

Announcements

Coronavirus – COVID-19

- Take care of yourself and others around you
- Follow CMU and government guidelines
- We're "here" to help in any capacity that we can
- Use tools like zoom to communicate with each other too!

Zoom

- Let us know if you have issues
- Etiquette: Turn on video when talking or your turn in OH

Feedback: See Piazza post

Announcements

Assignments

- HW6 (written + programming)
 - Due Thu 3/26, 11:59 pm

“Participation” Points

- Polls open until 10 am (EDT) day after lecture
- “Calamity” option announced in recorded lecture
 - Don’t select this calamity option or you’ll lose credit for one poll (-1) rather than gaining credit for one poll (+1).
- Participation percent calculated as usual

Introduction to Machine Learning

Cross-Validation Nonparametric Regression

Instructor: Pat Virtue

Validation

Why do we need validation?

- Choose hyperparameters
- Choose technique
- Help make any choices beyond our parameters

But now, we have another choice to make!

- How do we split training and validation?

Trade-offs

- More held-out data, better meaning behind validation numbers
- More held-out data, less data to train on!

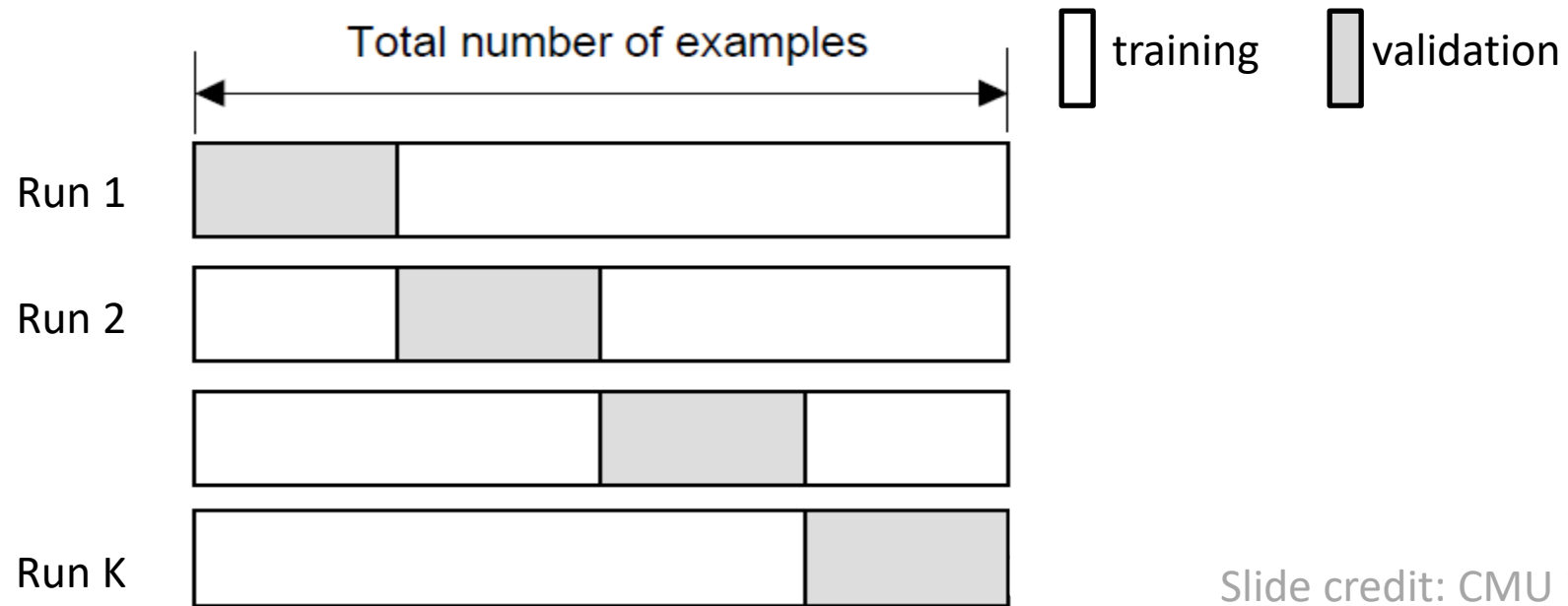
Cross-validation

K-fold cross-validation

Create K-fold partition of the dataset.

Do K runs: train using K-1 partitions and calculate validation error on remaining partition (rotating validation partition on each run).

Report average validation error

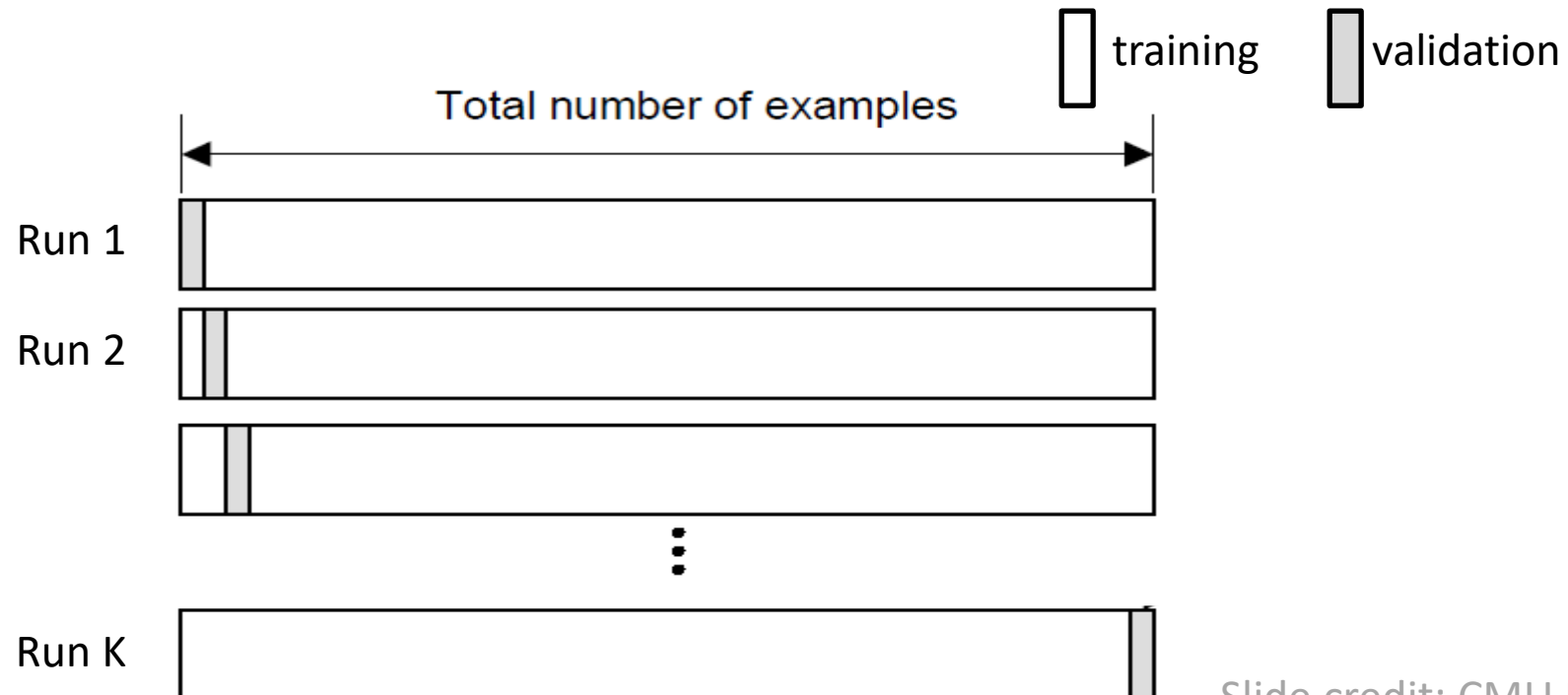


Cross-validation

Leave-one-out (LOO) cross-validation

Special case of K-fold with $K=N$ partitions

Equivalently, train on $N-1$ samples and validate on only one sample per run for N runs



Cross-validation

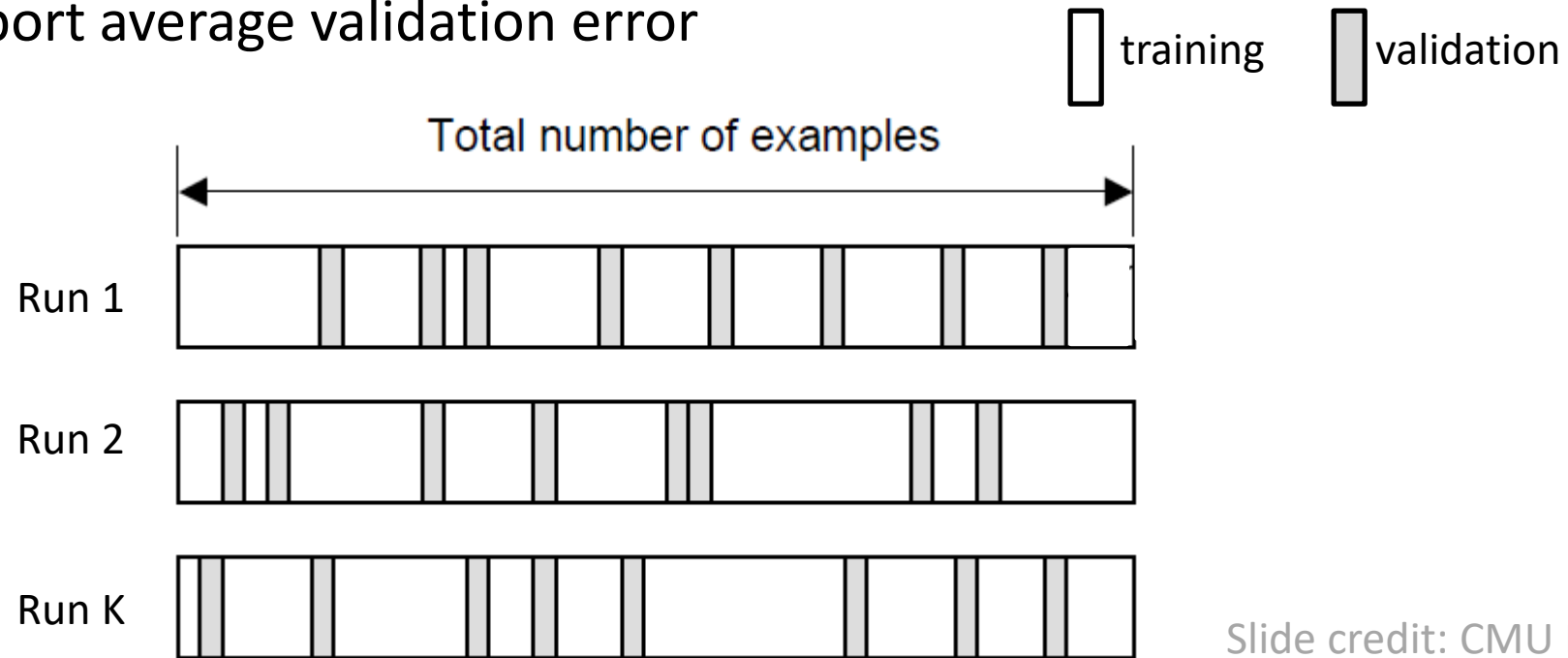
Random subsampling

Randomly subsample a fixed fraction αN ($0 < \alpha < 1$) of the dataset for validation.

Compute validation error with remaining data as training data.

Repeat K times

Report average validation error



Practical Issues in Cross-validation

How to decide the values for K and α ?

- Large K
 - + Validation error can approximate test error well
 - Observed validation error will be unstable (few validation pts)
 - The computational time will be very large as well (many experiments)
- Small K
 - + The # experiments and, therefore, computation time are reduced
 - + Observed validation error will be stable (many validation pts)
 - Validation error cannot approximate test error well

Common choice: $K = 10$, $\alpha = 0.1$ 😊

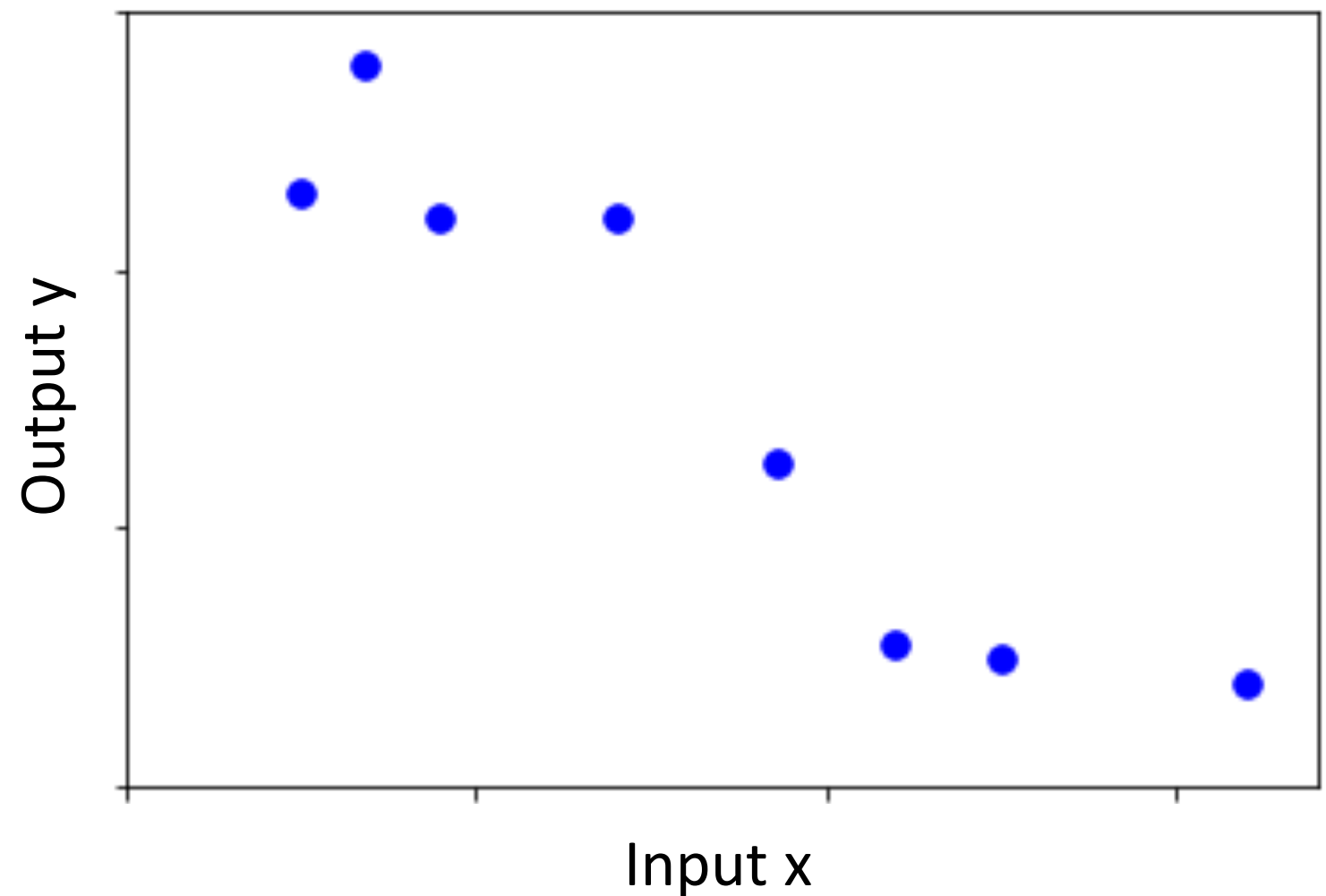
Piazza Poll 1

Say you are choosing amongst 10 values of lambda, and you want to do K=10 fold cross-validation.

How many times do I have to train my model?

- A. 0
- B. 1
- C. 10
- D. 20
- E. 100
- F. 10^{10}

Nonparametric Regression



Reminder: Parametric models

Assume some model (Gaussian, Bernoulli, Multinomial, logistic, network of logistic units, Linear, Quadratic) with fixed number of parameters

- Linear/Logistic Regression, Naïve Bayes, Discriminant Analysis, Neural Networks

Estimate parameters $(\mu, \sigma^2, \theta, w, \beta)$ using MLE/MAP and plug in

Pro – need fewer data points to learn parameters

Con – Strong distributional assumptions, not satisfied in practice

Reminder: Nonparametric models

Nonparametric: number of parameters scales with number of training data

- Typically don't make any distributional assumptions
- As we have more data, we should be able to learn more complex models

Example

- Nearest Neighbor (k-Nearest Neighbor) Classifier

Piazza Poll 2

Are decision trees parametric or non-parametric?

A.

B.

C.

Piazza Poll 2

Are decision trees parametric or non-parametric?

It depends :)

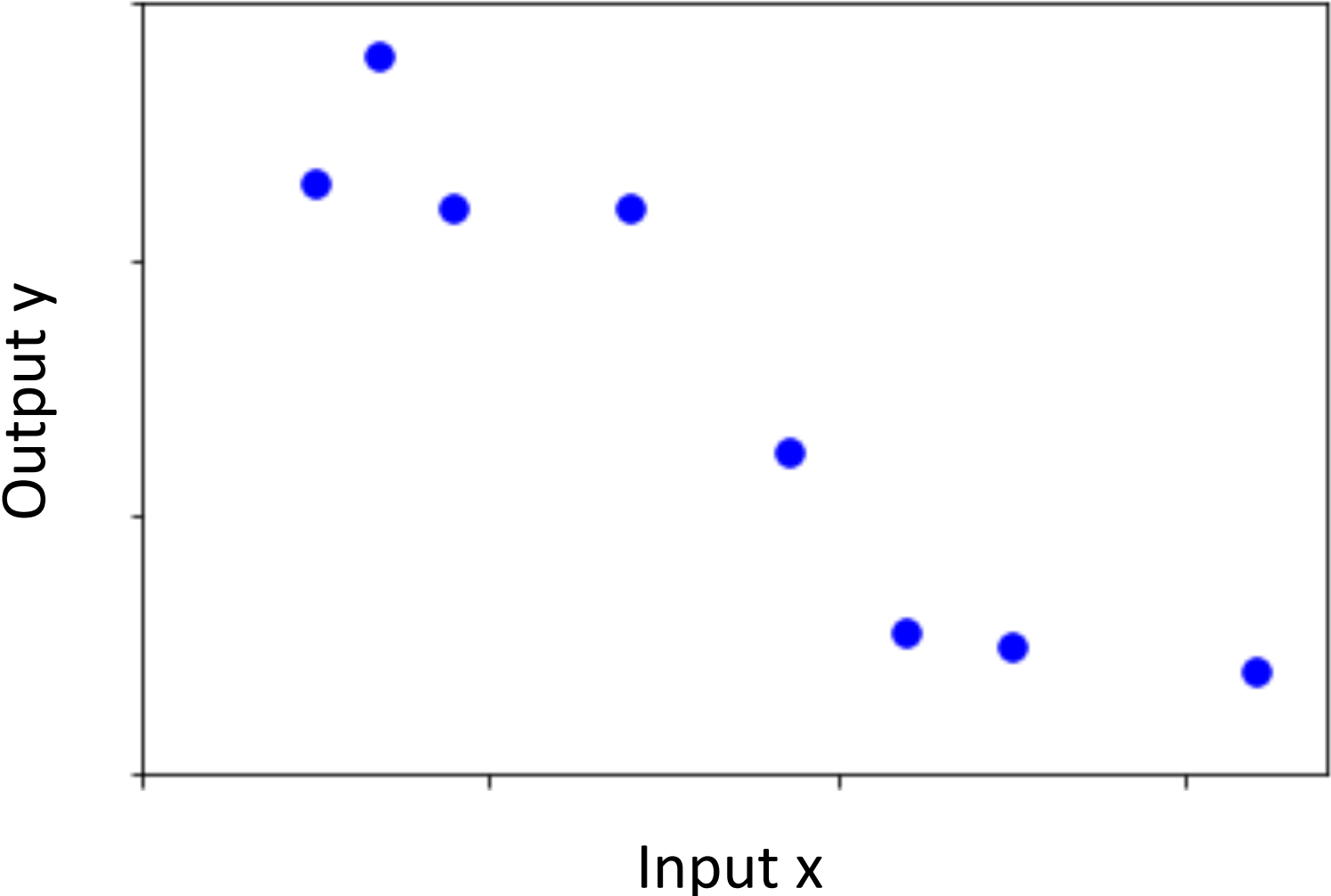
- If no limits on depth or reuse of attributes, then non-parametric
 - Model complexity will grow with data
- If pruned/limited to fix size
 - Parametric
- If attributes only used once
 - Parametric; model complexity is limited by number of features

Trade-offs

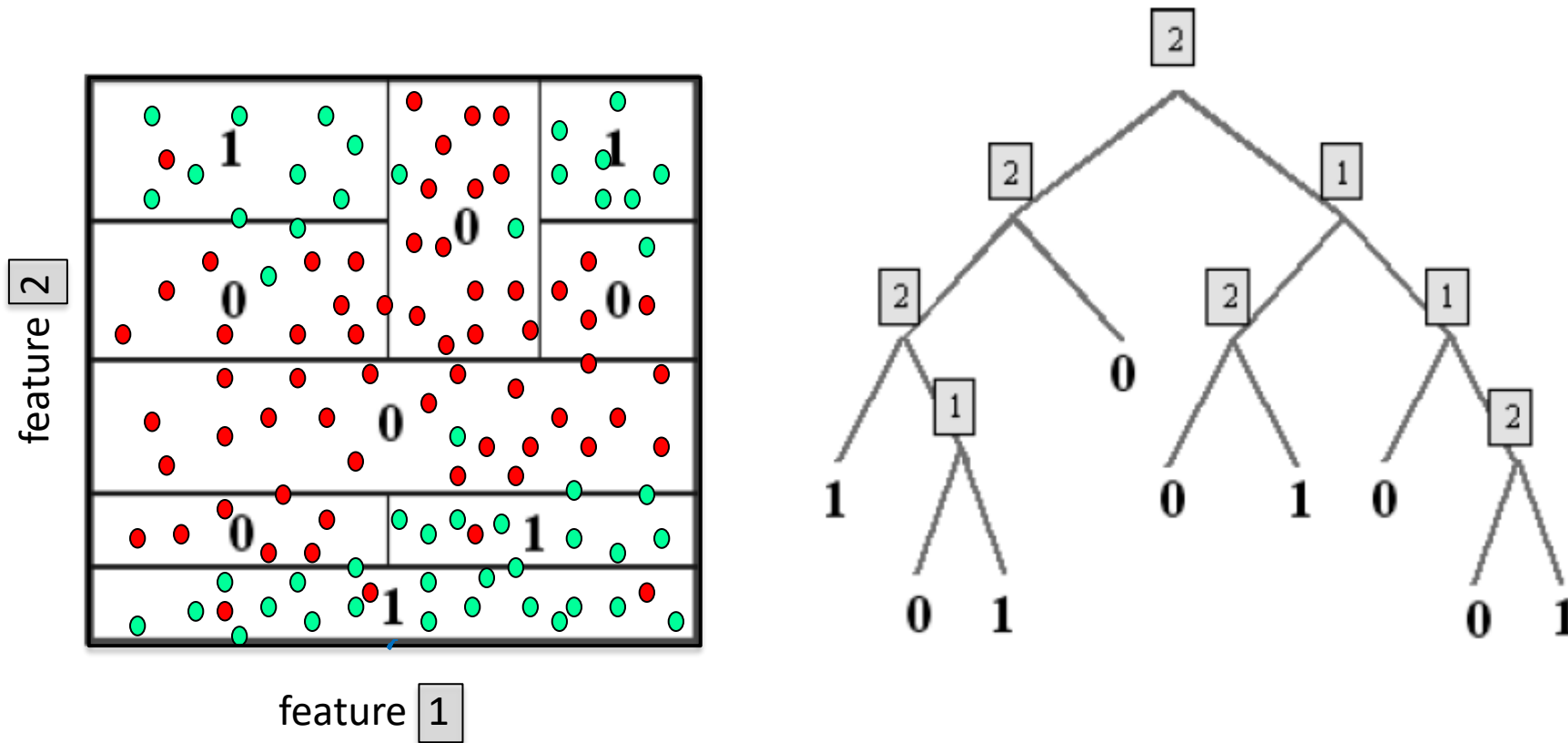
- Non-parametric methods have very powerful representation capabilities
- But
 - Easily overfit
 - Can take up memory proportional to training size too

Nonparametric Regression

Decision Trees

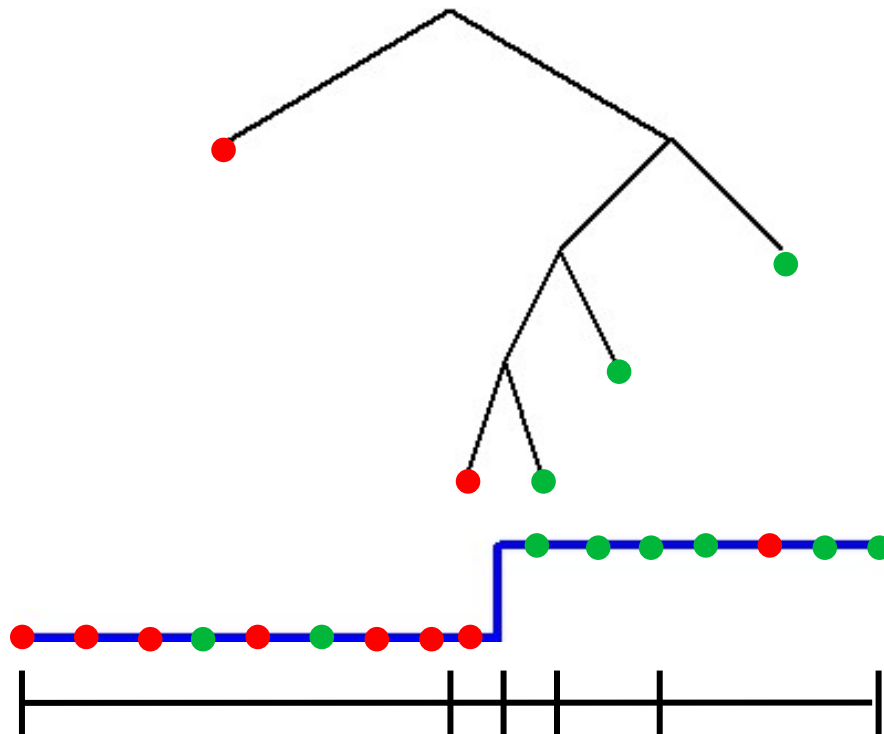


Dyadic decision trees (split on mid-points of features)

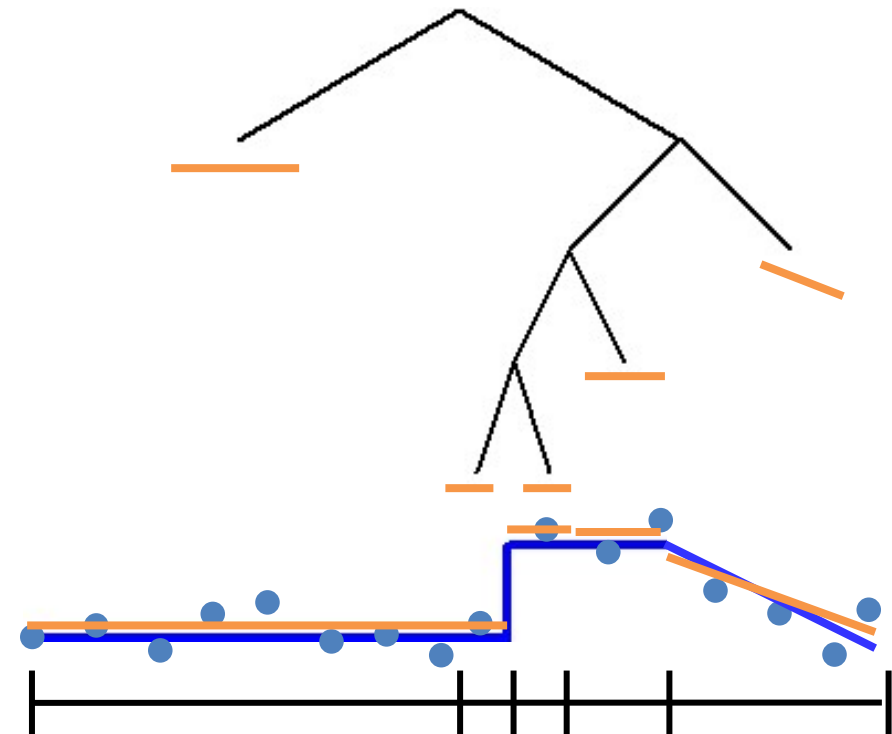


How to assign label to each leaf

Classification – Majority vote

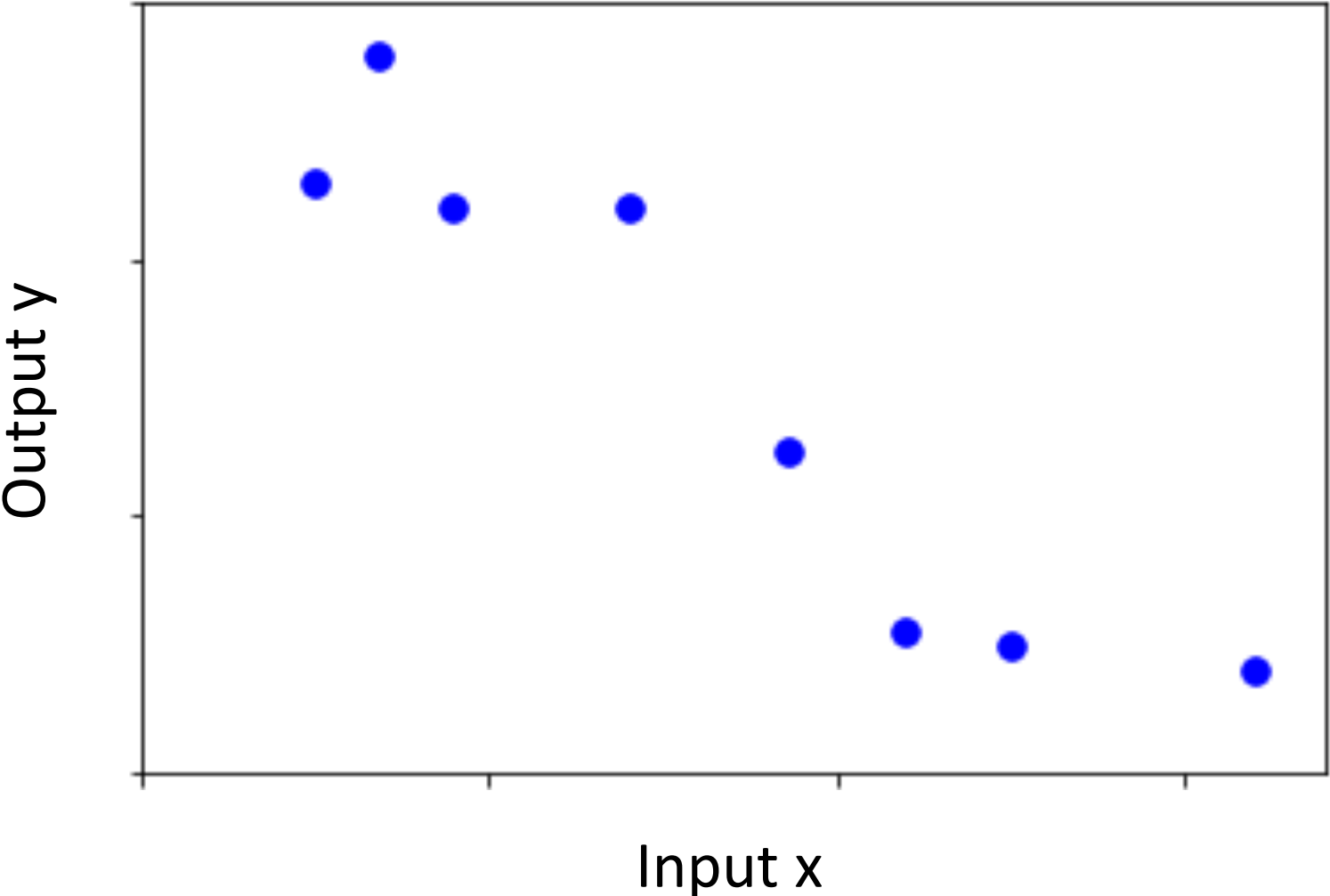


Regression – Constant/Linear/Poly fit



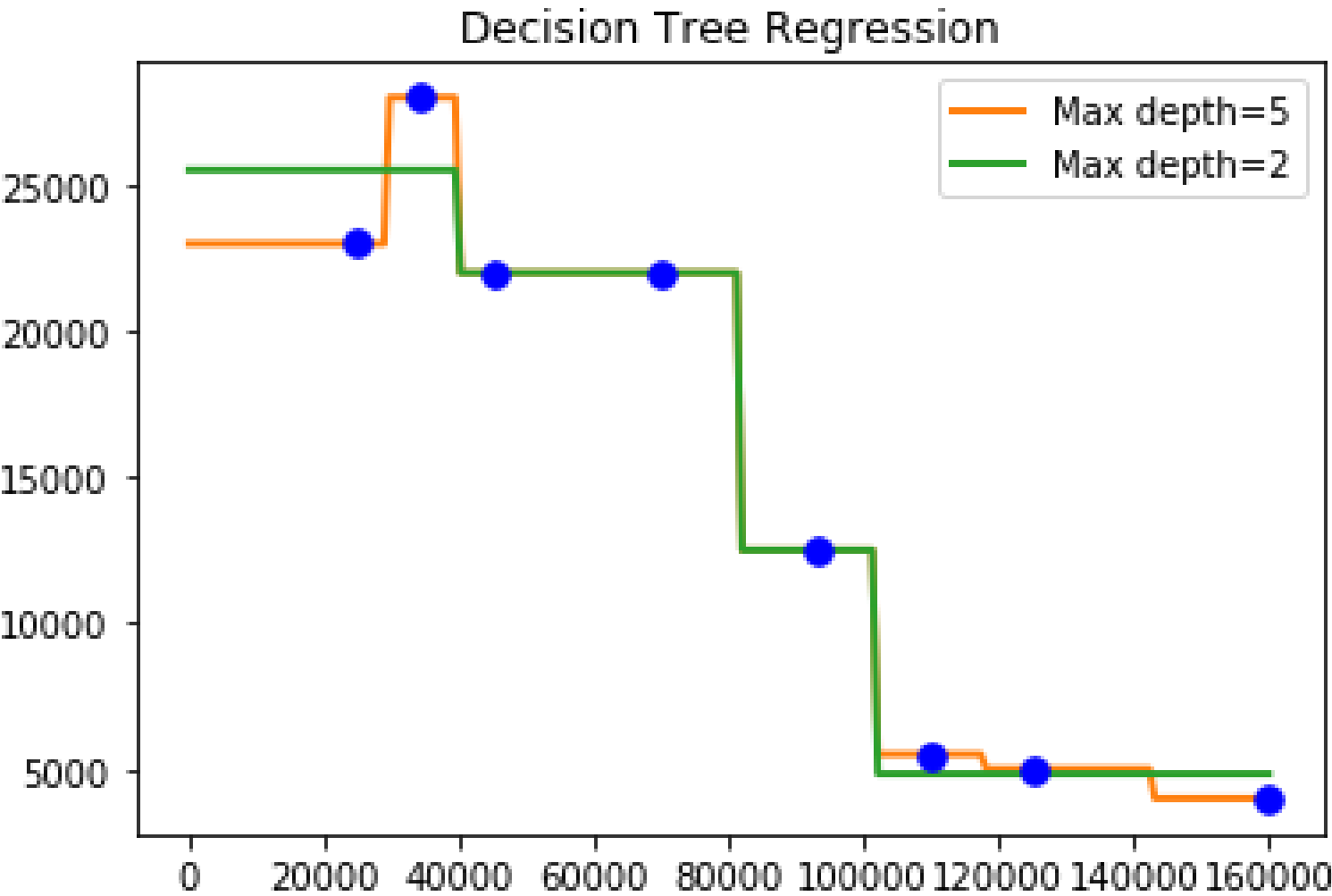
Nonparametric Regression

Decision Trees



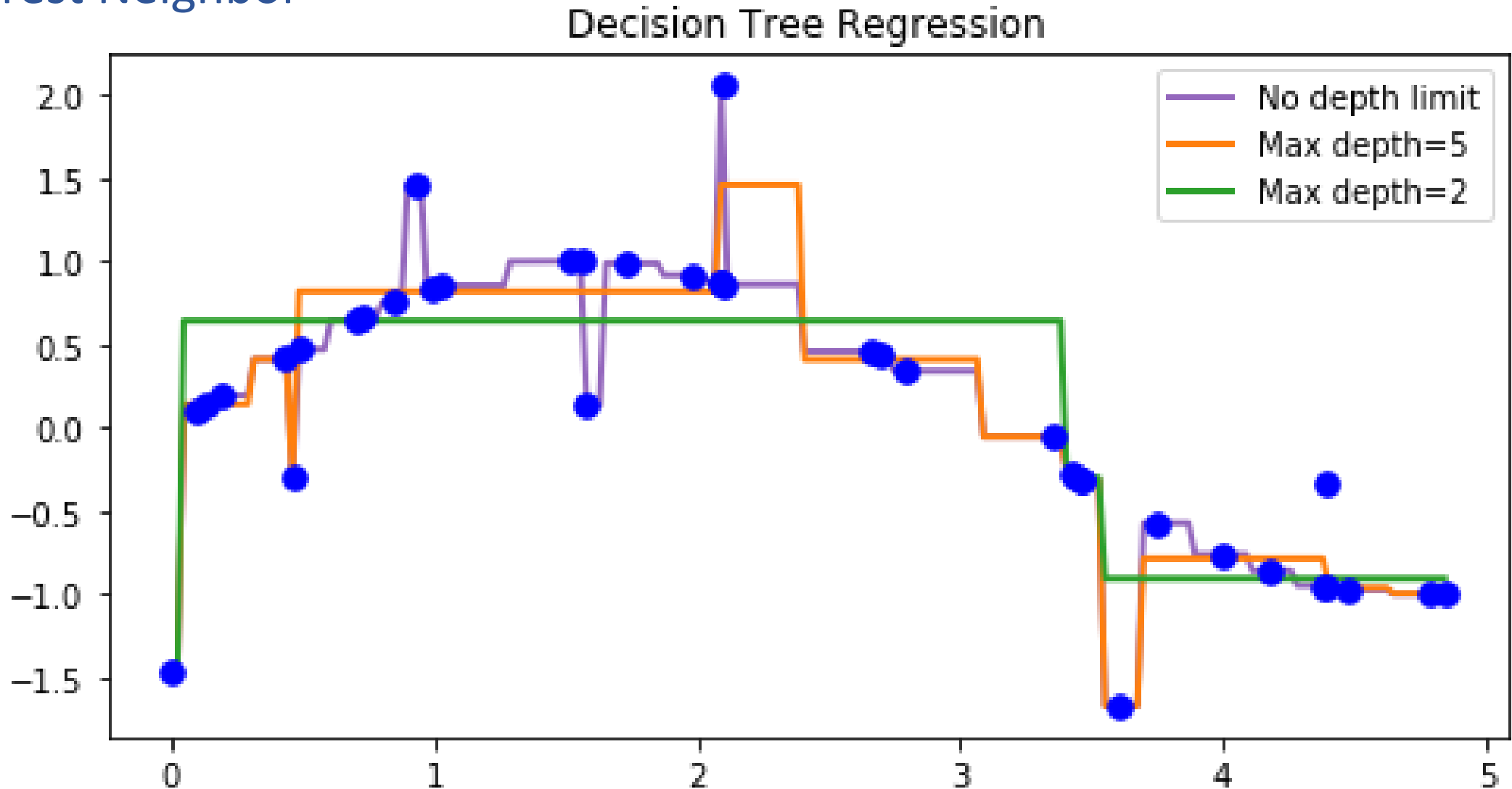
Nonparametric Regression

Nearest Neighbor



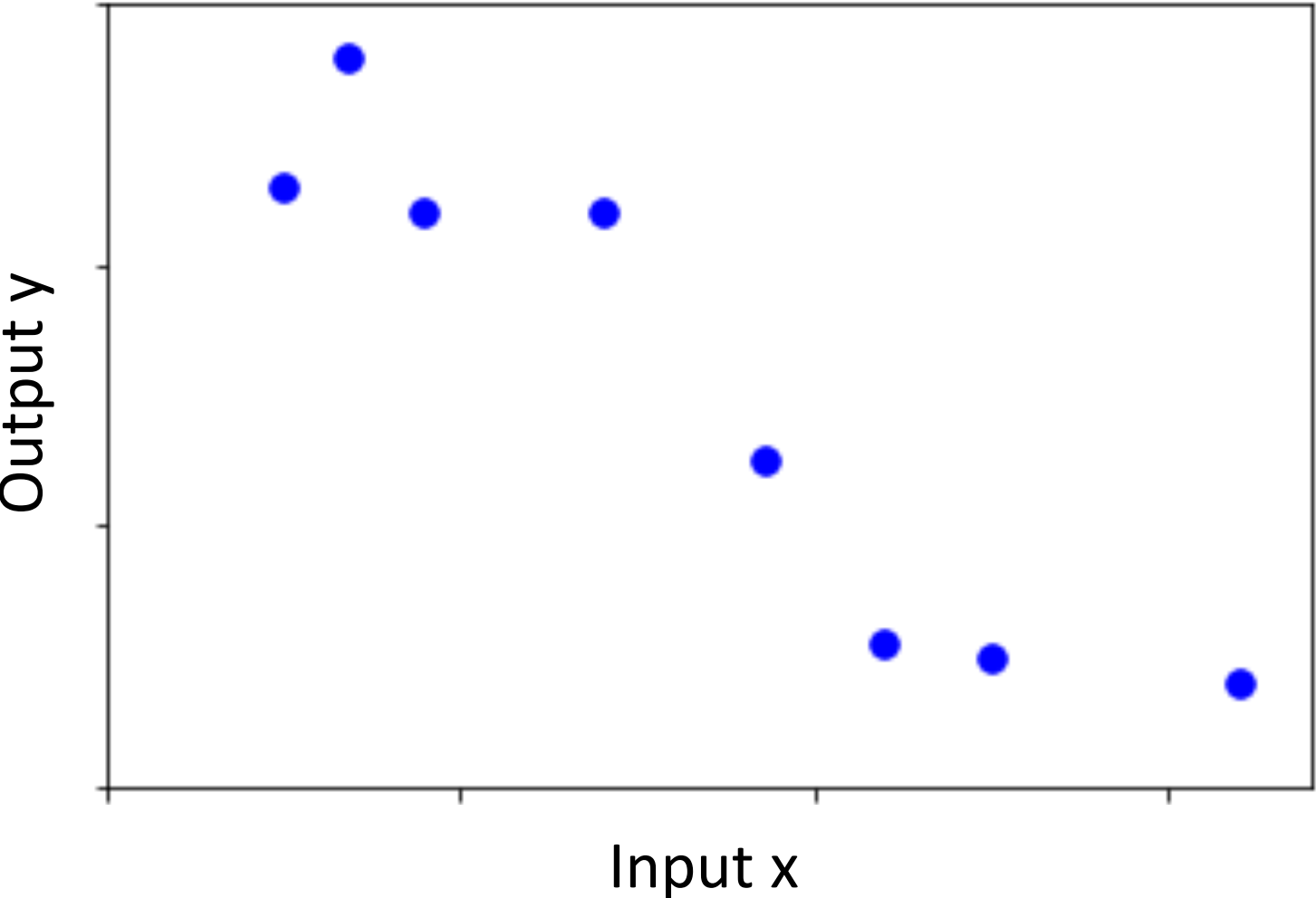
Nonparametric Regression

Nearest Neighbor



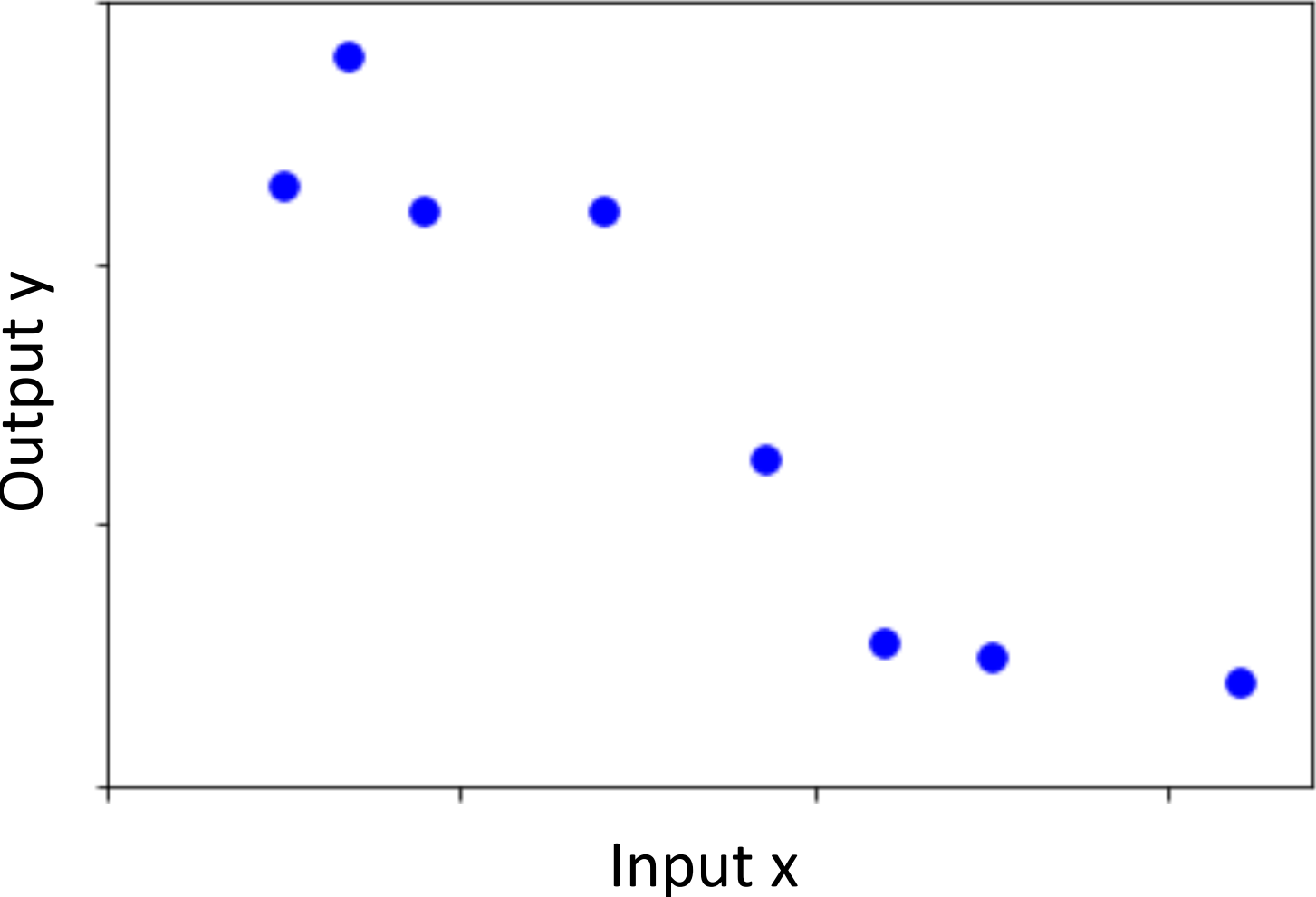
Nonparametric Regression

Nearest Neighbor



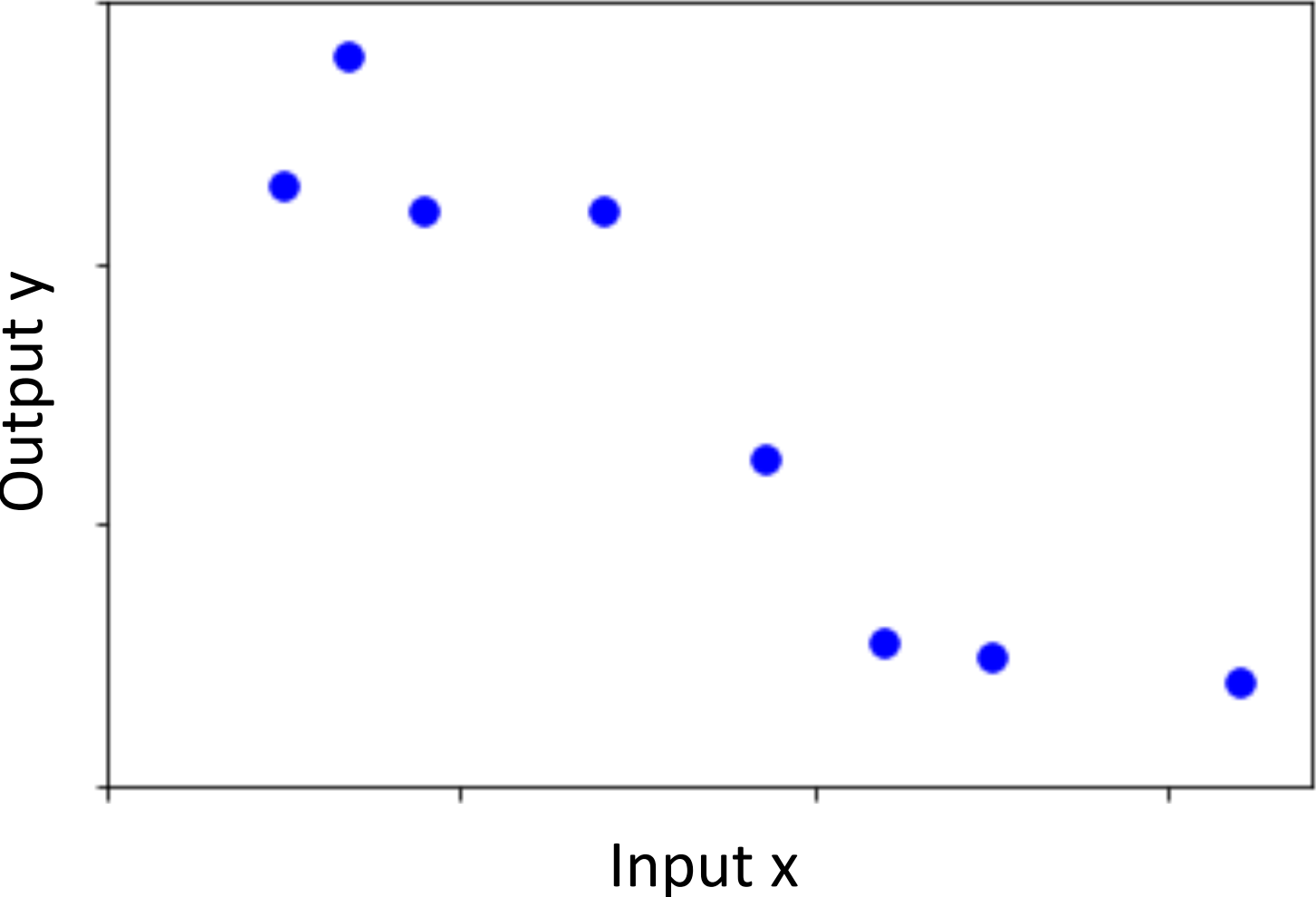
Nonparametric Regression

Kernel Regression

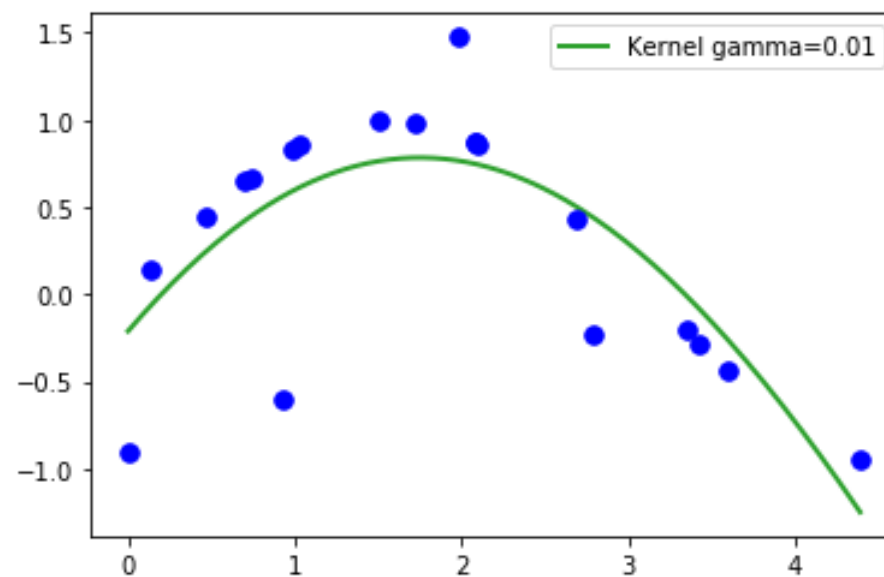
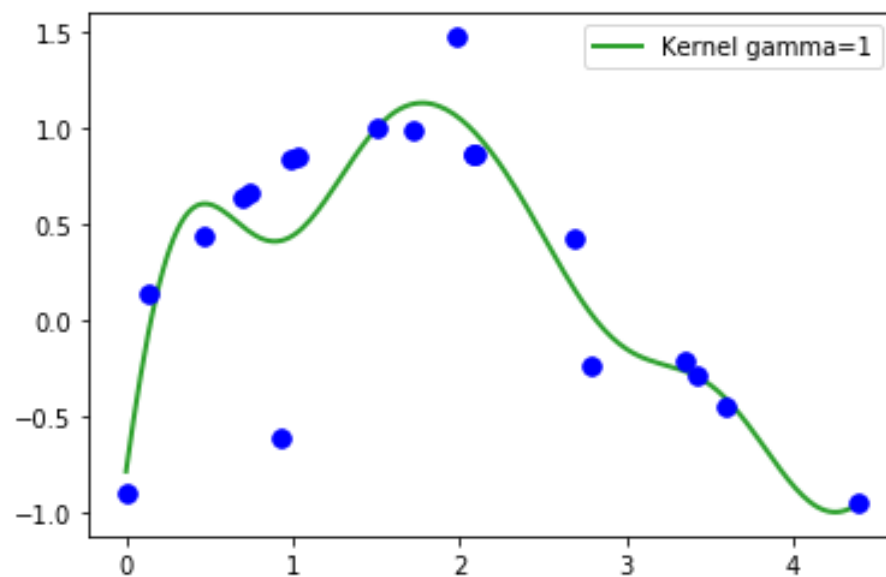
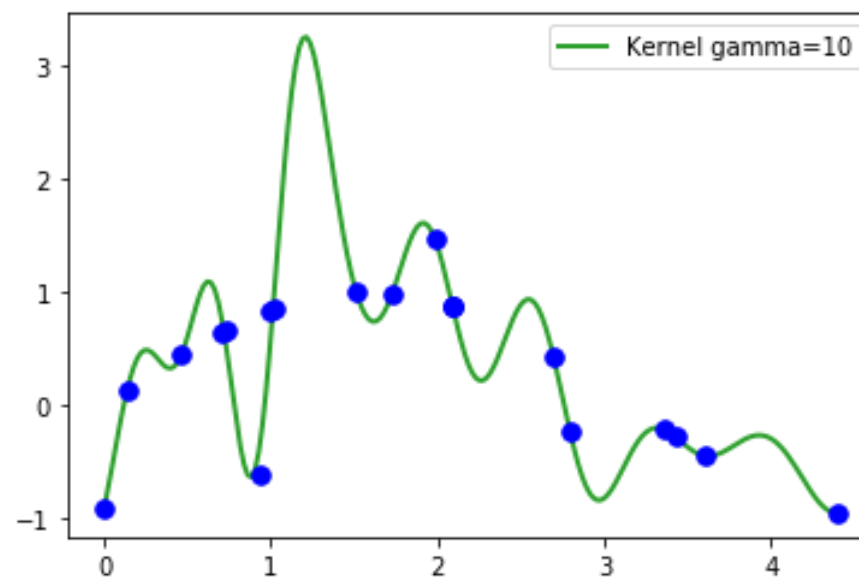
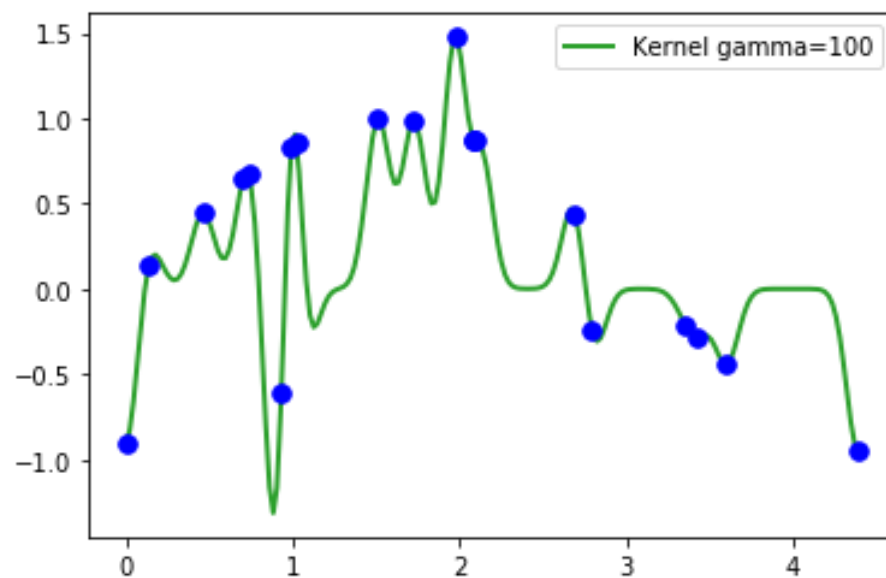


Nonparametric Regression

Kernel Regression



Kernel Regression



Kernel Regression

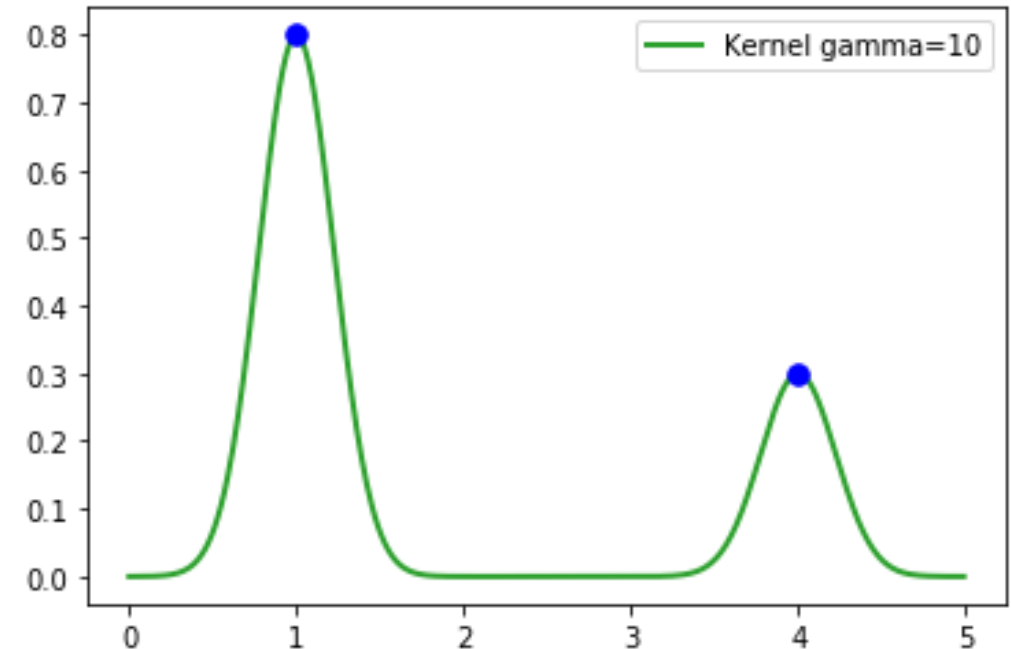
RBF kernel and corresponding hypothesis function

Distance kernel (Gaussian / Radial Basis Function)

- Close to point should be that point
- Far should be zero
- Mini Gaussian window

$$k(x, x') = e^{\frac{-\|x-x'\|_2^2}{2\sigma^2}} = e^{-\gamma\|x-x'\|_2^2}$$

- We control the variance

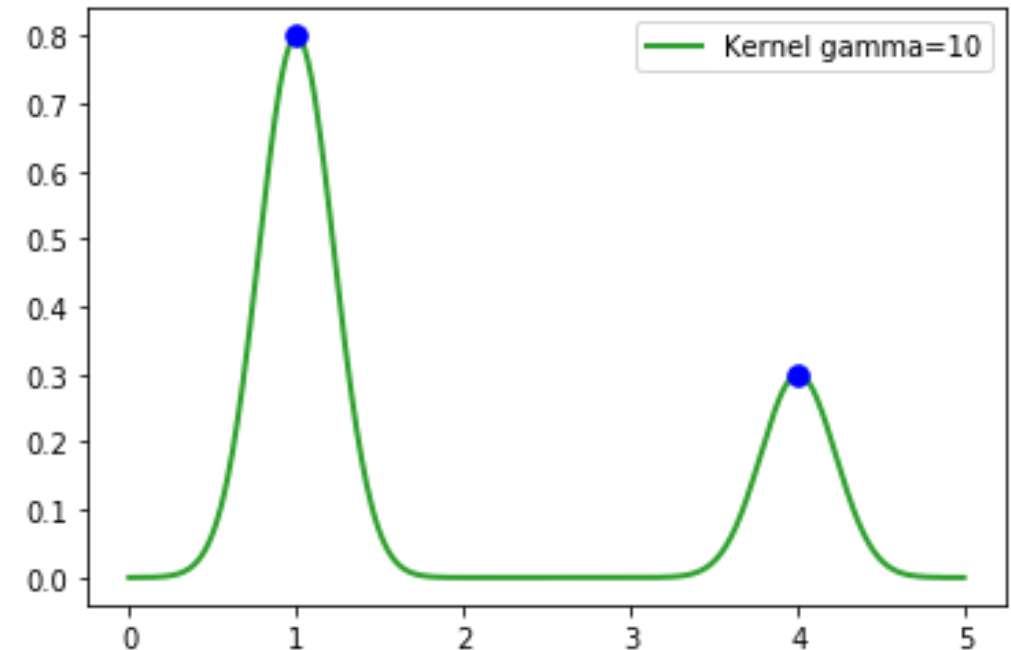


Kernel Regression

RBF kernel and corresponding hypothesis function

Prediction?

- Weighted sum of these little windows
 - $\hat{y} = h(x) = \sum_i \alpha_i k(x, x^{(i)})$
 - What should α_i be?
 - $\alpha_i = y_i, \alpha = y?$
 - Need to account for points that are close together

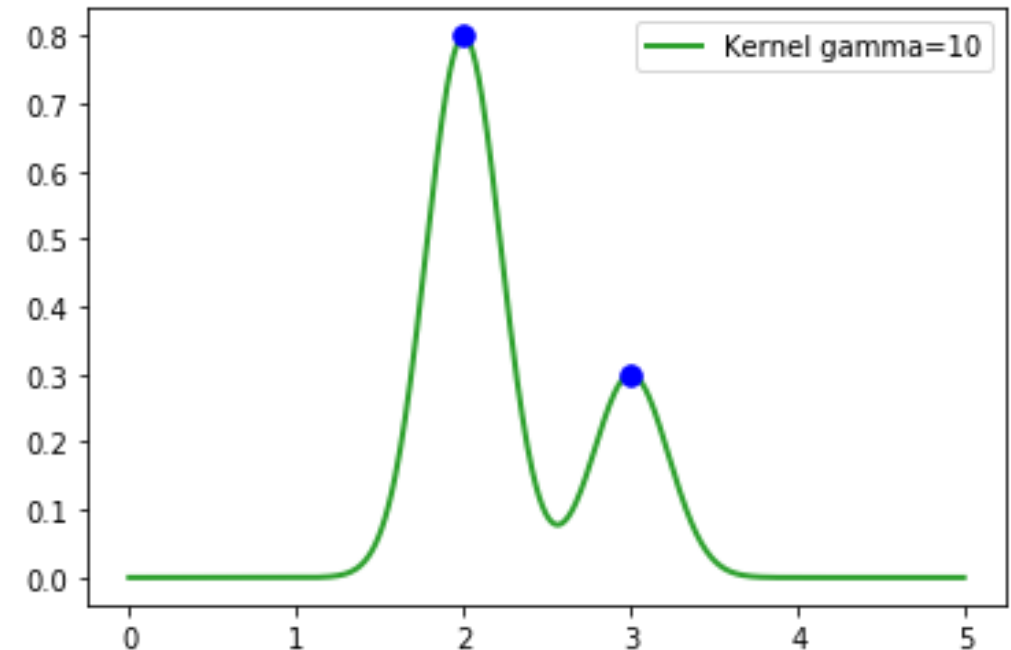


Kernel Regression

RBF kernel and corresponding hypothesis function

Prediction?

- Weighted sum of these little windows
 - $\hat{y} = h(x) = \sum_i \alpha_i k(x, x^{(i)})$
 - What should α_i be?
 - $\alpha_i = y_i, \alpha = y?$
 - Need to account for points that are close together

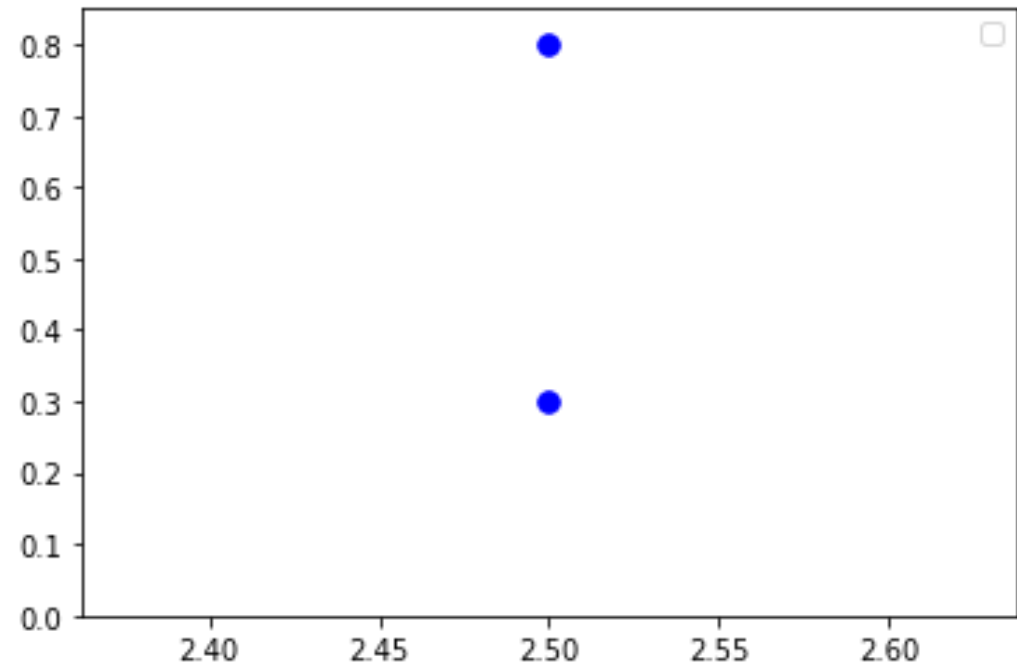


Kernel Regression

RBF kernel and corresponding hypothesis function

Prediction?

- Weighted sum of these little windows
 - $\hat{y} = h(x) = \sum_i \alpha_i k(x, x^{(i)})$
 - What should α_i be?
 - $\alpha_i = y_i, \alpha = y?$
 - Need to account for points that are close together

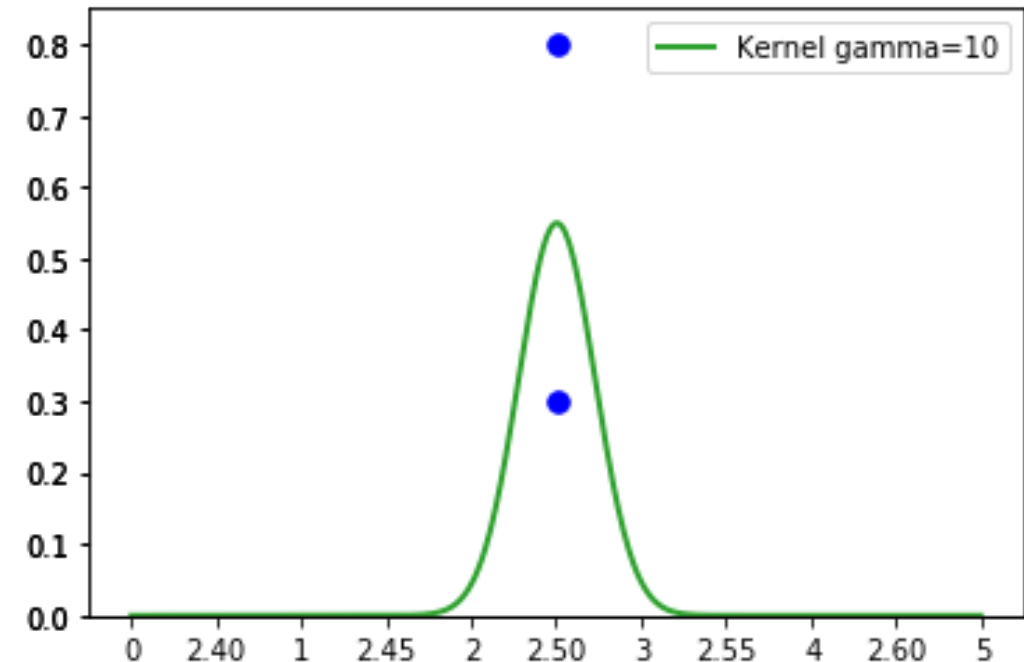


Kernel Regression

RBF kernel and corresponding hypothesis function

Prediction?

- Weighted sum of these little windows
 - $\hat{y} = h(x) = \sum_i \alpha_i k(x, x^{(i)})$
 - What should α_i be?
 - Need to account for points that are close together

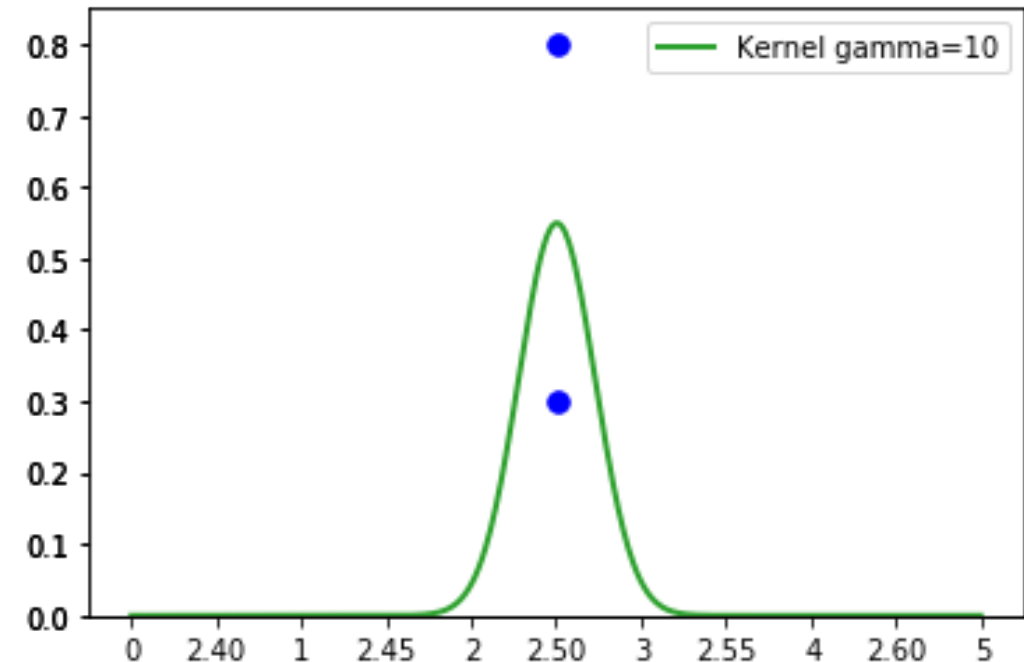


Kernel Regression

RBF kernel and corresponding hypothesis function

Prediction?

- Weighted sum of these little windows
 - $\hat{y} = h(x) = \sum_i \alpha_i k(x, x^{(i)})$
 - What should α_i be?
 - Need to account for points that are close together
- $$\alpha = (K)^{-1}y$$
- $$\alpha = (K + \lambda I)^{-1}y$$
- where $K_{ij} = k(x^{(i)}, x^{(j)})$
- and λ is small to help inversion



Kernelized Linear Regression

Reminder: Polynomial Linear Regression

Polynomial feature function

Least squares formulation

Least squares solution

Reminder: Polynomial Linear Regression

Polynomial feature function

- $x \rightarrow \phi(x) = [1, x, x^2, x^3]^T$
- $X \rightarrow \Phi$

Least squares formulation

- $\min_w \|y - \Phi w\|_2^2$

Least squares solution

- $w = (\Phi^T \Phi)^{-1} \Phi^T y$

Plus L2 regularization

- $\min_w \|y - \Phi w\|_2^2 + \lambda \|w\|_2^2$
- $w = (\Phi^T \Phi + \lambda I)^{-1} \Phi^T y$

Can rewrite as

- $w = \Phi^T (\Phi \Phi^T + \lambda I)^{-1} y$

Kernelized Linear Regression

L2 regularized linear regression (with feature function)

- $\min_w \|y - \Phi w\|_2^2 + \lambda \|w\|_2^2$
- $w = (\Phi^T \Phi + \lambda I)^{-1} \Phi^T y$

Can rewrite as

- $w = \Phi^T (\Phi \Phi^T + \lambda I)^{-1} y$

Let

$$\alpha = (\Phi \Phi^T + \lambda I)^{-1} y$$

Prediction

- $\hat{y} = h(x) = w^T x$