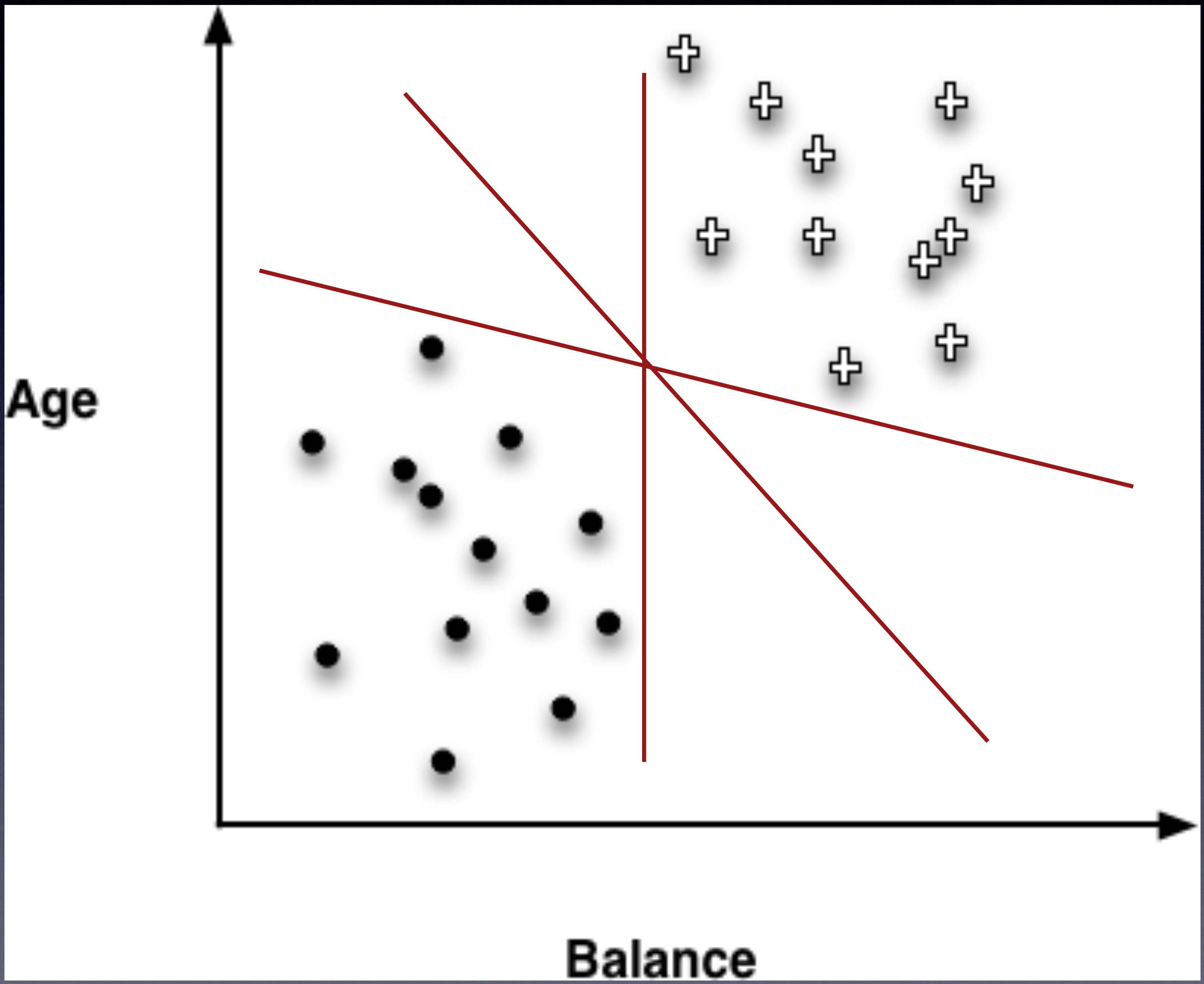
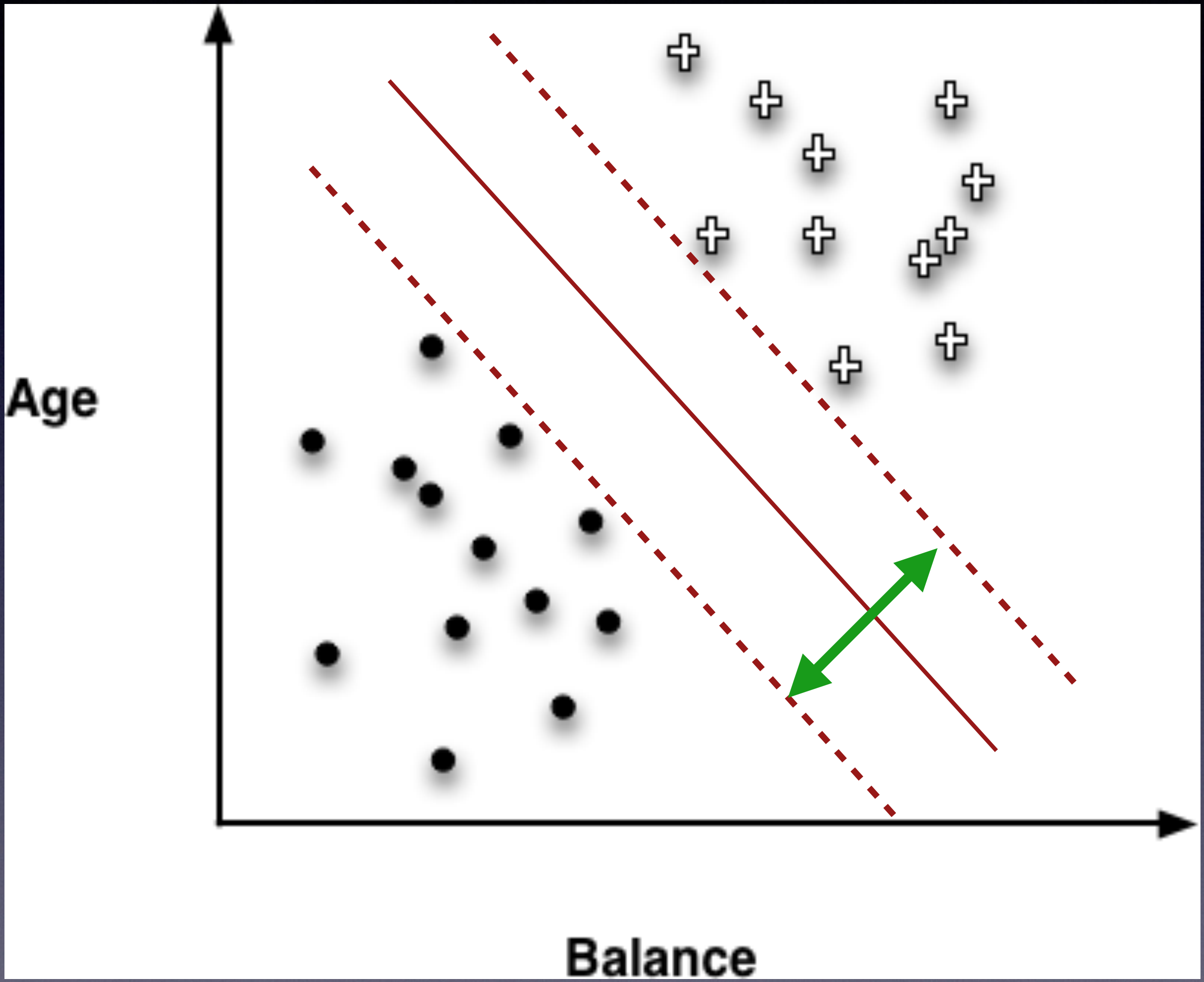
Support Vector Machines (SVMs), Linear regression, and Logistic regression

Possible Linear Classifiers



Split the data **in the best possible way**

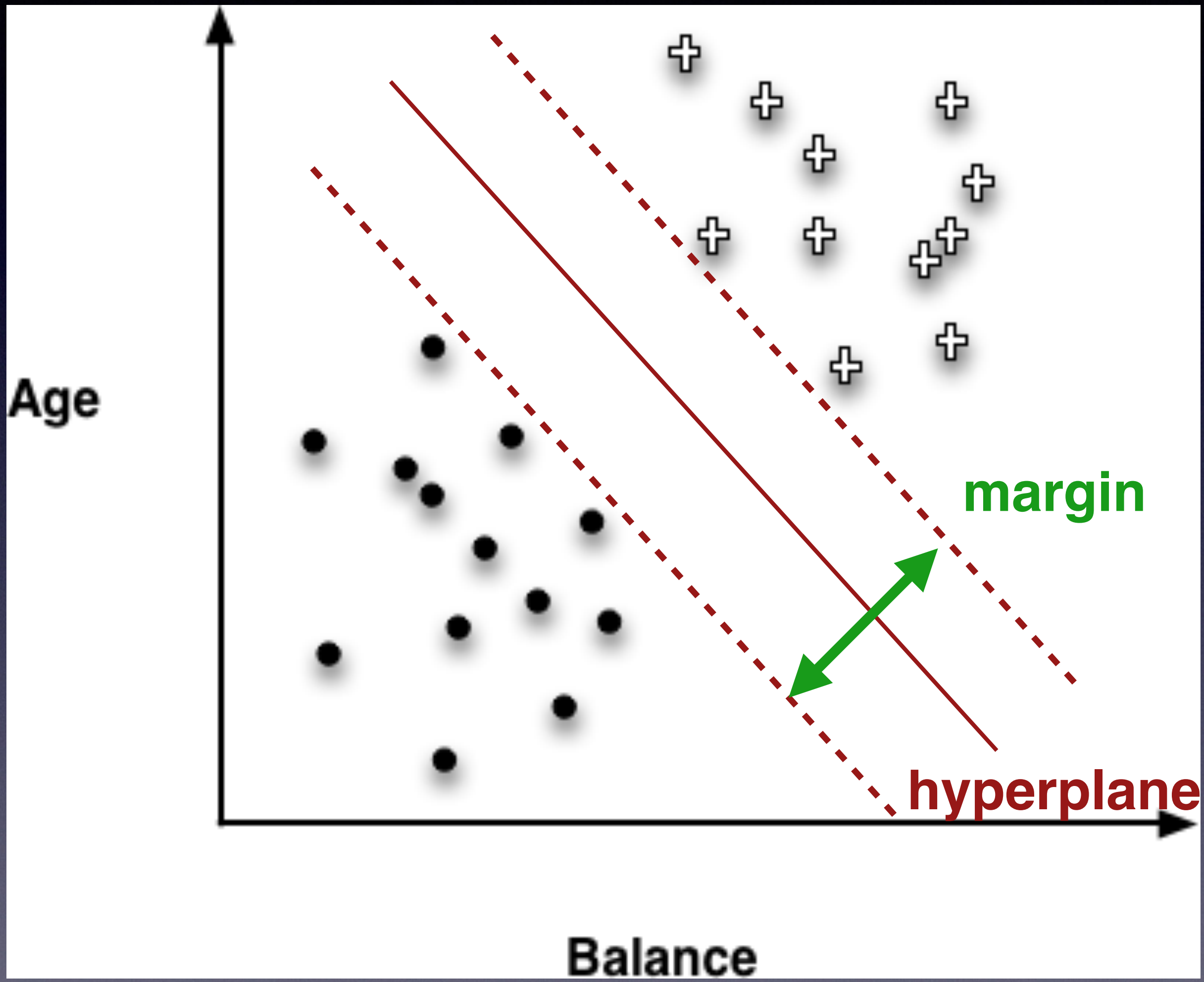
Support Vector Machine (SVM)



This is the widest road that separates the two groups

largest margin

Support Vector Machine (SVM)



The distance between the **support vectors** are as far as possible.

Let’s apply this SVM to a real world problem

Cupcakes vs Muffins

- The problem
  - Classify recipes as cupcakes or muffins
  - When given a new recipe, determine if it’s a cupcake or a muffin
- The Steps
  - Get the recipe data
  - Fit a data science/classification model to the data
  - Take the new recipe and predict the result

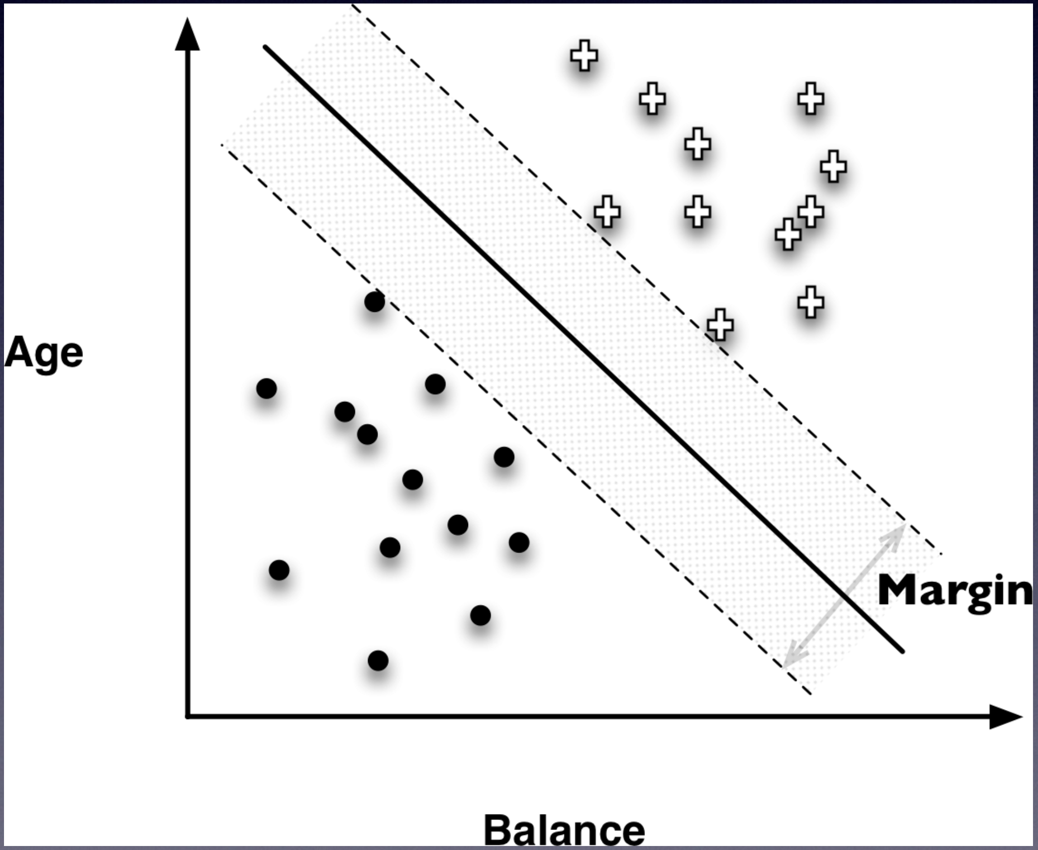
Jupter Notebook Demo

Fitting a Model to Data

- Finding “optimal” model and its parameters based on data
- Objective functions
- Loss function

Support Vector Machines (SVMs)

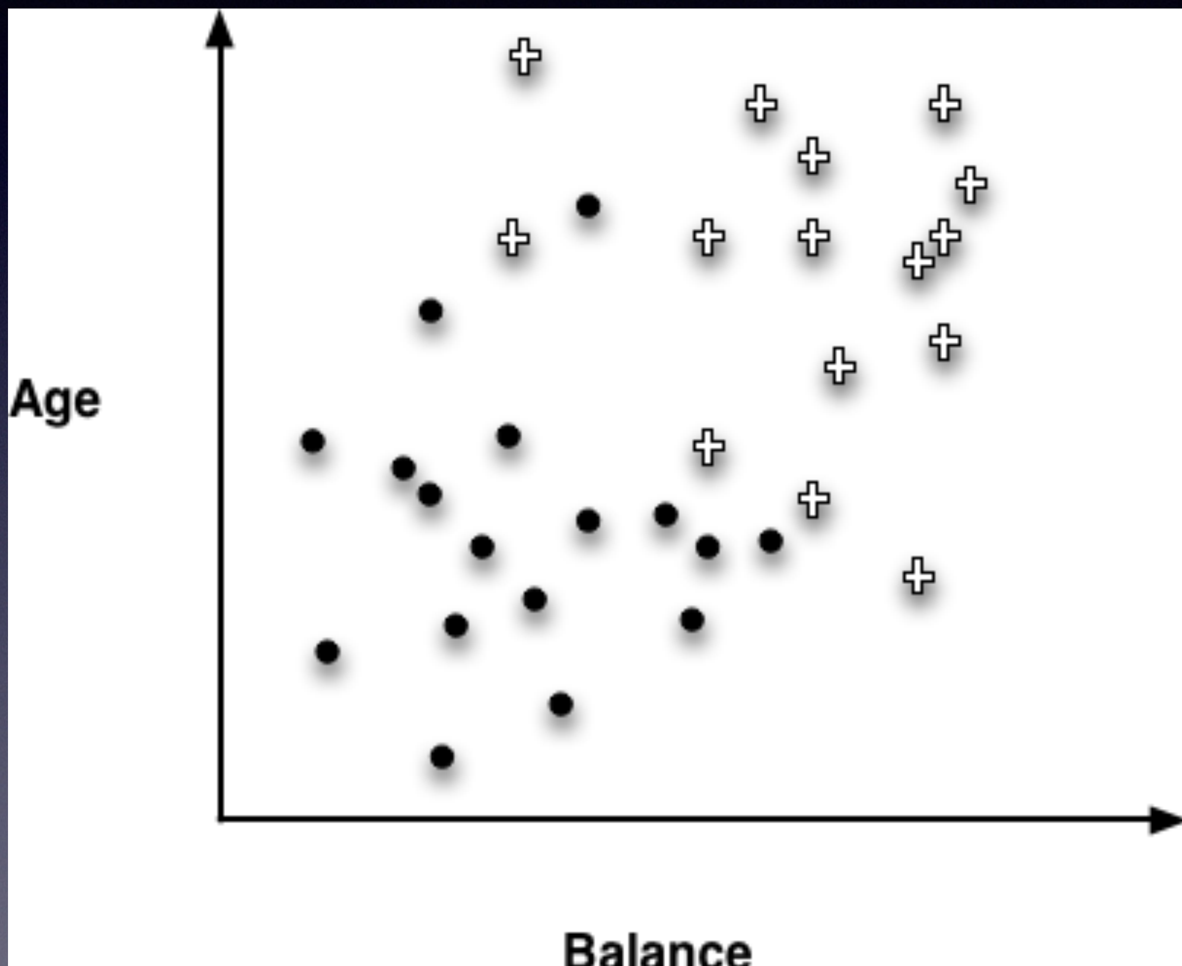
- SVMs are linear discriminant functions
  - Classify instances based on a linear function of the features
- Objective: maximize the margin



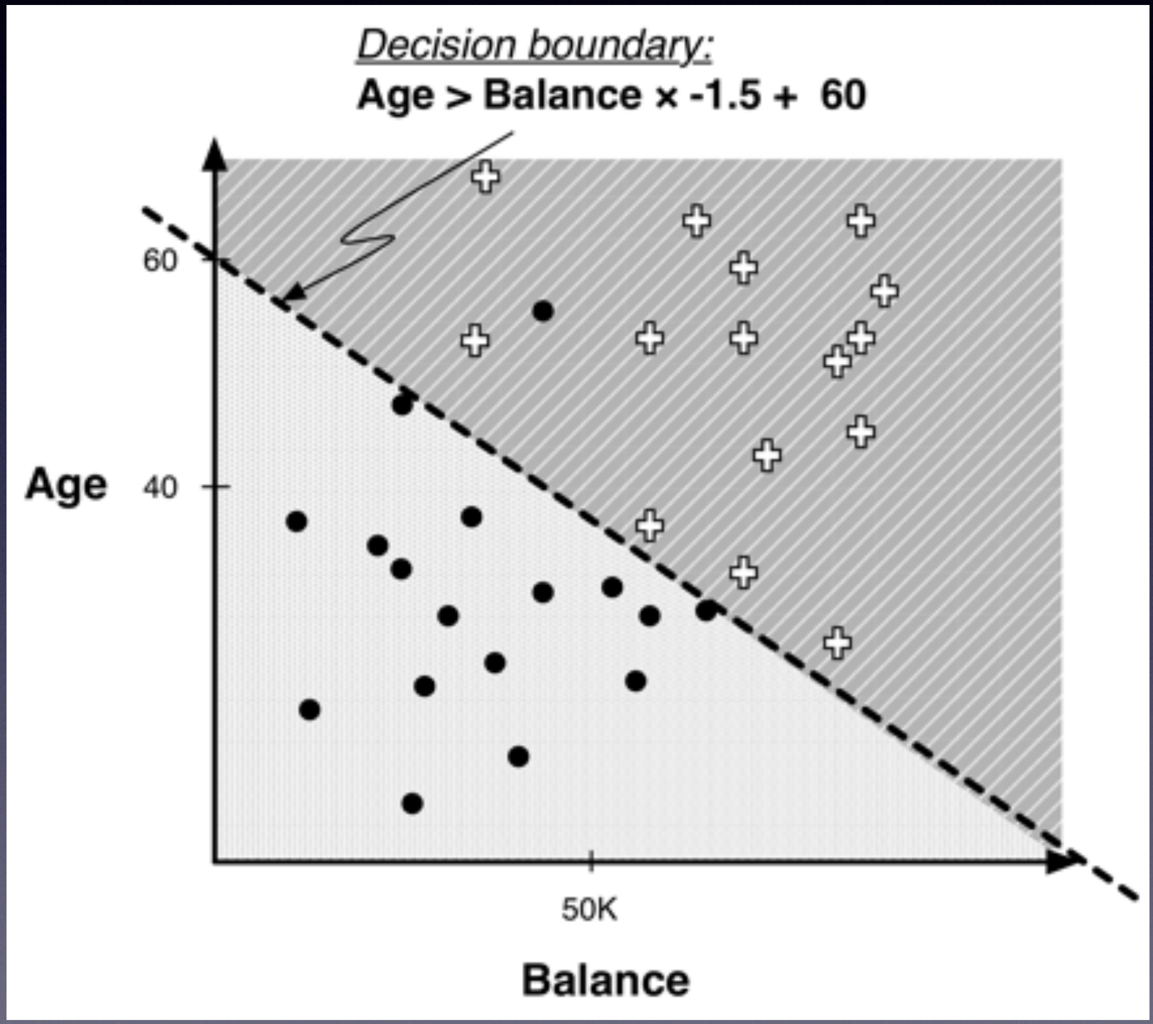
Fit a Model to Data (Decision Tree)

A dataset split by a classification tree with four leaf nodes

Other Ways to Partition?



Linear Classifier



Linear Discriminant Functions

- The equation of a line in two dimensions is

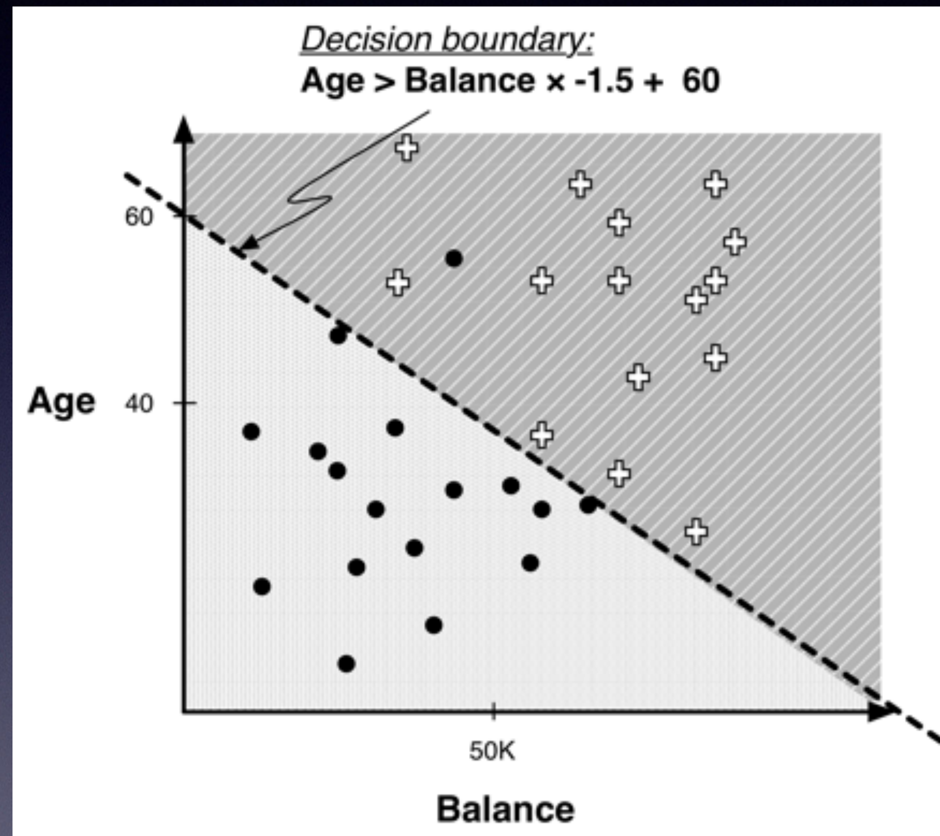
$$y = mx + b$$

- The line in the figure:

$$Age = (-1.5) \times Balance + 60$$

- Linear classification function

$$class(\mathbf{x}) = \begin{cases} + \text{ if } 1.0 \times Age - 1.5 \times Balance + 60 > 0 \\ \bullet \text{ if } 1.0 \times Age - 1.5 \times Balance + 60 \leq 0 \end{cases}$$



- Linear discriminant function:
  - Discriminates between the classes
  - The decision boundary is a linear combination of the features

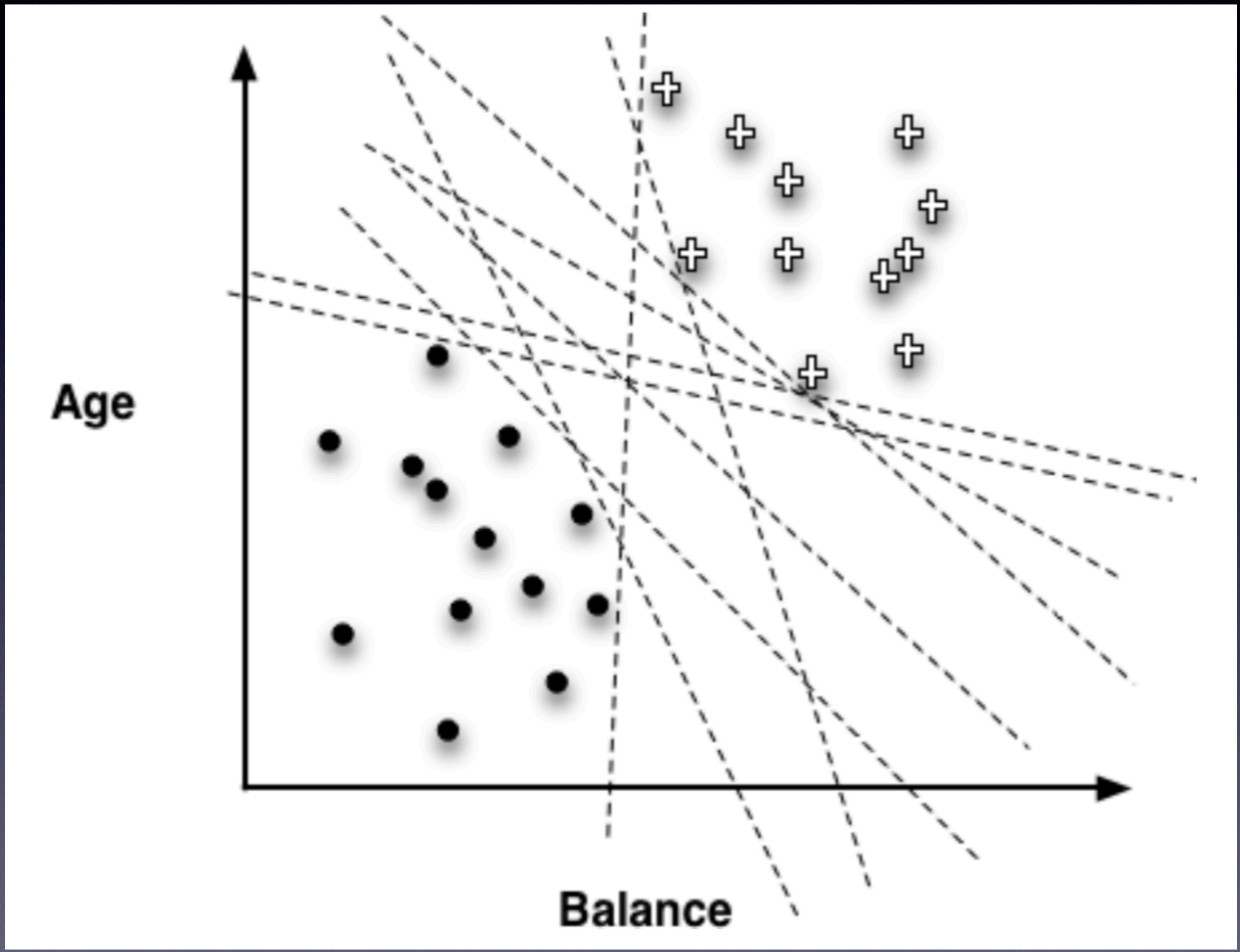
Linear Model

- A general linear regression model

$$\mathbf{y} = f(\mathbf{w}, \mathbf{x}) = w_0 + w_1 x_1 + w_2 x_2 + \ldots$$

- $\mathbf{y}$: label vector, with each individual label being y_i
- $\mathbf{x}$: feature vector, with each individual feature being x_i
- w_i : model parameters
- Goal: fit the parameterized model f(x) to a particular dataset with features $\mathbf{x}$ and labels $\mathbf{y}$, i.e., find a good set of weights w_i on the features $\mathbf{x}$ and labels $\mathbf{y}$

Possible Linear Classifiers



Many different possible linear boundaries can separate the two groups of points. How to pick the best one?

How to Find the Best Weights

- Define an objective function that presents our analysis goal
- Find the the optimal value for the weights by maximizing or minimizing the objective function
- Common and basic techniques to fit a (linear) model to data (each uses a different objective function)
  - Linear regression
  - Support Vector Machine (SVM)
  - Logistic regression

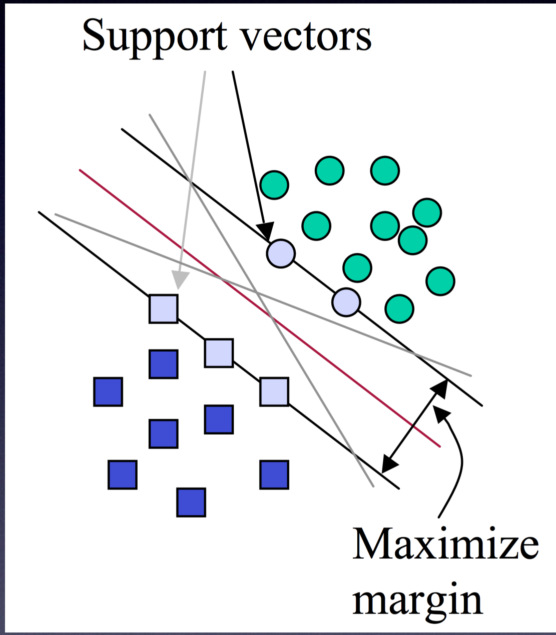
SVM and Its Objective Function

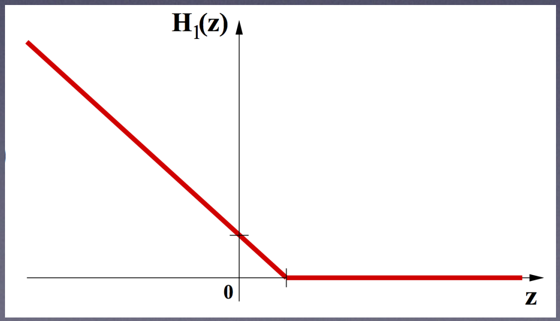
$$\mathbf{y} = f(\mathbf{w}, \mathbf{x}) = w_0 + w_1 x_1 + w_2 x_2 + \ldots$$

- **Input**: set of (features, labels) training pair samples (x_i, y_i)
- **Output**: set of weights w_i
- **Objective**: Find $\mathbf{w}$ to

$$\underbrace{\frac{1}{2}\|\boldsymbol{w}\|^2}_{\text{maximize margin}} + \underbrace{C \sum_i H_1[\, y_i\, f(\boldsymbol{x}_i)\,]}_{\text{minimize training error}}$$

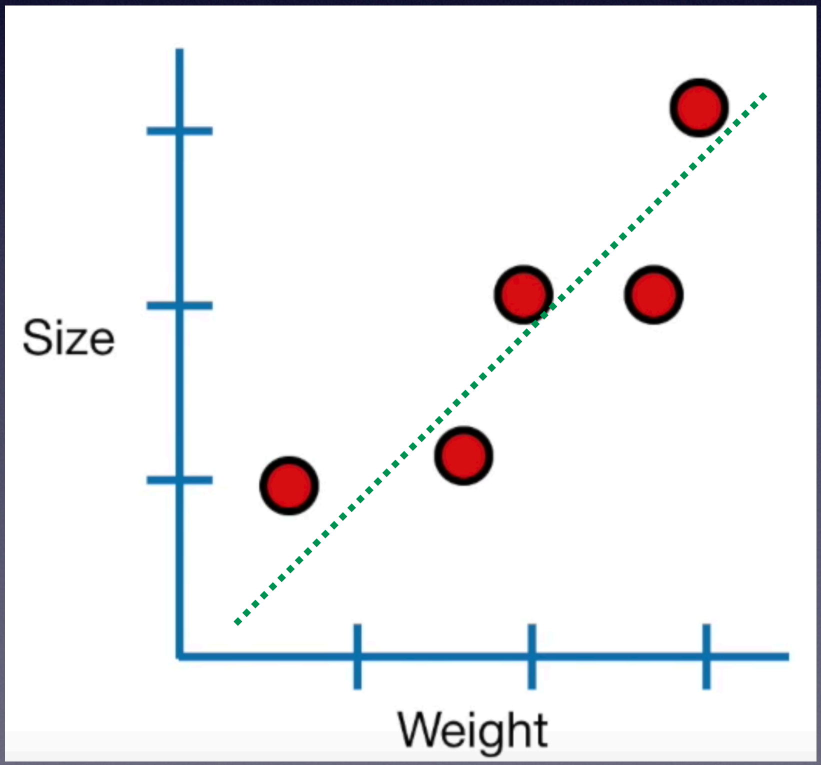
- Hinge Loss Function: $H_1(z) = \max(0, 1 - z)$





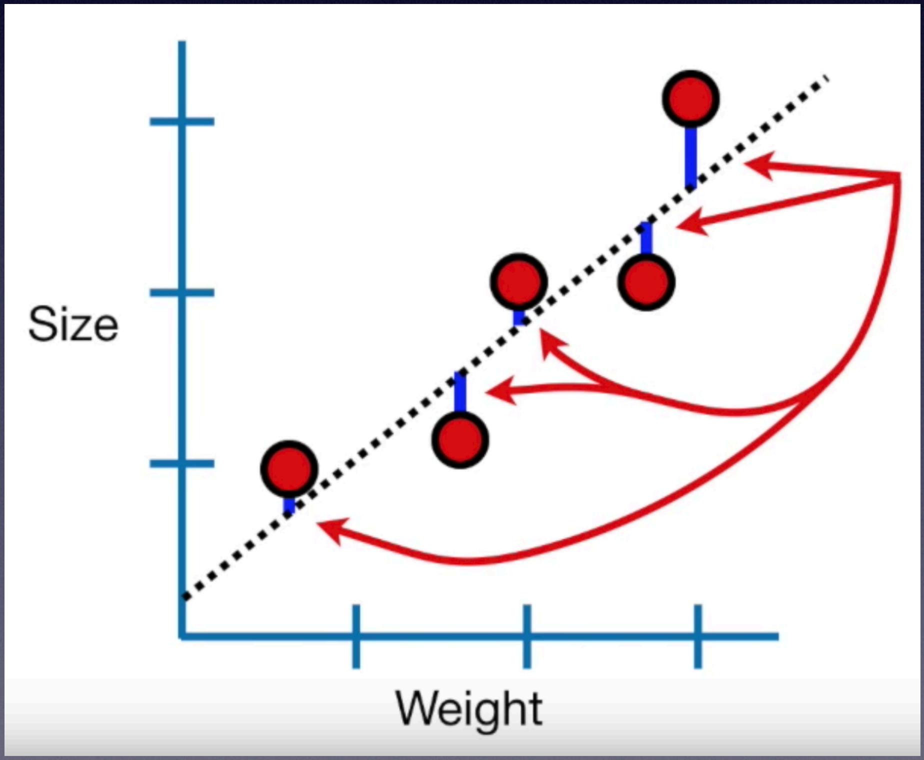
Linear Regression

- Linear regression can be used to fit a predictive model to an observed dataset



Linear Regression's Objective

- Linear regression models are often fitted using the least square objective
- A residual: the distance from the line to the data point
- Objective: we find the line that minimizes the sum of the squared of these residuals

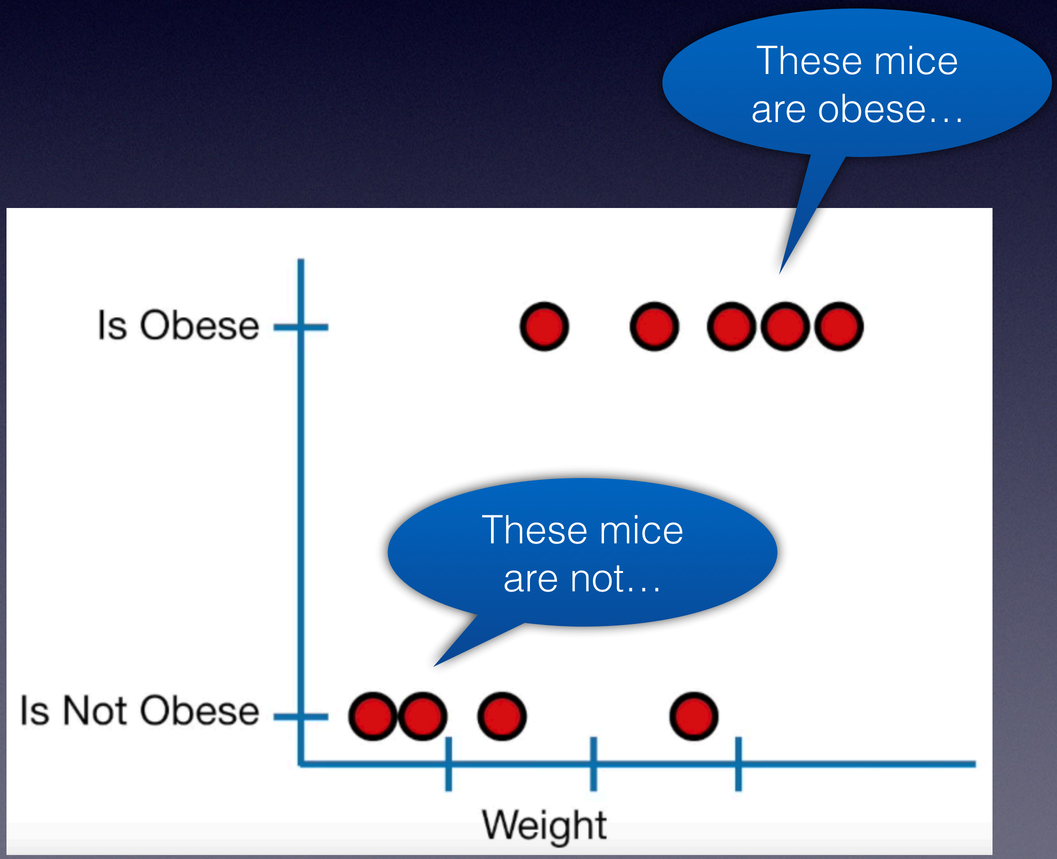


Logistic Regression

- We want to estimate not only which class an sample belongs to, but also the probability that a new data belongs to the class
- The most common approach logistic regression
- Logistic regression is widely used for probability estimation

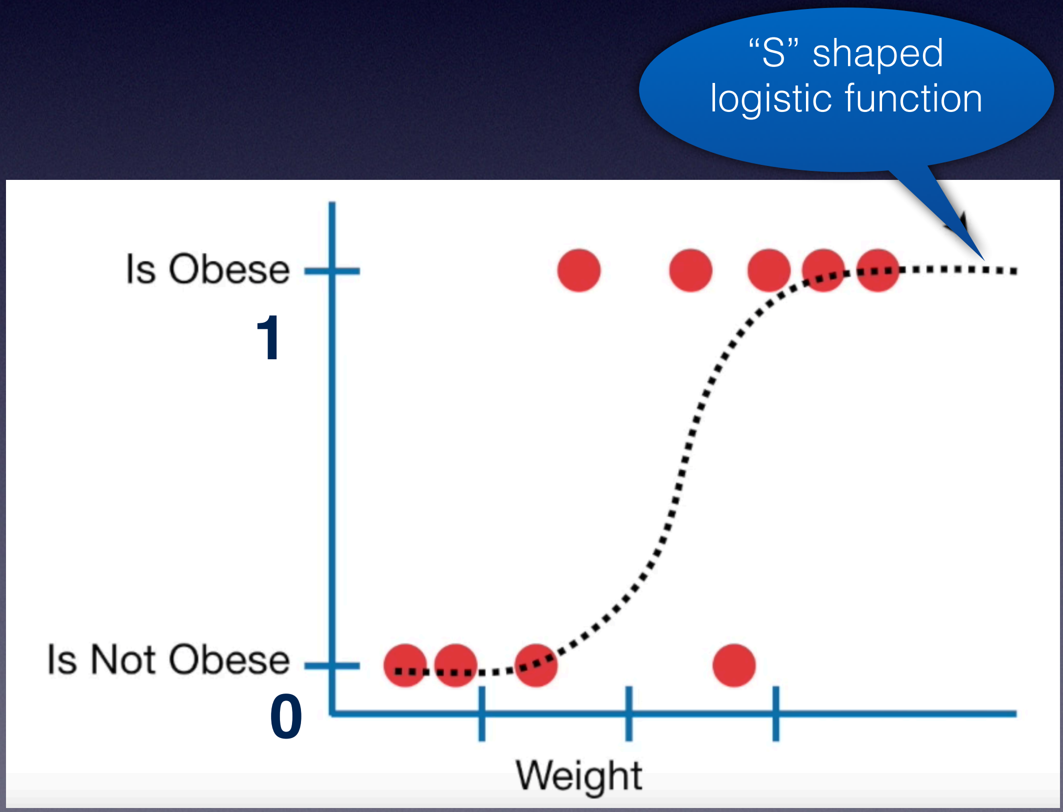
Logistic Regression Example

- Problem: estimate the probability of a mouse being obese given it's weight



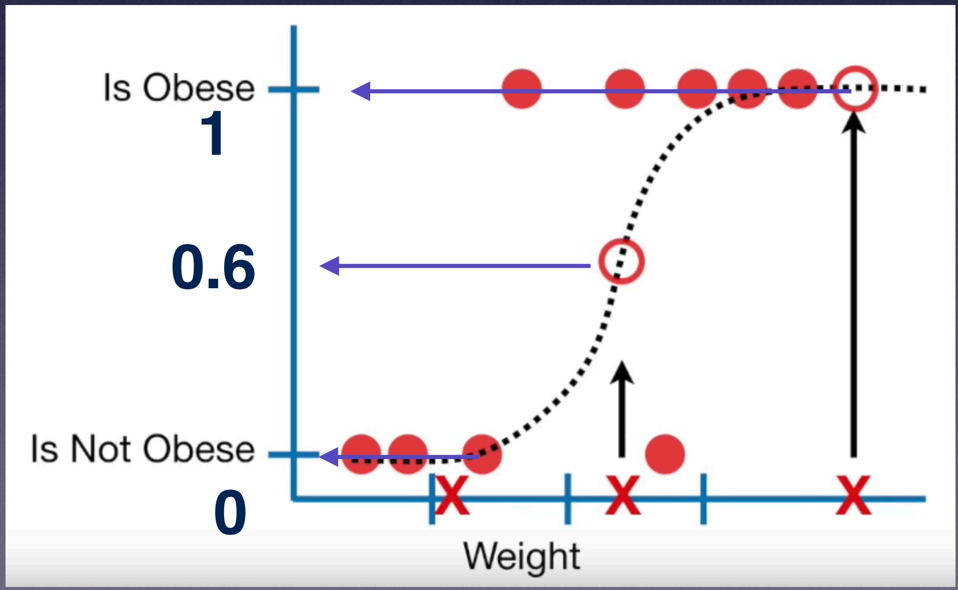
Logistic Regression Curve

- Logistic regression fits an “S” shaped “logistic function” to the data



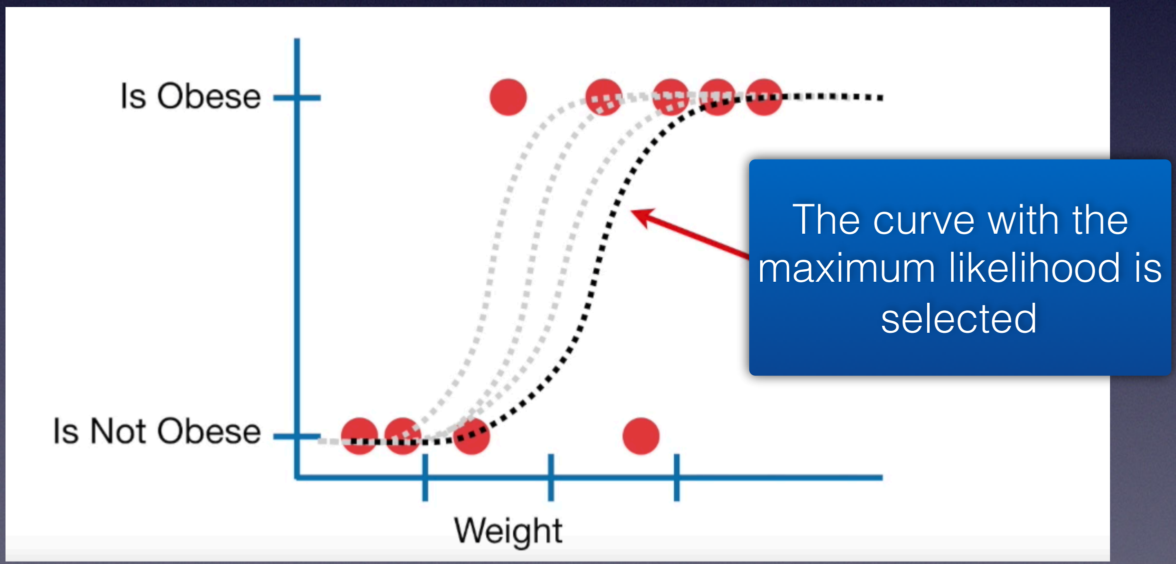
Logistic Regression

- Obesity is predicated with probability by weight
- Logistic regression's ability to provide probabilities makes it a popular machine learning method.



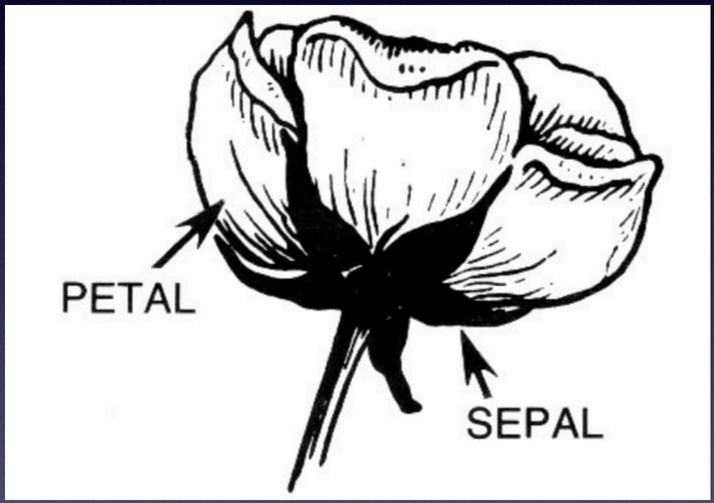
Logistic Regression’s Objective

- Objective: find the model (curve, or set of weights) with the“maximum likelihood”



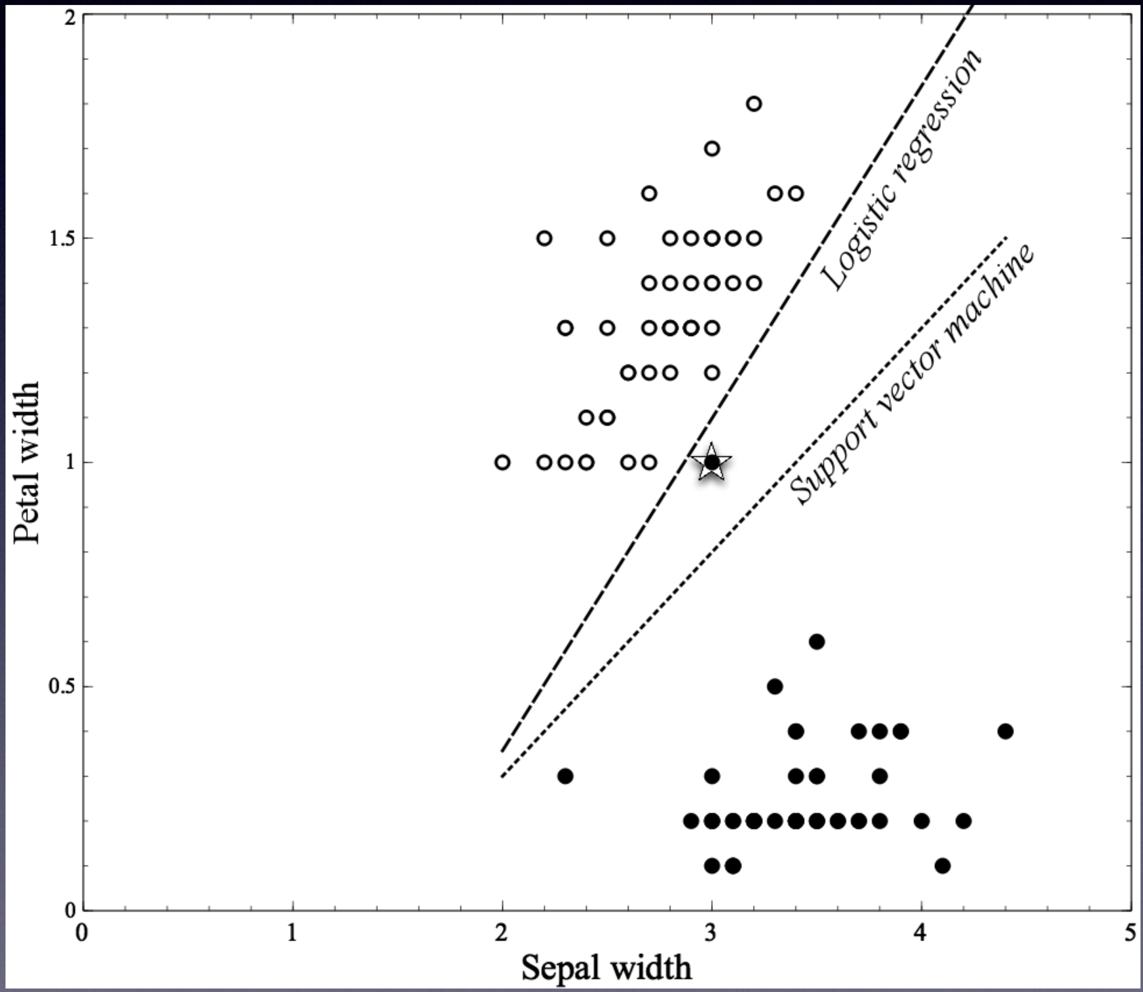
An Example with Iris Dataset

- Iris Dataset (UCI Dataset Repository, Bache & Lichman, 2013)
- Three specifies of irises
- Four attributes
- Problem: classify each instance as belonging to one of the three specified based on the attributes



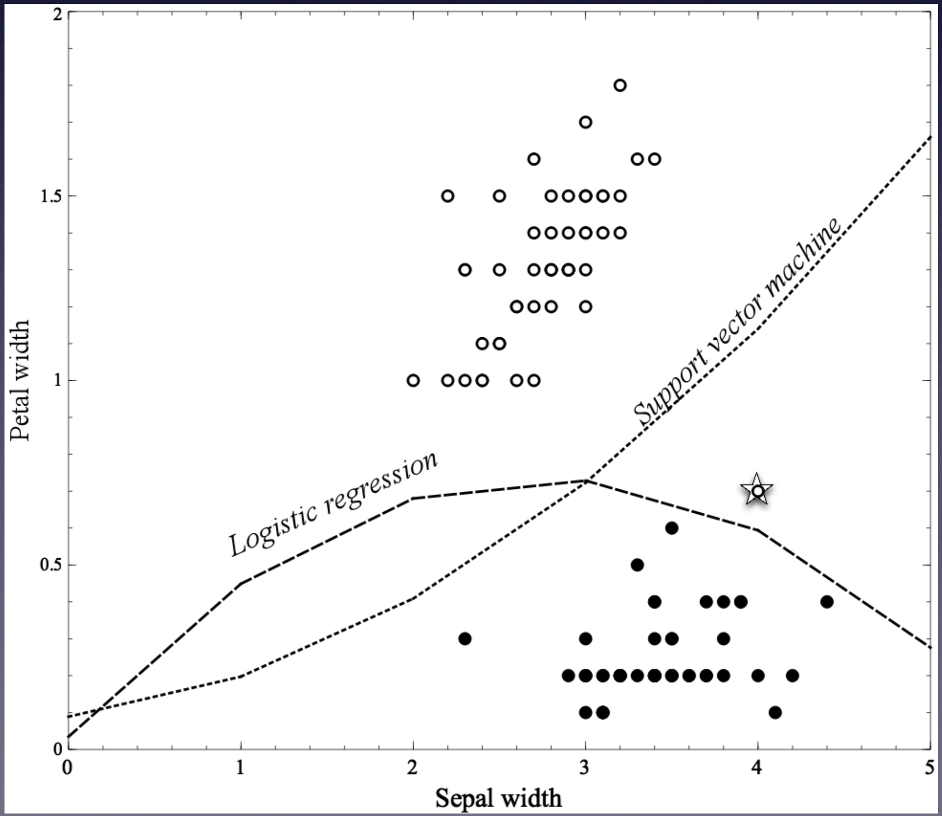
Two parts of a flower. Width measurements of these are used in the Iris dataset.

Two Learned Linear Classifiers on the Iris Dataset



Using Linear Functions to Present Nonlinear Classifiers

- **Linear functions can actually represent nonlinear models**, if we include more complex features in the functions



In this figure, logistic regression and support vector machine—both linear models—are provided an additional feature, Sepal width^2, which allows both the freedom to create more complex, nonlinear models.

Nonlinear SVMs

- Kernel Function: maps the original features to some other feature space
- Then the linear model is fit to this new feature space
- It considers “higher-order” combinations of the original features (e.g., squared features, products of features)

Hands-on: Logistic Regression

- Download Class 5 lab from blackboard

Next Week

- Quiz #1 (Monday, Oct. 15)
  - Slides of class 1-5
  - Chapter 1-4 of “Data Science for Business” by Foster Provost & Tom Fawcett
  - Open book & open notes
  - Bring a pen!
  - No laptops, smartphone, tablets