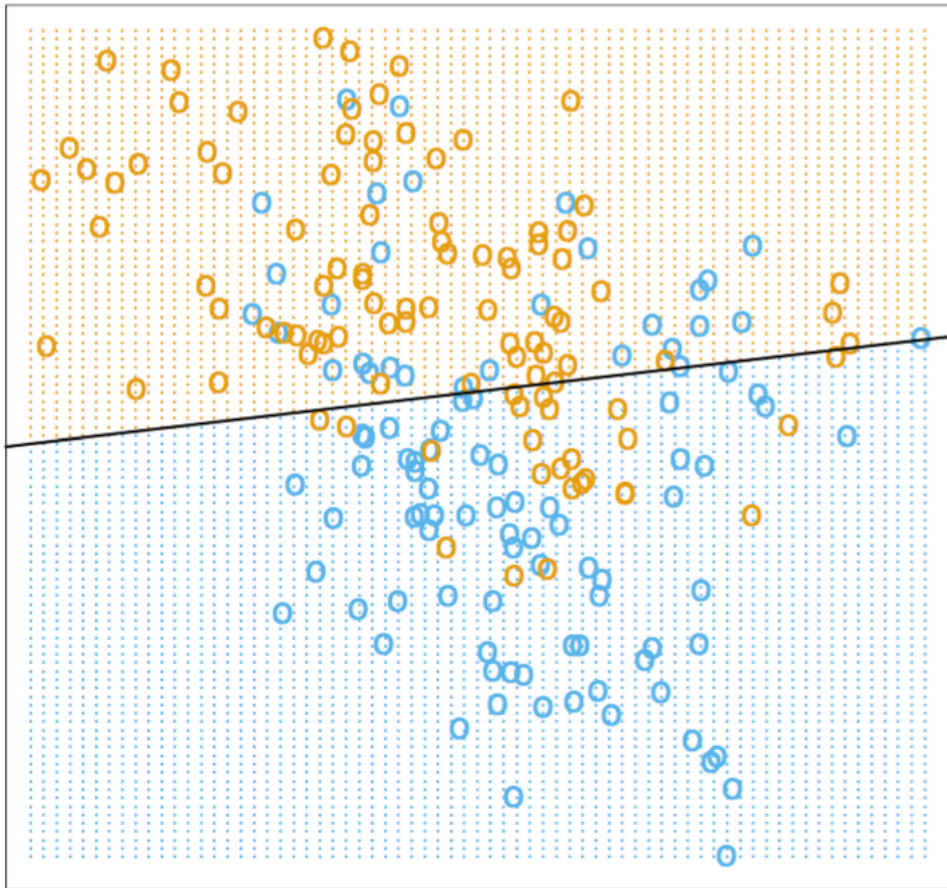
- True label: +1
- True label: -1

Optimal “Bayes” classifier:

$$\mathbb{P}(Y = 1|X = x) = \frac{1}{2}$$

- Predicted label: +1
- Predicted label: -1

# Linear Decision Boundary



Training data:

- True label: +1
- True label: -1

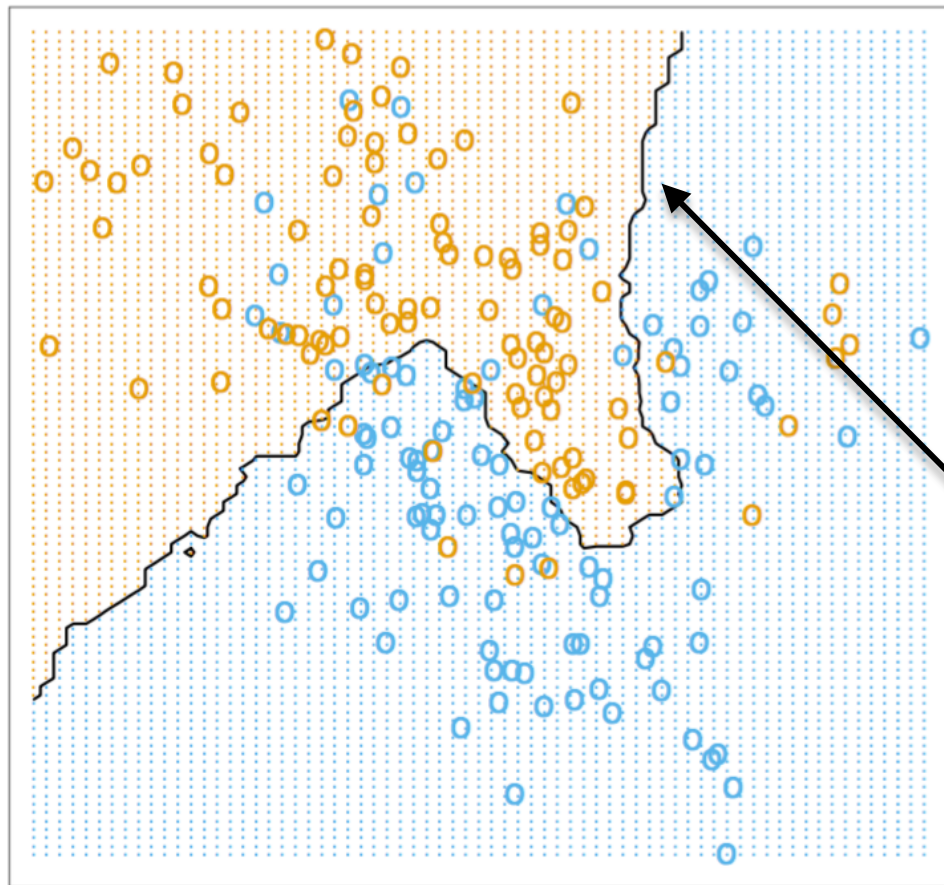
Learned:

Linear Decision boundary

$$x^T w + b = 0$$

- ▢ Predicted label: +1
- ▢ Predicted label: -1

# $k=15$ Nearest Neighbor Boundary



Training data:

- True label: +1
- True label: -1

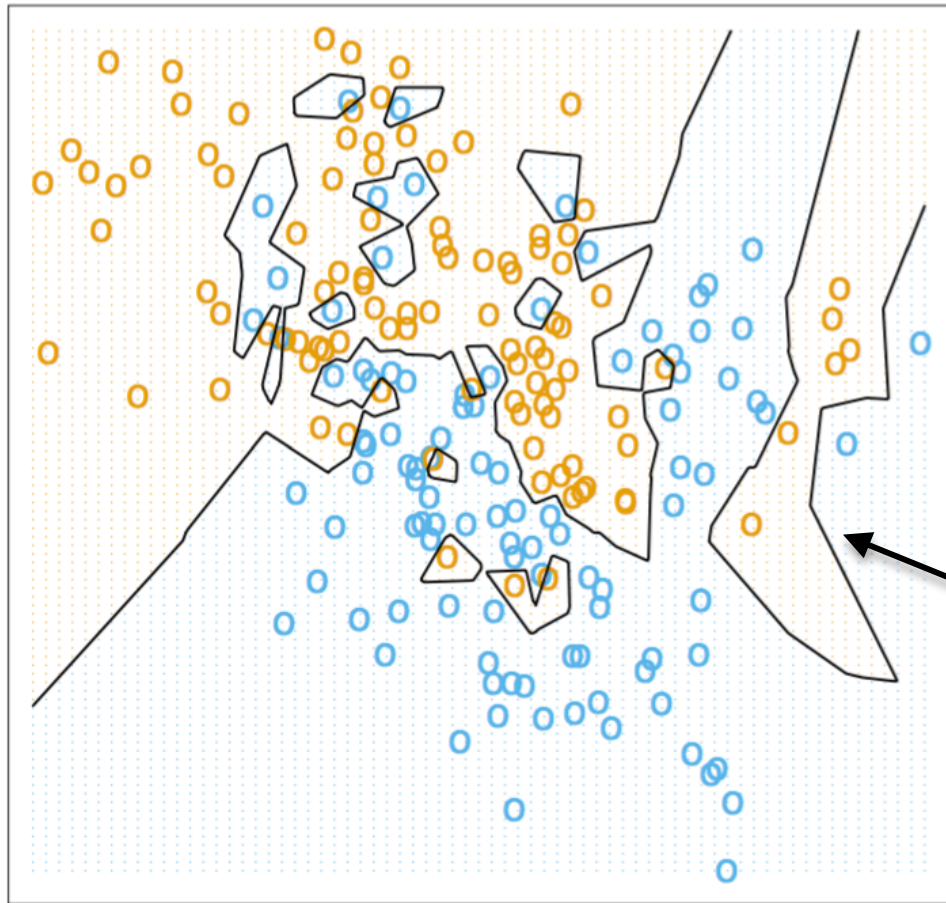
Learned:

**15** nearest neighbor decision boundary (majority vote)

- Predicted label: +1
- Predicted label: -1

- Nearest neighbor gives non-linear decision boundaries
- What happens if we use a small  $k$  or a large  $k$ ?

# k=1 Nearest Neighbor Boundary



Training data:

- True label: +1
- True label: -1

Learned:

1 nearest neighbor decision boundary (majority vote)

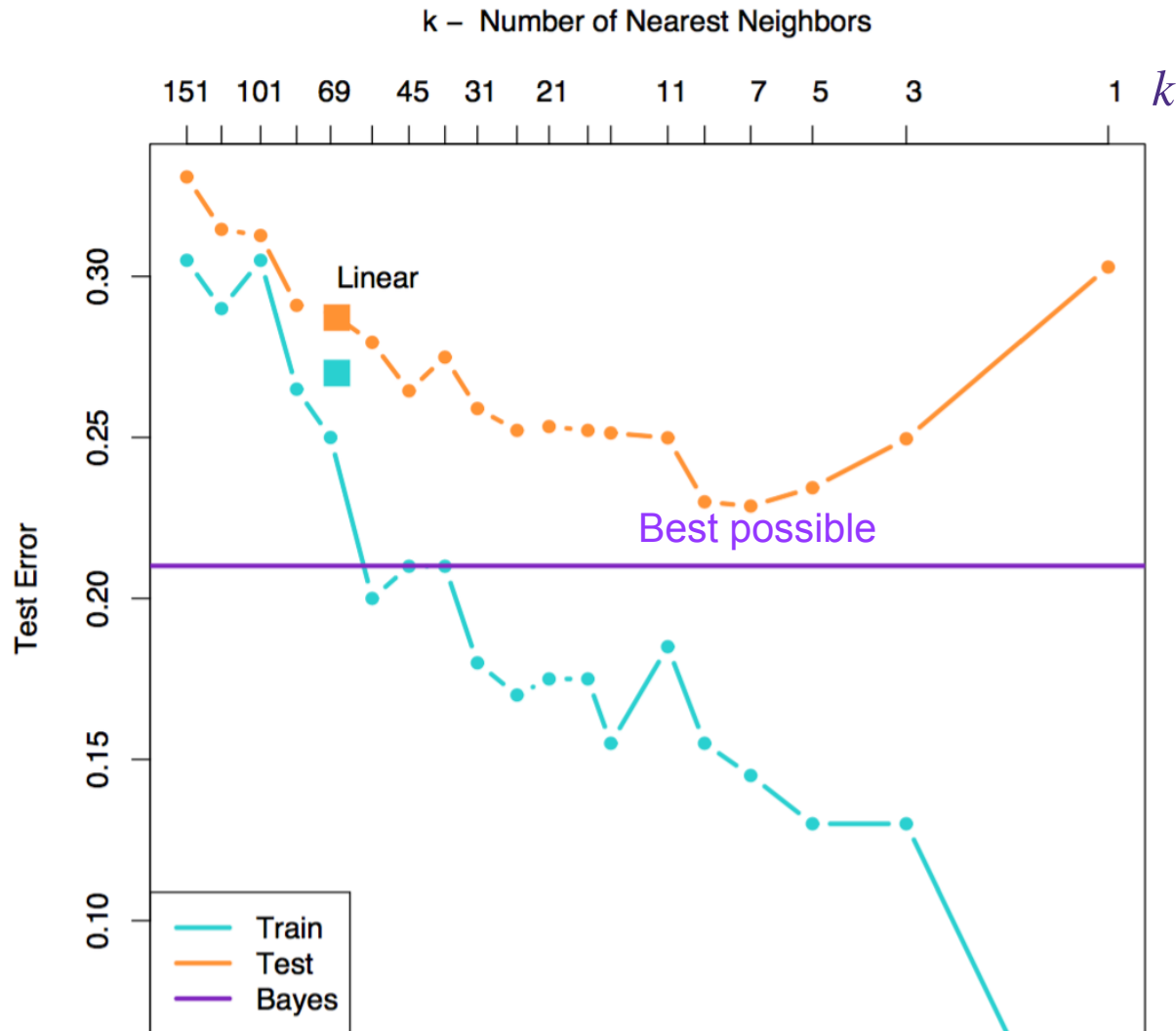
- Predicted label: +1
- Predicted label: -1

- With a small  $k$ , we tend to overfit.

# k-Nearest Neighbor Error

Model complexity low

Model complexity high



Bias-Variance tradeoff

As  $k \rightarrow \infty$ ?

Bias: *high*

Variance: *low*

As  $k \rightarrow 1$ ?

Bias: *basically 0*

Variance: *high*

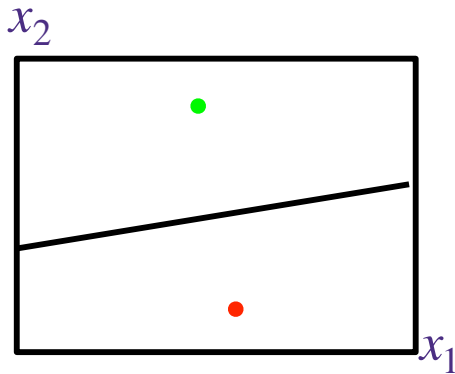
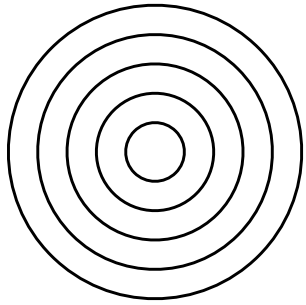
Figures from Hastie et al

- The error achieved by Bayes optimal classifier provides a lower bound on what any estimator can achieve

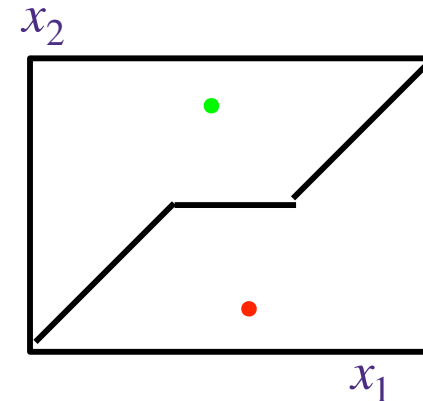
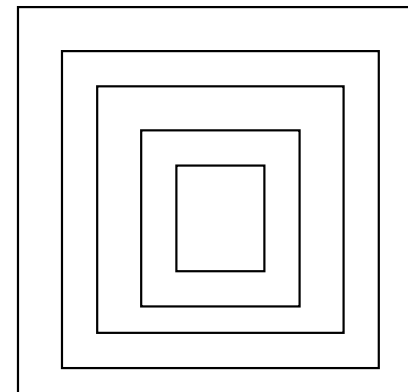
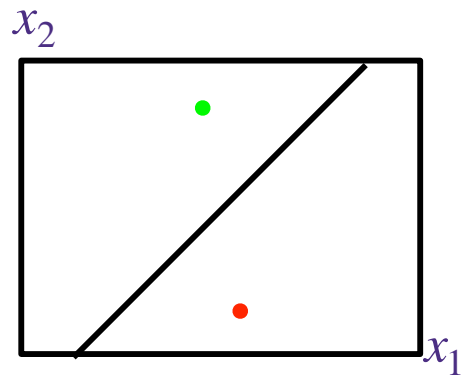
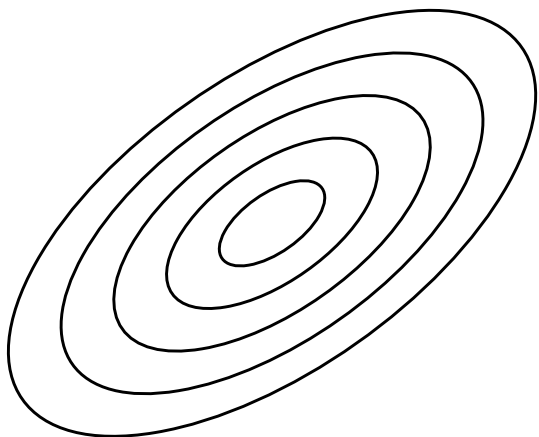
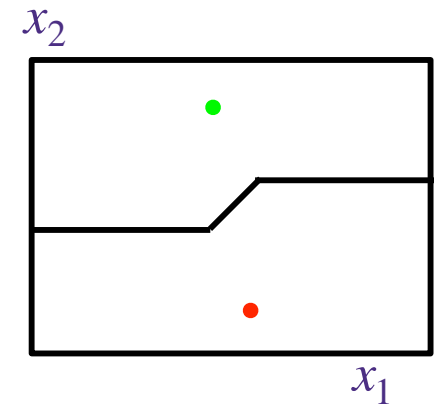
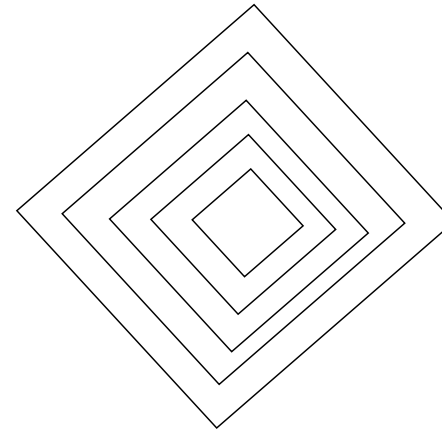
# Notable distance metrics (and their level sets)

Consider 2 dimensional example with 2 data points with labels green, red, and we show  $k = 1$  nearest neighbor decision boundaries for various choices of distances

**L<sub>2</sub> norm** :  $d(x, y) = \|x - y\|_2$



**L<sub>1</sub> norm (taxi-cab)**

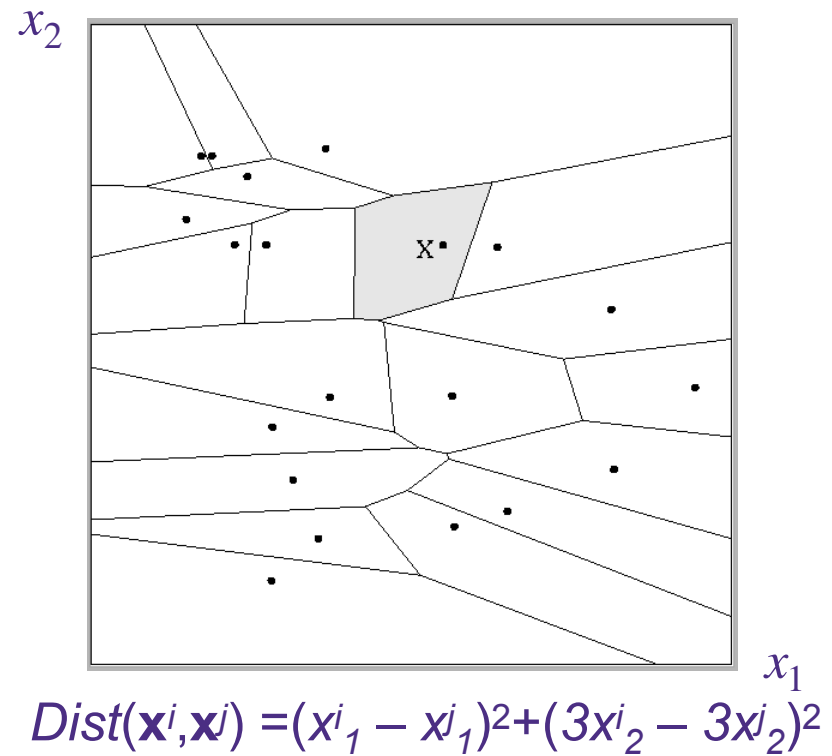
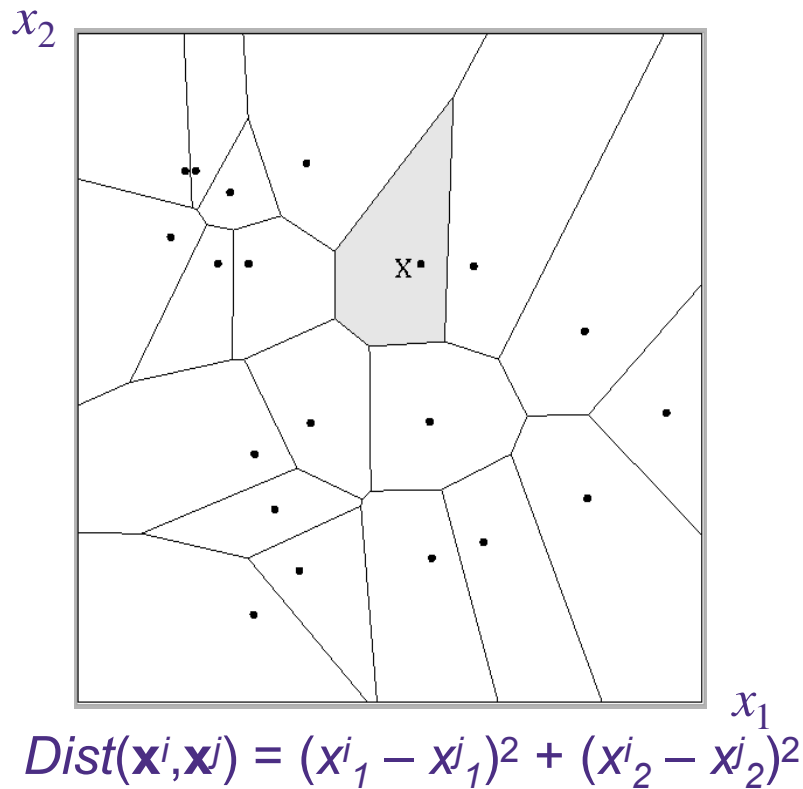


**Mahalanobis norm**:  $d(x, y) = (x - y)^T M (x - y)$

**L-infinity (max) norm**

# $k = 1$ nearest neighbor

One can draw the nearest-neighbor regions in input space.



The relative scalings in the distance metric affect region shapes

# 1 nearest neighbor guarantee - classification

---

$$\{(x_i, y_i)\}_{i=1}^n \quad x_i \in \mathbb{R}^d, \quad y_i \in \{0, 1\} \quad (x_i, y_i) \stackrel{iid}{\sim} P_{XY}$$

**Theorem**[Cover, Hart, 1967] If  $P_X$  is supported everywhere in  $\mathbb{R}^d$  and  $P(Y = 1|X = x)$  is smooth everywhere, then as  $n \rightarrow \infty$  the 1-NN classification rule has error at most twice the Bayes error rate.

# 1 nearest neighbor guarantee - classification

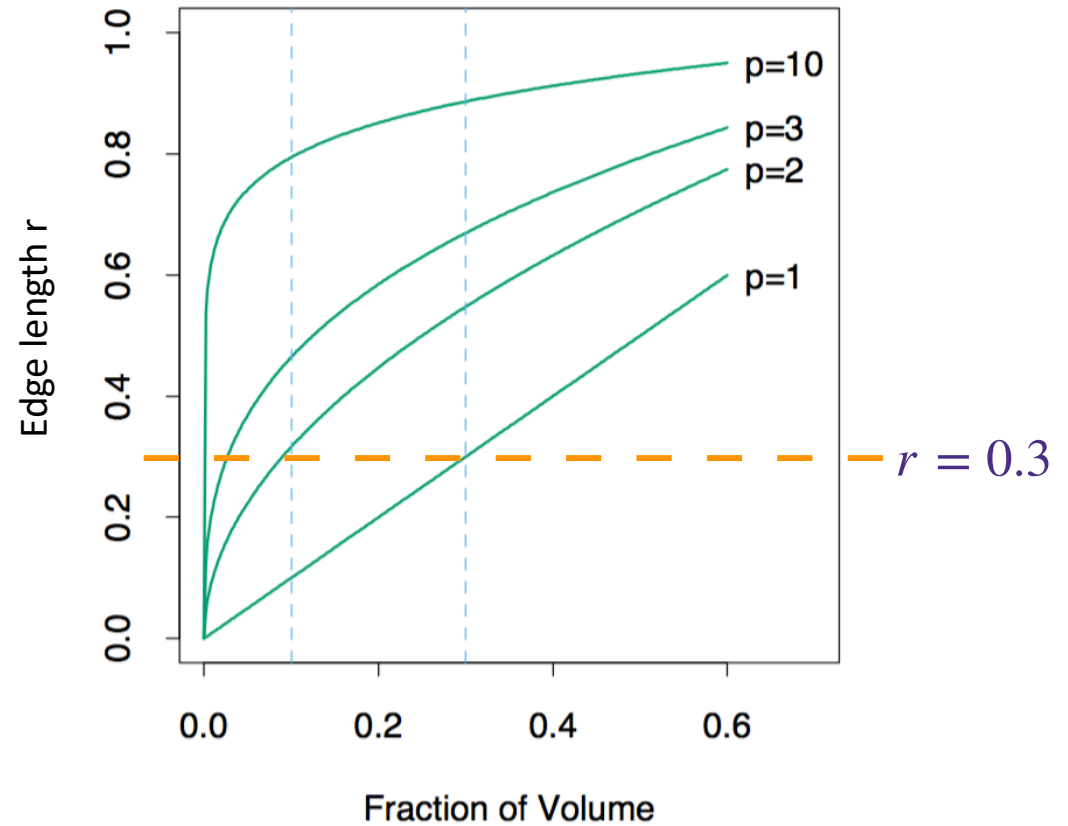
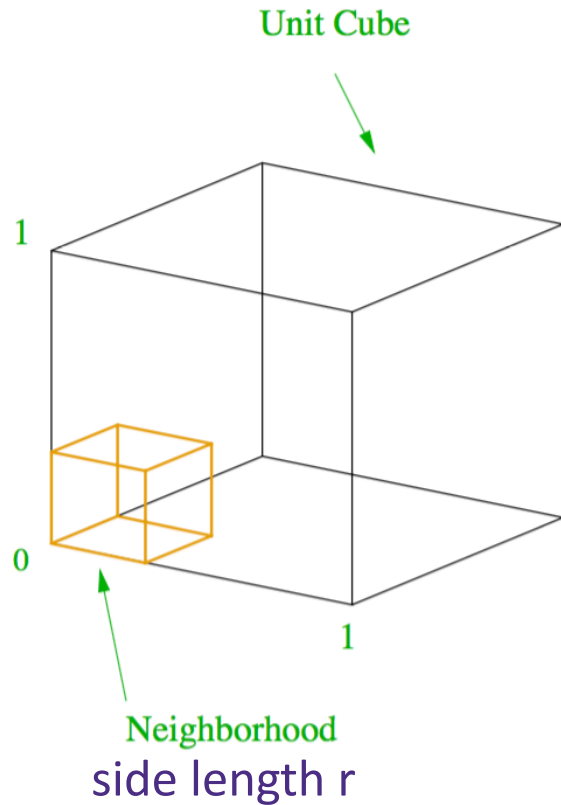
$$\{(x_i, y_i)\}_{i=1}^n \quad x_i \in \mathbb{R}^d, \quad y_i \in \{0, 1\} \quad (x_i, y_i) \stackrel{iid}{\sim} P_{XY}$$

**Theorem**[Cover, Hart, 1967] If  $P_X$  is supported everywhere in  $\mathbb{R}^d$  and  $P(Y = 1|X = x)$  is smooth everywhere, then as  $n \rightarrow \infty$  the 1-NN classification rule has error at most twice the Bayes error rate.

- Let  $x_{NN}$  denote the nearest neighbor at a point  $x$
- First note that as  $n \rightarrow \infty$ ,  $P(y = +1 | x_{NN}) \rightarrow P(y = +1 | x)$
- Let  $p^* = \min\{P(y = +1 | x), P(y = -1 | x)\}$  denote the Bayes error rate
- At a point  $x$ ,
  - Case 1: nearest neighbor is +1, which happens with  $P(y = +1 | x)$  and the error rate is  $P(y = -1 | x)$
  - Case 2: nearest neighbor is -1, which happens with  $P(y = -1 | x)$  and the error rate is  $P(y = +1 | x)$
- The average error of a 1-NN is

$$P(y = +1 | x) P(y = -1 | x) + P(y = -1 | x) P(y = +1 | x) = 2p^*(1 - p^*)$$

# Curse of dimensionality Ex. 1

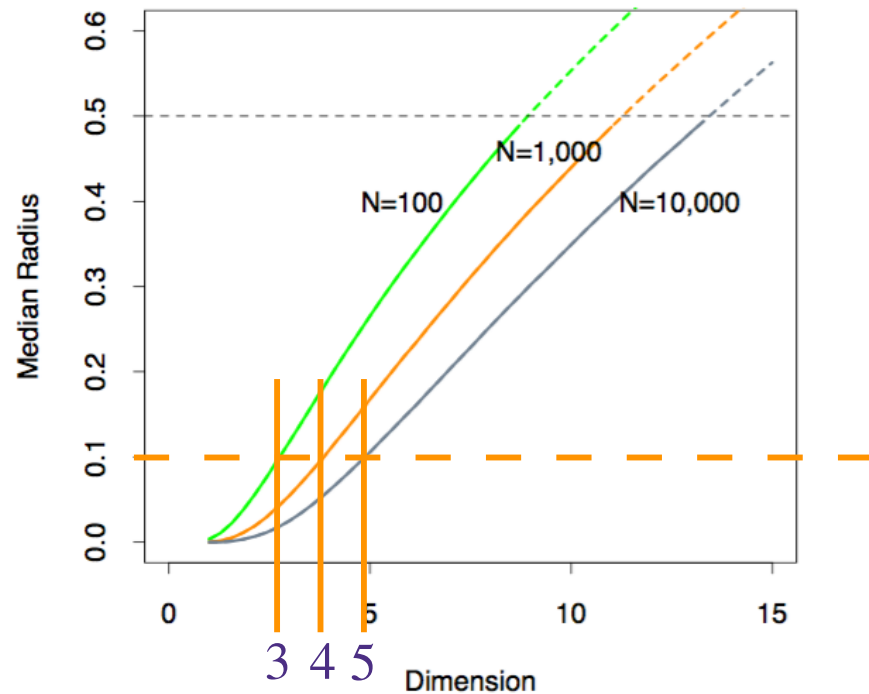
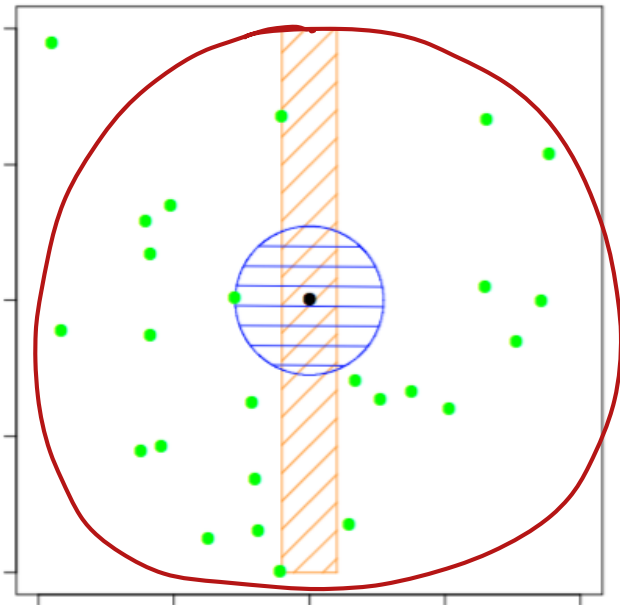


$X$  is uniformly distributed over  $[0, 1]^p$ . What is  $\mathbb{P}(X \in [0, r]^p)$ ?

How many samples do we need so that a nearest neighbor is within a cube of side length  $r$ ?

# Curse of dimensionality Ex. 2

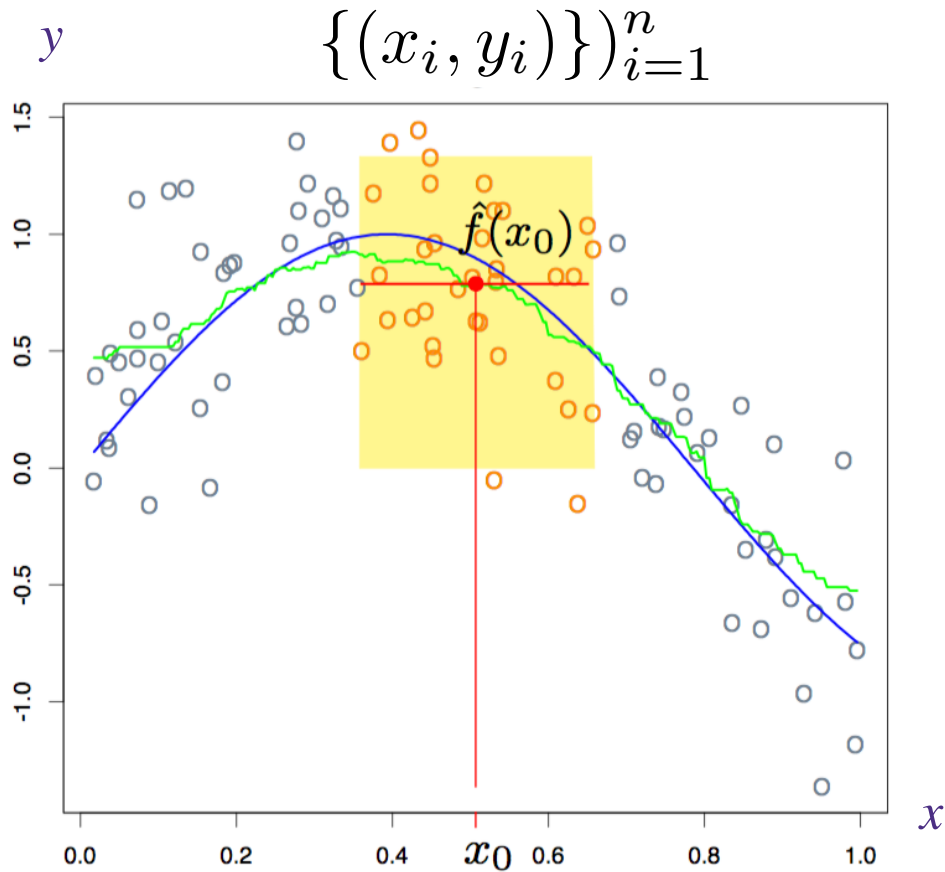
$\{X_i\}_{i=1}^n$  are uniformly distributed over  $[-.5, .5]^p$ .



What is the median distance from a point at origin to its 1NN?

How many samples do we need so that a median Euclidean distance is within  $r$ ?

# Nearest neighbor regression



- What is the optimal classifier that minimizes MSE  $\mathbb{E}[(\hat{y} - y)^2]$ ?

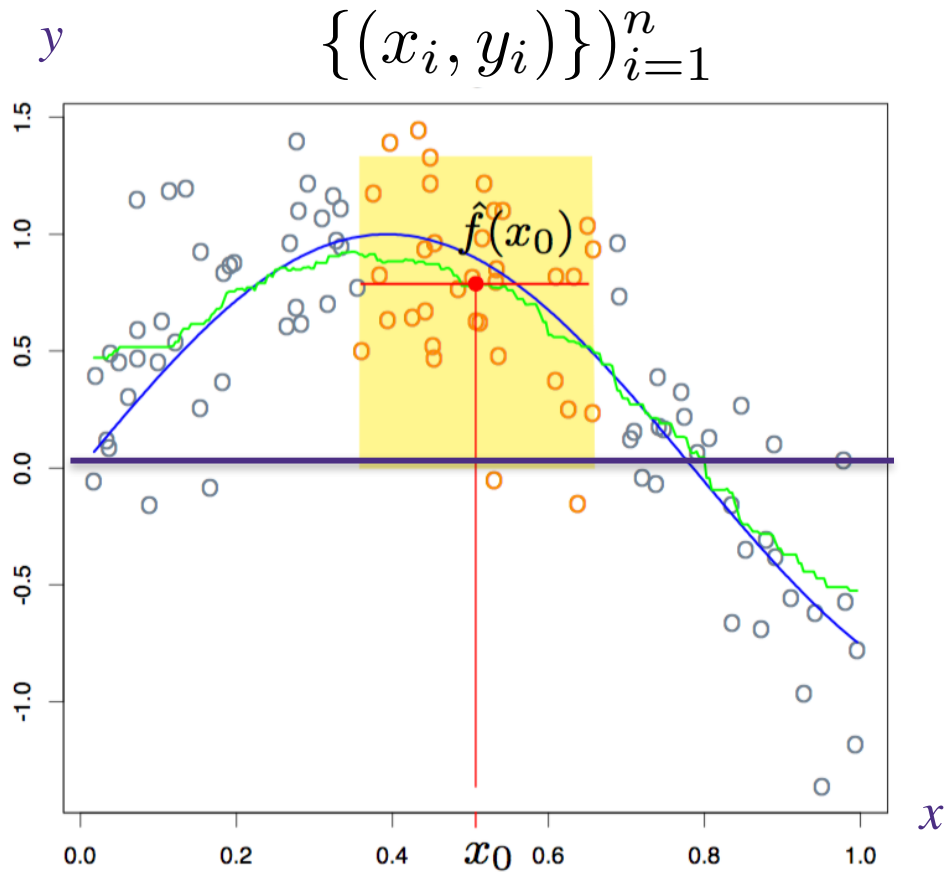
$$\hat{y} = \mathbb{E}[y | x]$$

- $k$ -nearest neighbor regressor is

$$\hat{f}(x) = \frac{1}{k} \sum_{j \in \text{nearest neighbor}} y_j$$

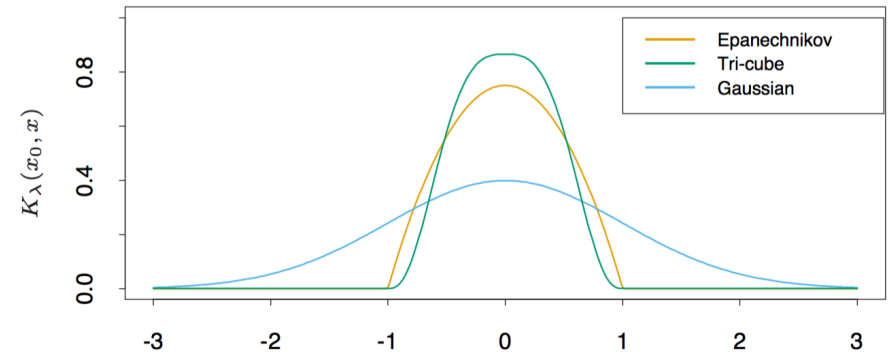
$$= \frac{\sum_{i=1}^n y_i \times \text{Ind}(x_i \text{ is a } k \text{ nearest neighbor})}{\sum_{i=1}^n \text{Ind}(x_i \text{ is a } k \text{ nearest neighbor})}$$

# Nearest neighbor regression



In nearest neighbor methods, the “weight” changes abruptly

smoothing:  $K(x, y)$

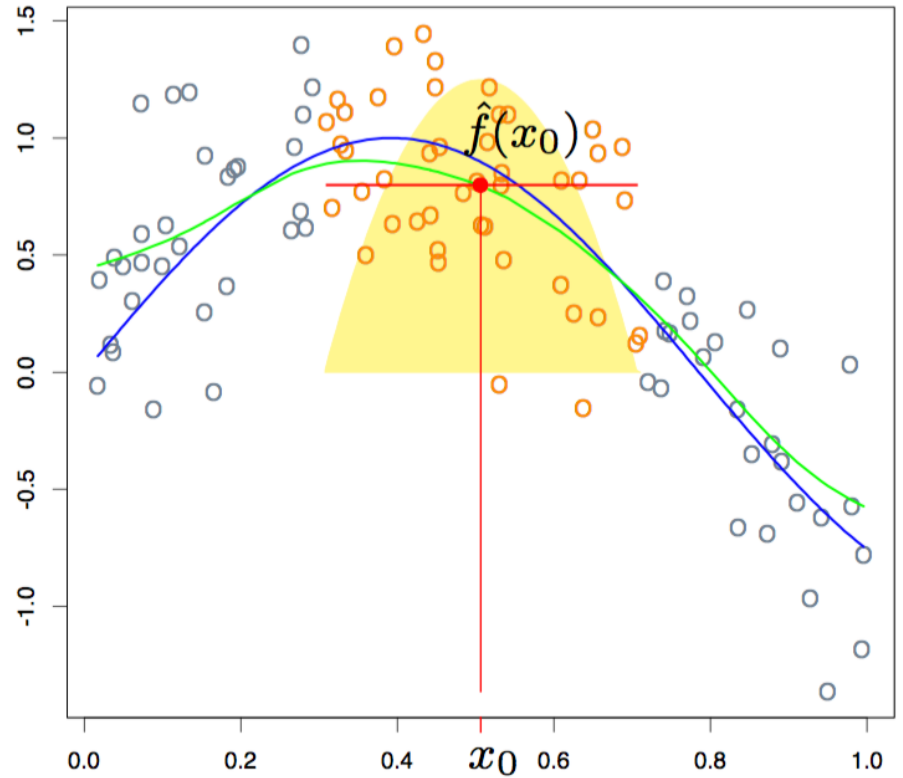
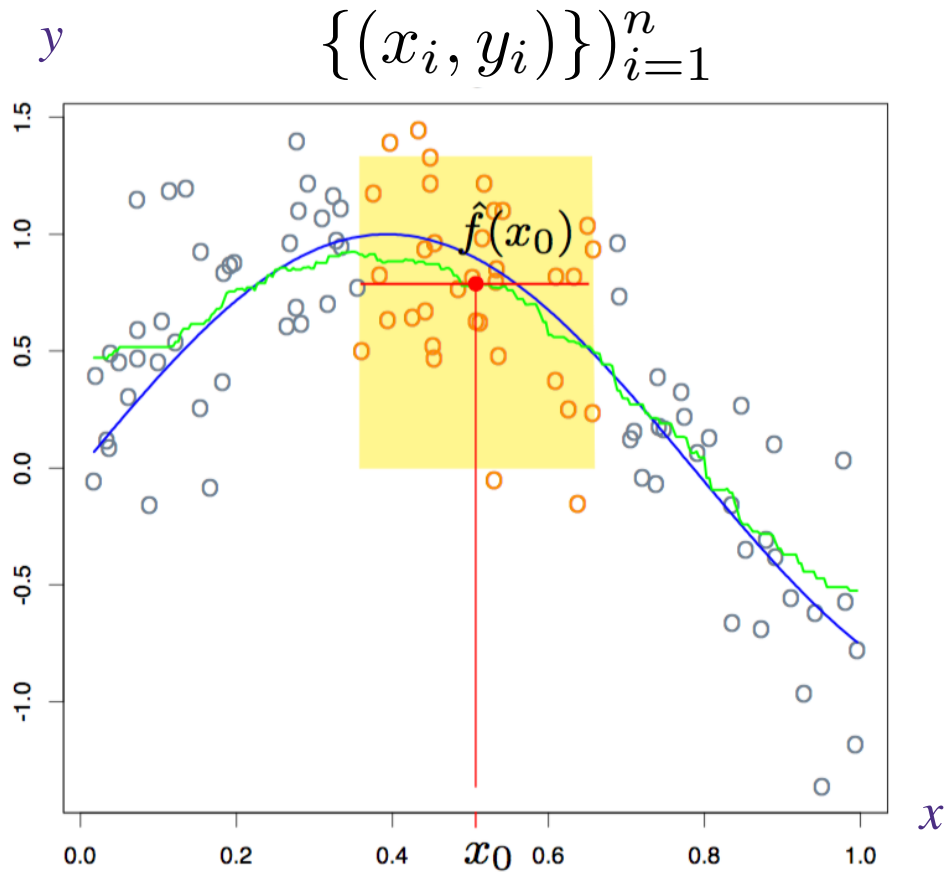


- $k$ -nearest neighbor regressor is

$$\hat{f}(x_0) = \frac{\sum_{i=1}^n y_i \times \text{Ind}(x_i \text{ is a } k \text{ nearest neighbor})}{\sum_{i=1}^n \text{Ind}(x_i \text{ is a } k \text{ nearest neighbor})}$$

$$\hat{f}(x_0) = \frac{\sum_{i=1}^n K(x_0, x_i) y_i}{\sum_{i=1}^n K(x_0, x_i)}$$

# Nearest neighbor regression

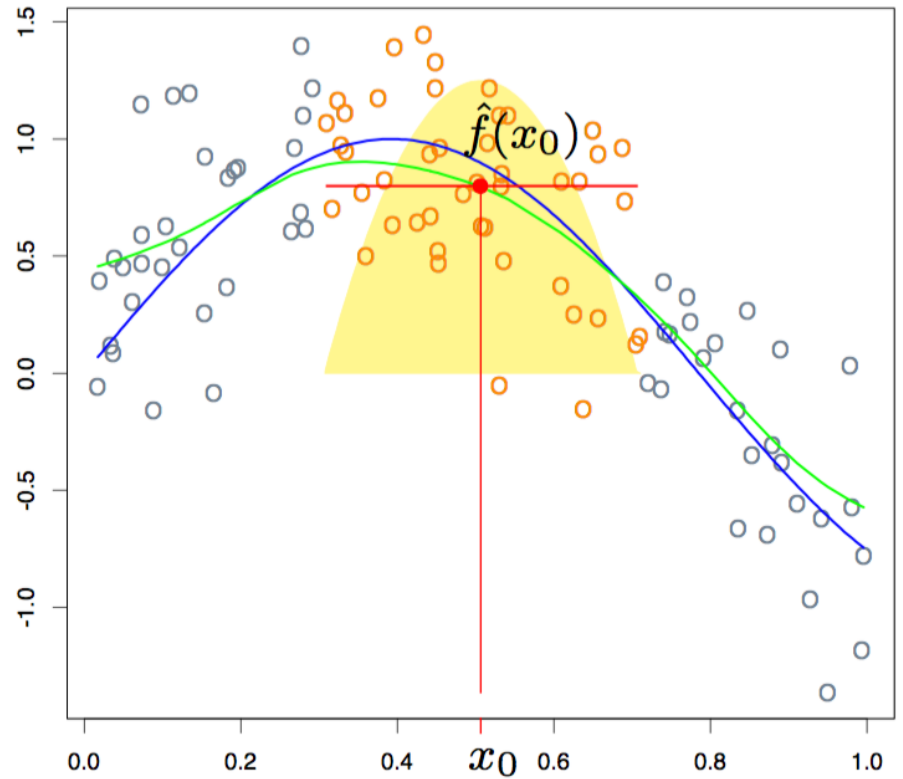
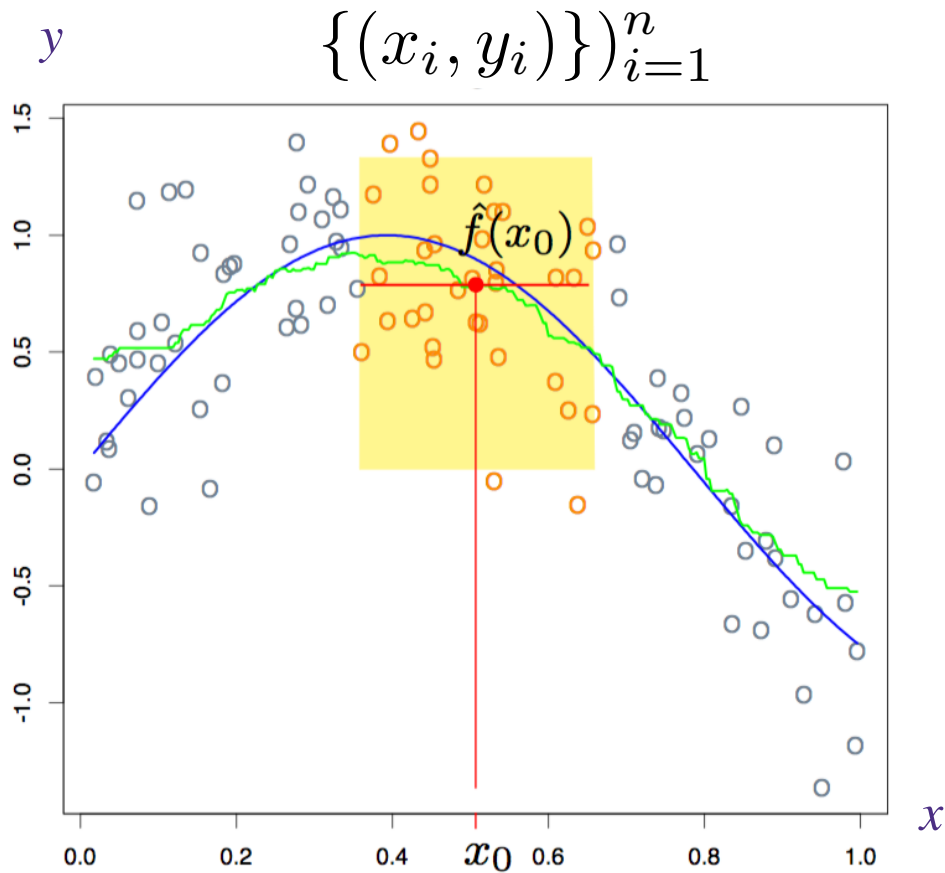


- $k$ -nearest neighbor regressor is

$$\hat{f}(x_0) = \frac{\sum_{i=1}^n y_i \times \text{Ind}(x_i \text{ is a } k \text{ nearest neighbor})}{\sum_{i=1}^n \text{Ind}(x_i \text{ is a } k \text{ nearest neighbor})}$$

$$\hat{f}(x_0) = \frac{\sum_{i=1}^n K(x_0, x_i) y_i}{\sum_{i=1}^n K(x_0, x_i)}$$

# Nearest neighbor regression



Why just average them?

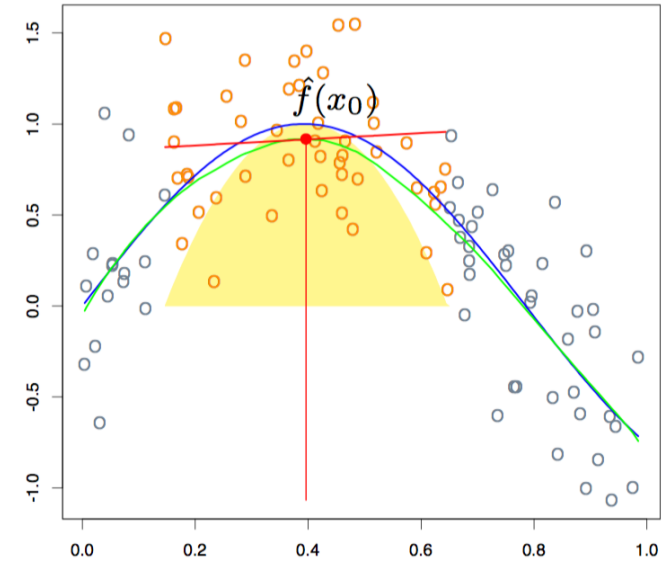
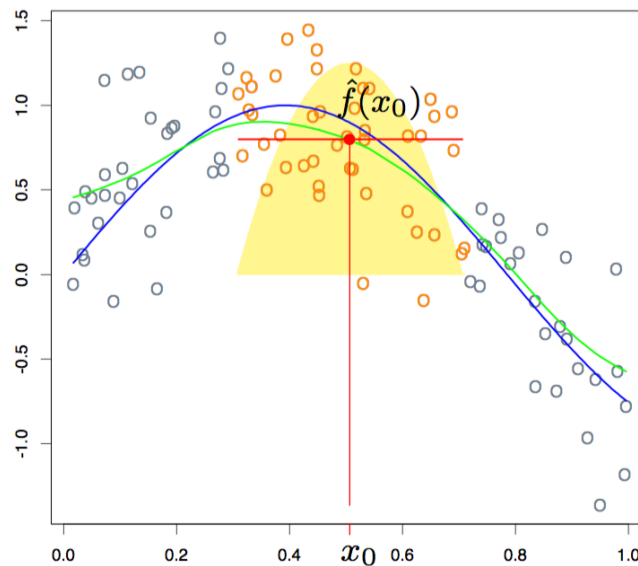
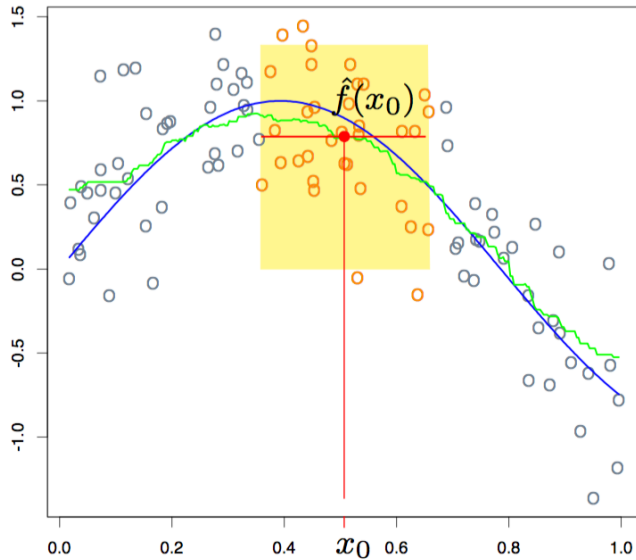
- $k$ -nearest neighbor regressor is

$$\hat{f}(x_0) = \frac{\sum_{i=1}^n y_i \times \text{Ind}(x_i \text{ is a } k \text{ nearest neighbor})}{\sum_{i=1}^n \text{Ind}(x_i \text{ is a } k \text{ nearest neighbor})}$$

$$\hat{f}(x_0) = \frac{\sum_{i=1}^n K(x_0, x_i) y_i}{\sum_{i=1}^n K(x_0, x_i)}$$

# Nearest neighbor regression

$$\{(x_i, y_i)\}_{i=1}^n$$



- $k$ -nearest neighbor regressor is

$$\hat{f}(x_0) = \frac{\sum_{i=1}^n y_i \times \text{Ind}(x_i \text{ is a } k \text{ nearest neighbor})}{\sum_{i=1}^n \text{Ind}(x_i \text{ is a } k \text{ nearest neighbor})}$$

$$\hat{f}(x_0) = \frac{\sum_{i=1}^n K(x_0, x_i) y_i}{\sum_{i=1}^n K(x_0, x_i)}$$

$$\hat{f}(x_0) = b(x_0) + w(x_0)^T x_0$$

$$w(x_0), b(x_0) = \arg \min_{w, b} \sum_{i=1}^n K(x_0, x_i) (y_i - (b + w^T x_i))^2$$

**Local Linear Regression**

# Nearest Neighbor Overview

---

- Very simple to explain and implement
- No training! But finding nearest neighbors in large dataset at test can be computationally demanding (KD-trees help)
- You can use other forms of distance (not just Euclidean)
- Smoothing and local linear regression can improve performance (at the cost of higher variance)
- With a lot of data, “local methods” have strong, simple theoretical guarantees.
- Without a lot of data, neighborhoods aren’t “local” and methods suffer (curse of dimensionality).

# Questions?

---