Institut Informatika & Bisnis DARMAJAYA
Yayasan Alfian Husin

SAINS DATA
Institut Informatika & Bisnis Darmajaya

Kampus Merdeka INDONESIA JAYA

MERDEKA BELAJAR

DATA SCIENCE DARMAJAYA
"YOUR BEST FUTURE IN DATA"

MEETING: [9]

FITTING A MODEL TO DATA

BY: HENDRA KURNIAWAN

[12-09-2023] | [KODE MK: BDG23407 | MATAKULIAH: DATA SCIENCE AND BUSINESS ANALYTICS | SKS: 2/2 | VERSI: 01

Fitting a Model to Data

1. Classification via Mathematical Functions
2. Regression via Mathematical Functions
3. Class Probability Estimation and Logistic “Regression”

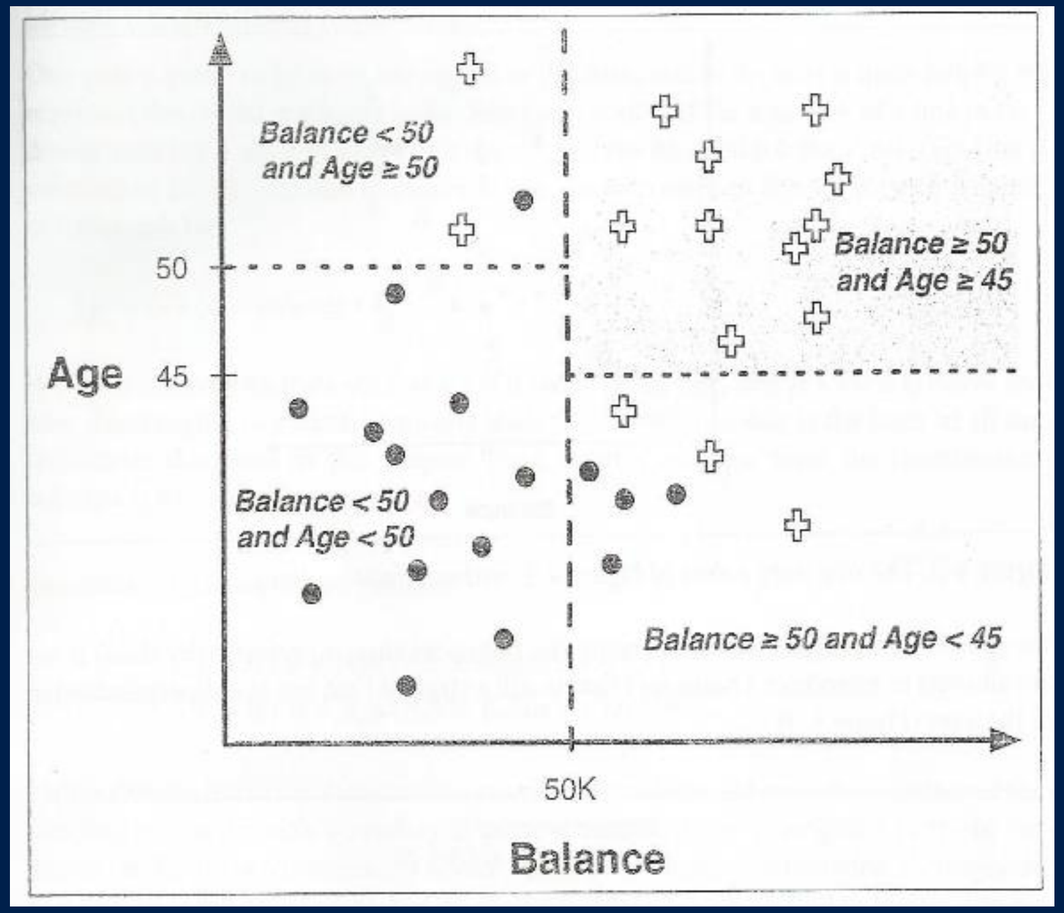
Fundamental concepts:

- Finding “optimal” model parameters based on data;
- Choosing the goal for data mining;
- Objective functions;
- Loss functions.

Fitting a model to data

- An alternative method for learning a predictive model from a dataset is to start by specifying the structure of the model with certain numeric parameters left unspecified.
- Then the data mining calculates the best parameter values given a particular set of training data.
- The goal of data mining is to tune the parameters so that the model fits the data as well as possible.
- This general approach is called *parameter learning* or *parametric modeling*.

Classification via Mathematical Function

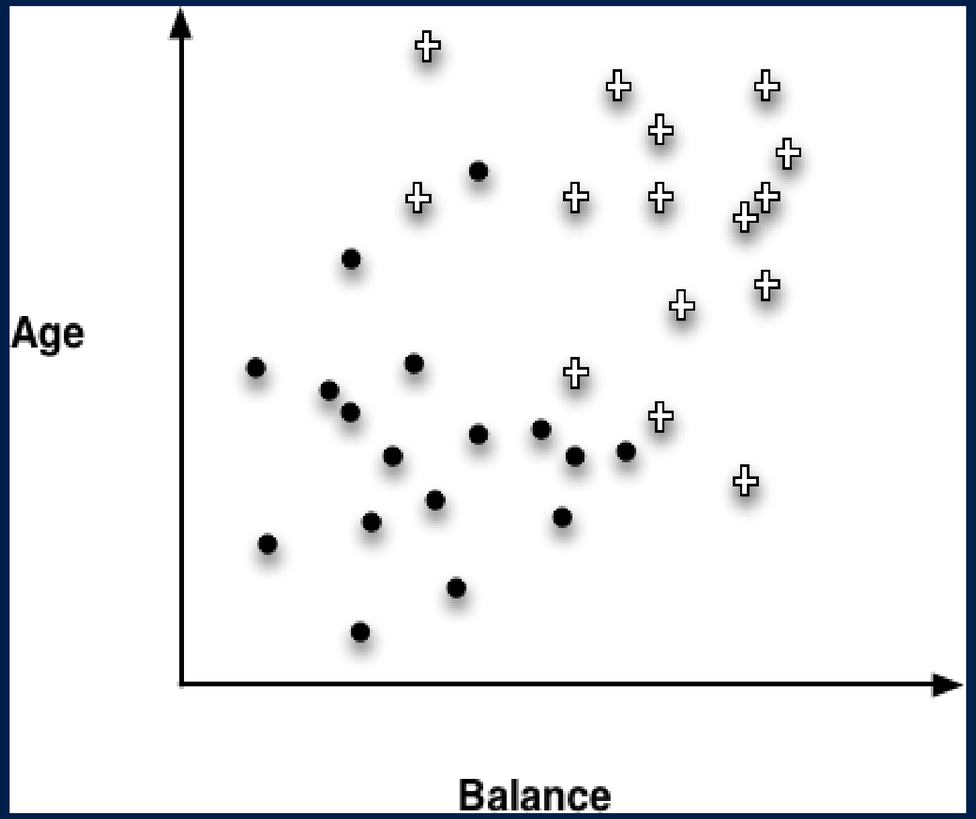


Action: Separate row data into some similar regions

- It shows the space broken up into regions by horizontal and vertical decision boundaries that partition the instance space into similar regions.
- A main purpose of creating homogeneous regions is so that we can predict the target variable of a new, unseen instance by determining which segment it falls into.

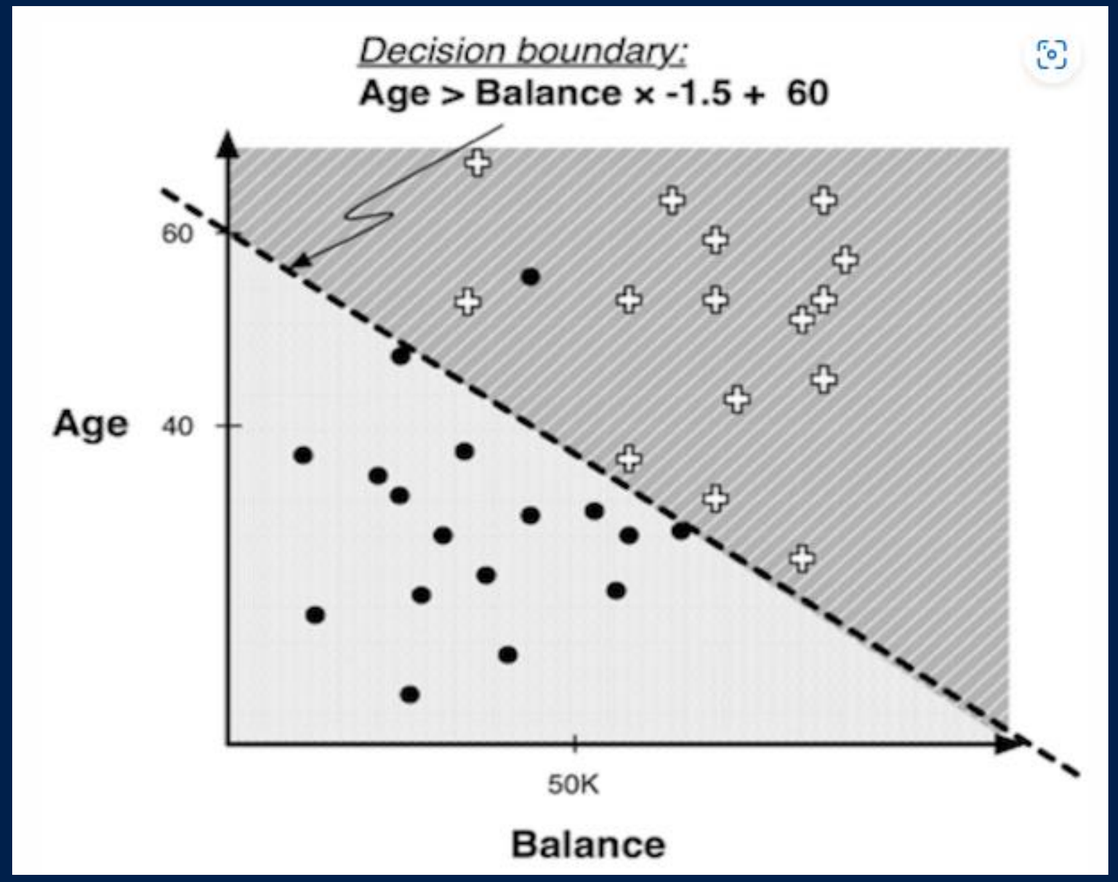
Decision barriers are perpendicular to the axis. Is there another way we can classify this data?

Classification via Mathematical Function



The raw data without decision lines.

Classification via Mathematical Function (Linear Discrimination)



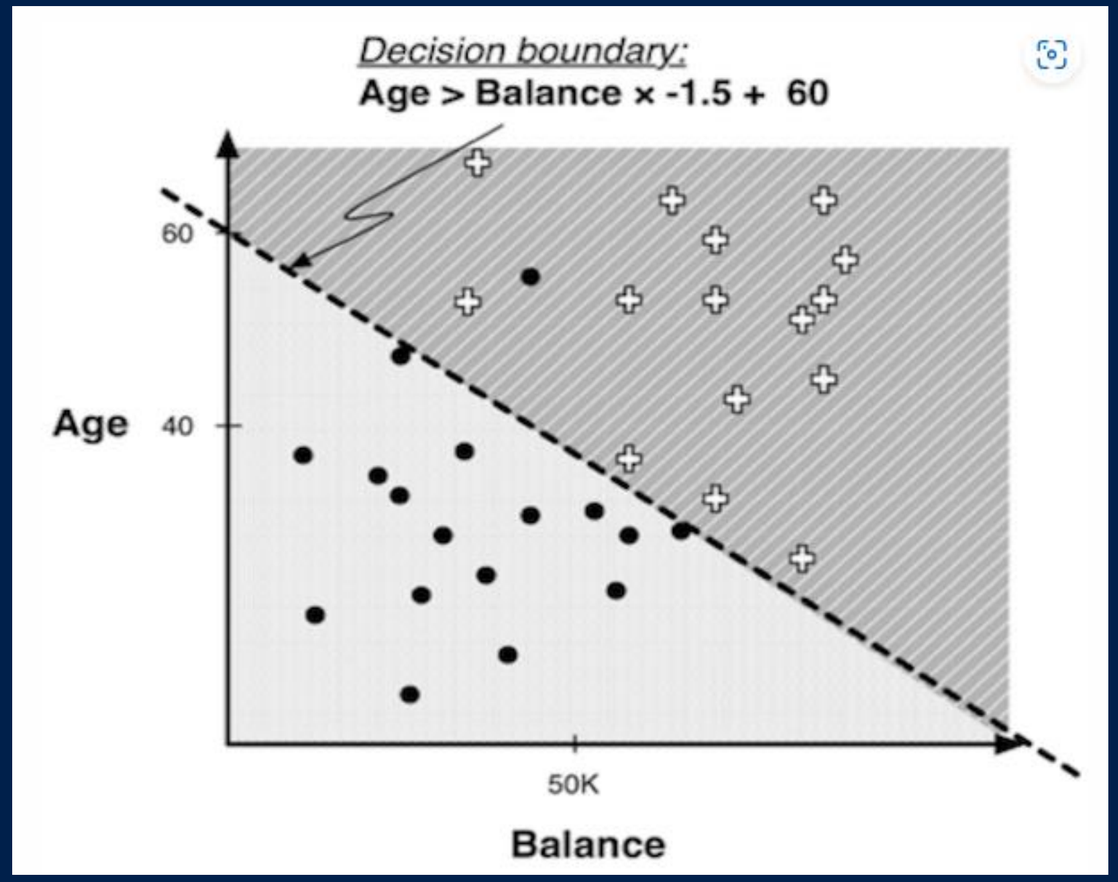
By being able to draw a line that is still straight, but not perpendicular, we can better divide the space:

- It shows the space broken up into regions by horizontal and vertical decision boundaries that partition the instance space into similar regions.
- A main purpose of creating homogeneous regions is so that we can predict the target variable of a new, unseen instance by determining which segment it falls into.

The new decision boundary is essentially a weighted sum of the values of the various attributes:

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

Classification via Mathematical Function (Linear Discrimination)



By being able to draw a line that is still straight, but not perpendicular, we can better divide the space:

- It shows the space broken up into regions by horizontal and vertical decision boundaries that partition the instance space into similar regions.
- A main purpose of creating homogeneous regions is so that we can predict the target variable of a new, unseen instance by determining which segment it falls into.

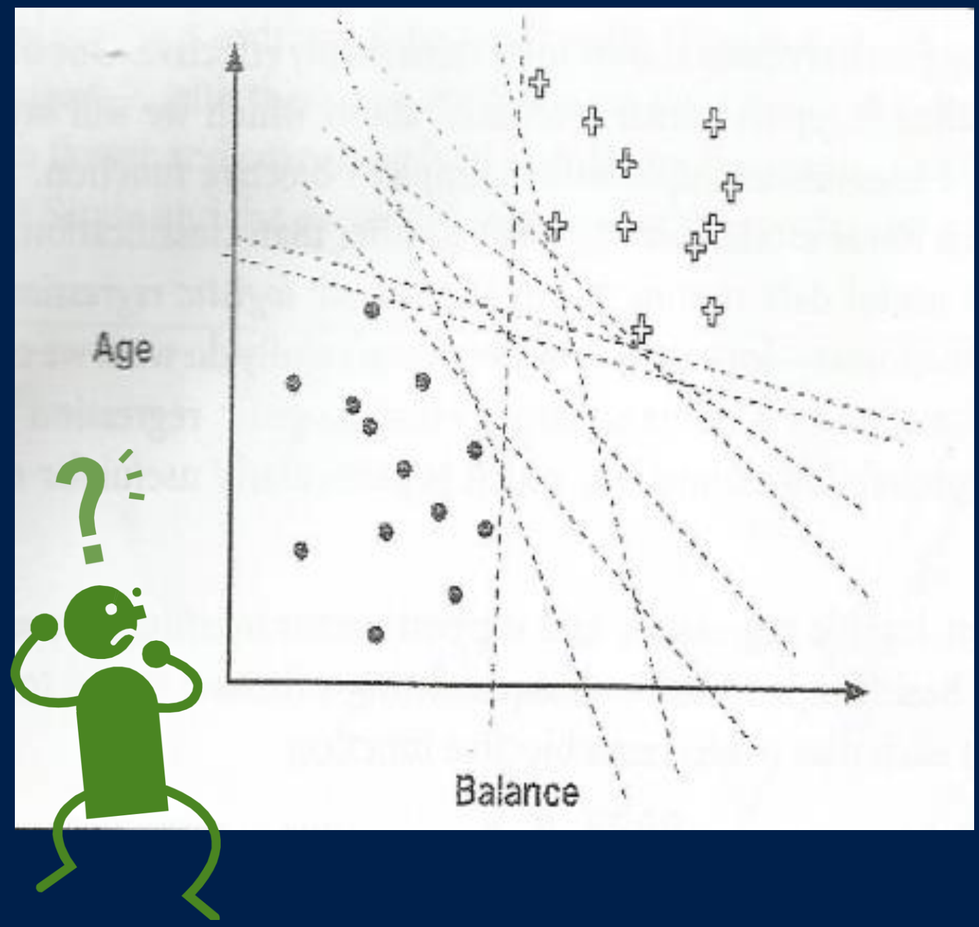
The new decision boundary is essentially a weighted sum of the values of the various attributes:

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

Classification via Mathematical Function (Linear Discrimination)

There are actually many different linear discriminants that can separate the classes perfectly. They have different slopes and intercepts, and each represents a different model of the data.

Which is the “best” line?

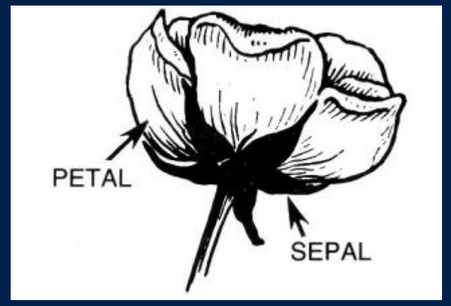


Optimizing an Objective Function

- **STEP1**：Define an objective function that represents our goal.
- **STEP2**：The function can be calculated for a particular set of weights and a particular set of data.
- **STEP3**：Find the optimal value for the weights by maximizing or minimizing the objective function.

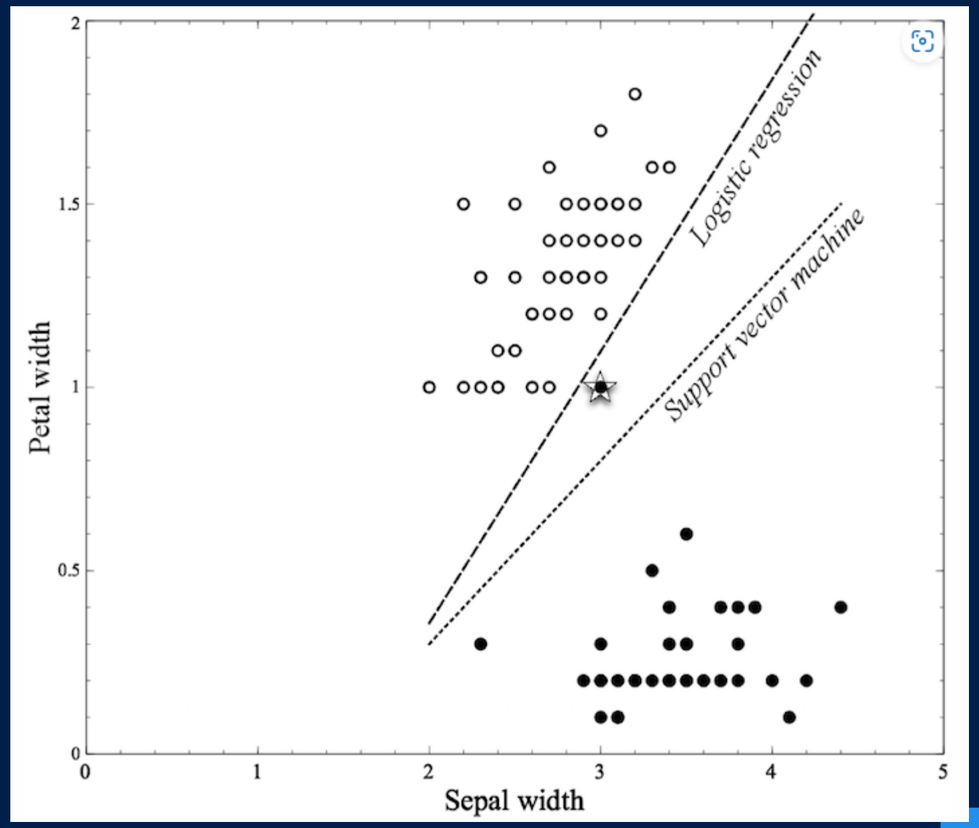
An example of mining a linear discriminant from data

To illustrate linear discriminant functions, we use an adaptation of the Iris dataset taken from the UCI Dataset Repository (Bache & Lichman, 2013). This is an old and fairly simple dataset representing various types of iris, a genus of flowering plant. The original dataset includes three species of irises represented with four attributes, and the data mining problem is to classify each instance as belonging to one of the three species based on the attributes.



For this illustration we'll use just two species of irises, Iris Setosa and Iris Versicolor. The dataset describes a collection of flowers of these two species, each described with two measurements: the Petal width and the Sepal width

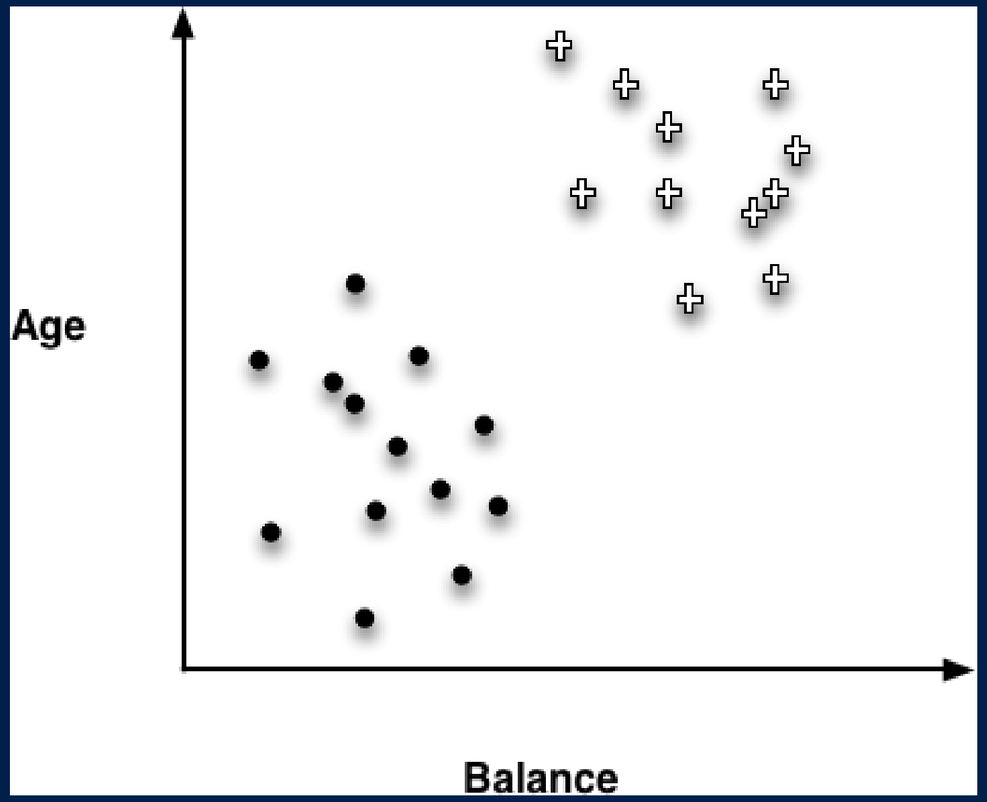
Two different separation lines are shown in the figure, one generated by logistic regression and the second by another linear method, a support vector machine



Linear Models for Scoring and Ranking Instances

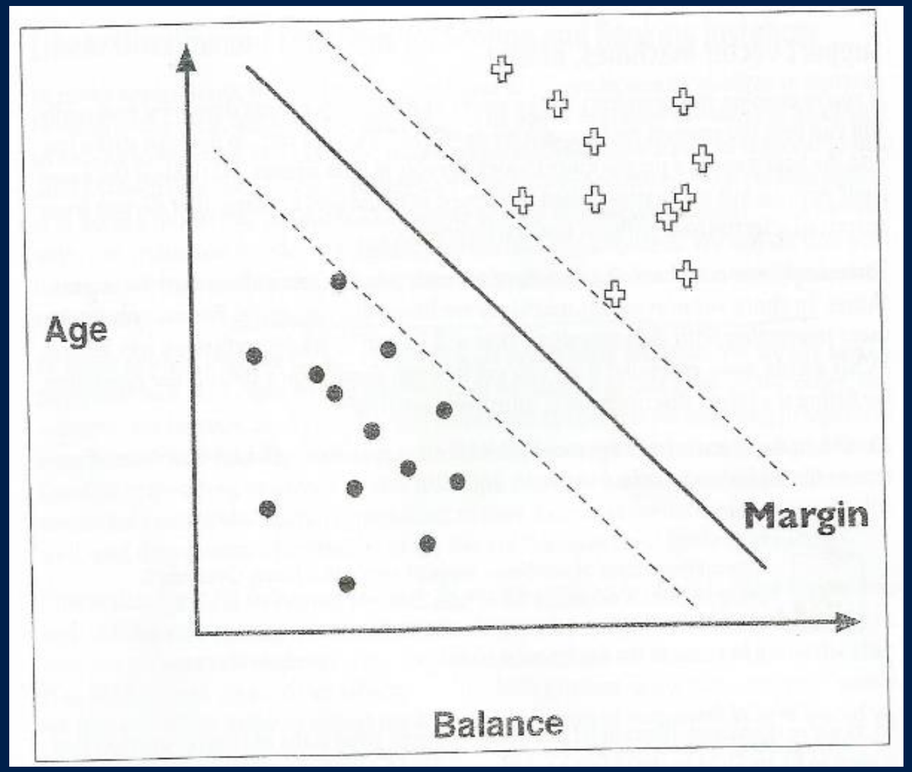
- In some applications, we don’t need a precise probability estimate. We simply need to rank cases by the likelihood of belonging to one class or the other.
- For example, for targeted marketing we may have a limited budget for targeting prospective customers.
- We need a list of customers as to their likelihood for responding to our marketing offers.

Linear discriminant functions can give us such a ranking for free



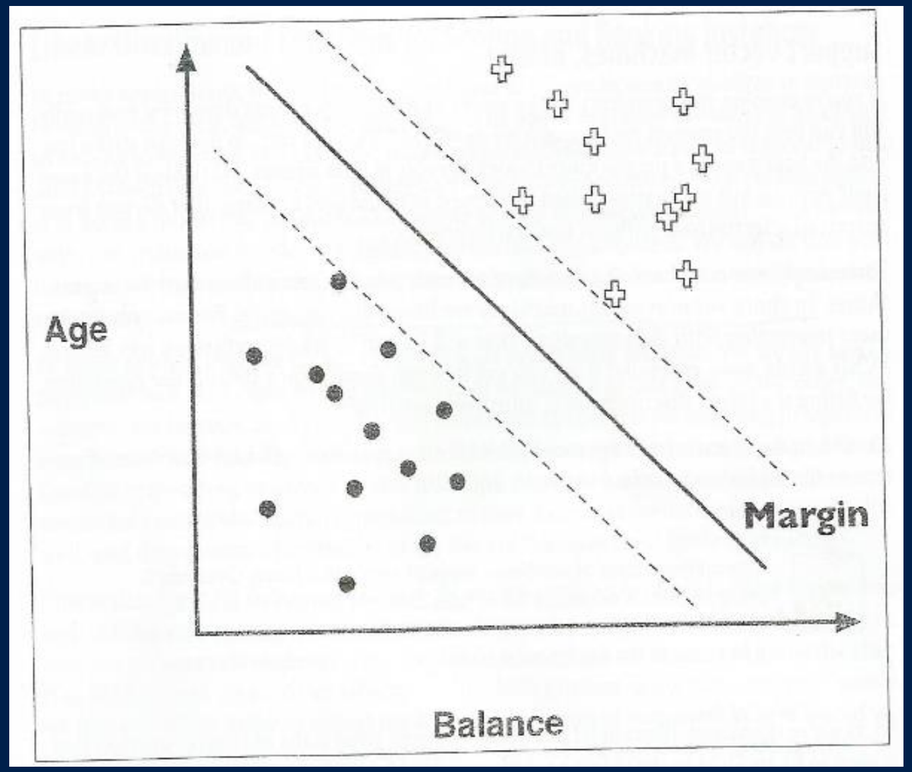
- Look at Figure beside
- Consider the + instances to be responders and • instances to be non-responders.
- A linear discriminant function gives a line that separate the two classes.
- If a new customer **x** happens to be on the line, i.e., f(**x**) = 0, then we are most uncertain about his/her class.
- But if f(**x**) is **positive** and **large**, then we would be certain that **x** will most likely be a responder.
- Thus f(**x**) essentially gives a ranking.

Support Vector Machine



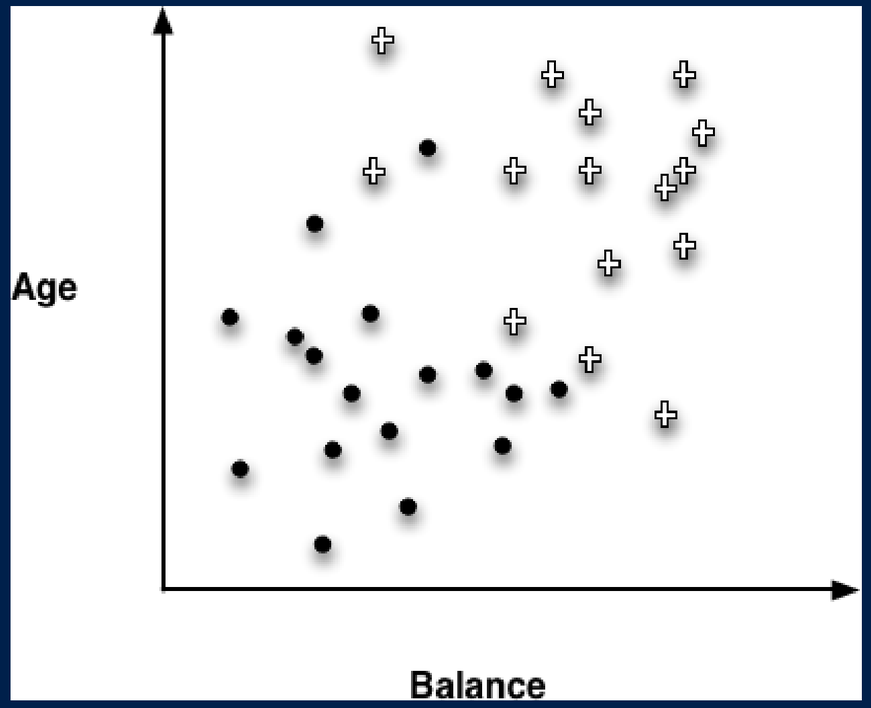
- SVM is a type of linear discriminants.
- Instead of thinking about separating with a line, first fit the **fattest bar** between the classes. Shown by **parallel dashed lines** in the figure.
- SVM's objective function incorporates the idea that a wider bar is better.
- Once the widest bar is found, the linear discriminant will be the center line through the bar.

Support Vector Machine



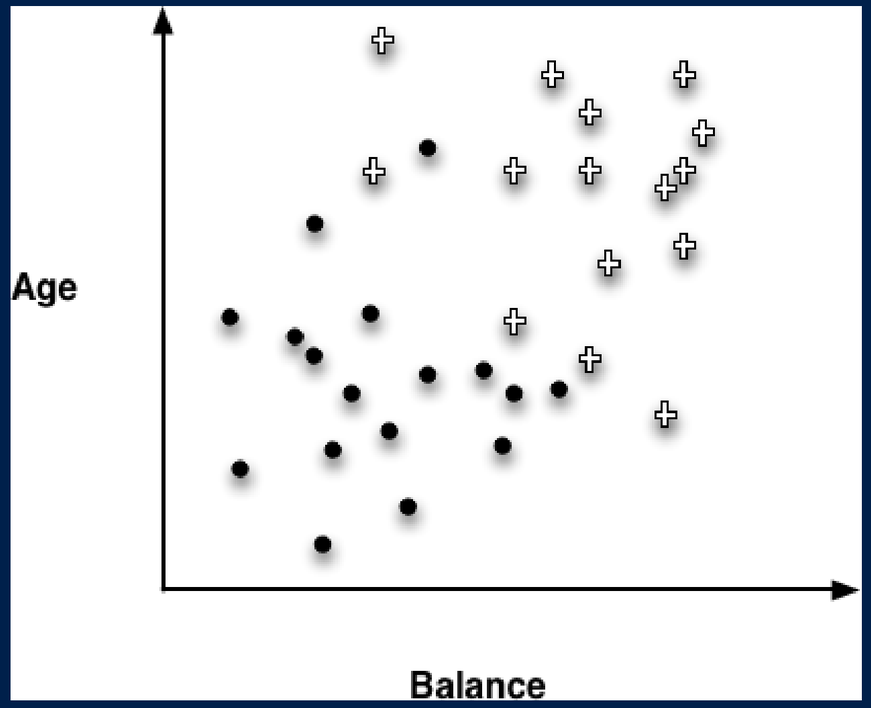
- The distance between the dashed parallel lines is called the margin around the linear discriminant.
- The objective is to maximize the margin.
- The margin-maximizing boundary gives the maximal leeway for classifying points that fall closer to the boundary.

Support Vector Machine



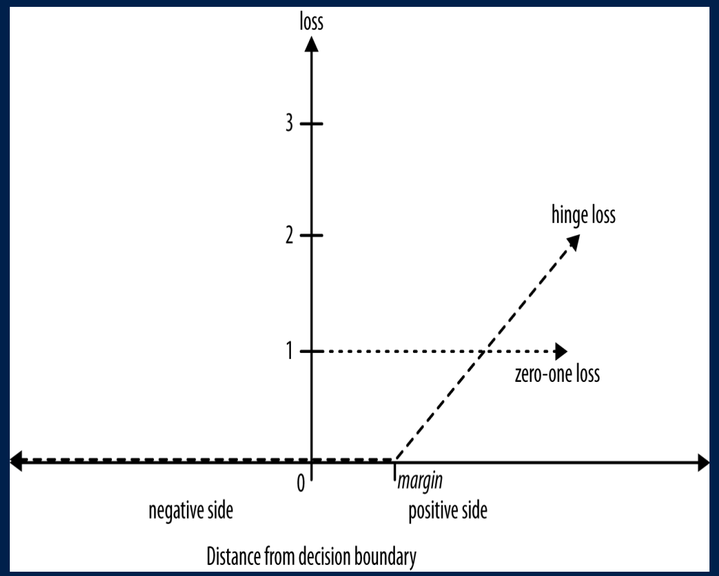
- Sometimes a single line can not perfectly separate the data into classes, as the following figure shows.
- SVM's objective function will penalize a training point for being on the wrong side of the decision boundary.

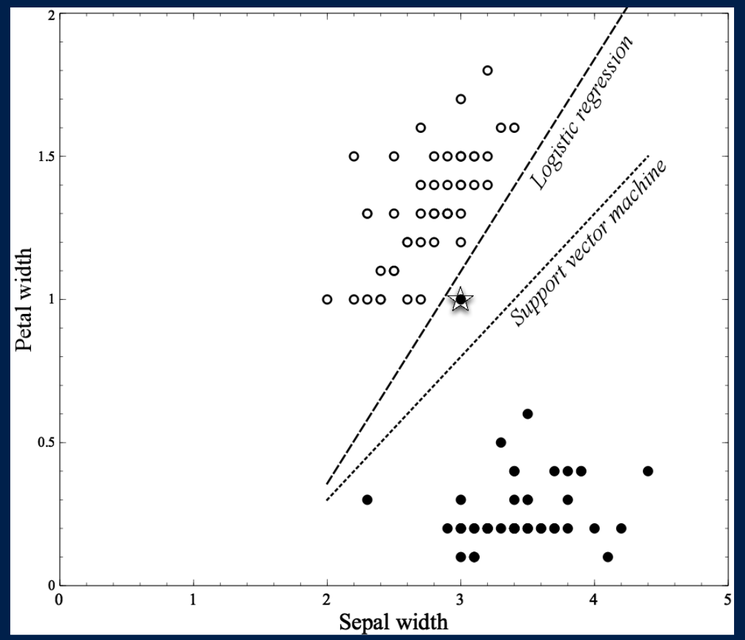
Support Vector Machine



- If the data are not linearly separable, the best fit is some balance between a fat margin and a low total error penalty.
- The penalty for a misclassified point is proportional to the distance from the marginal boundary..

Support Vector Machine





The term "**loss**" is used a general term for error penalty. Support vector uses hinge loss. The **hinge loss** only becomes positive when an example is on the wrong side of the boundary and beyond the margin. **Zero-one loss** assigns a loss of zero for a correct decision and one for an Incorrect decision.

Class probability estimation and logistic regression

- Linear discriminants could be used to give estimates of class probability.
- The most common procedure is called logistic regression.
- Consider the problem of using the basic linear model to estimate the class probability: $f(x) = w_0+w_1x_1+w_2x_2+.....$

Class probability estimation and logistic regression

- As we discussed, an instance being further from the separating boundary ought to lead to a higher probability of being in one class or the other, and the output of the linear function, f(x), gives the distance from the separating boundary.
- However, this shows a problem: f(x) ranges from -∞ to ∞, and a probability should range from zero to one.

Class probability estimation and logistic regression

Probability	Corresponding odds
0.5	50:50 or 1
0.9	90:10 or 9
0.999	999:1 or 999
0.01	1:99 or 0.0101
0.001	1:999 or 0.001001

Probability ranges from zero to one, odds range from 0 to ∞, log-odds ranges from -∞ to ∞.

Probability	Odds	Log-odds
0.5	50:50 or 1	0
0.9	90:10 or 9	2.19
0.999	999:1 or 999	6.9
0.01	1:99 or 0.0101	−4.6
0.001	1:999 or 0.001001	−6.9

Class probability estimation and logistic regression

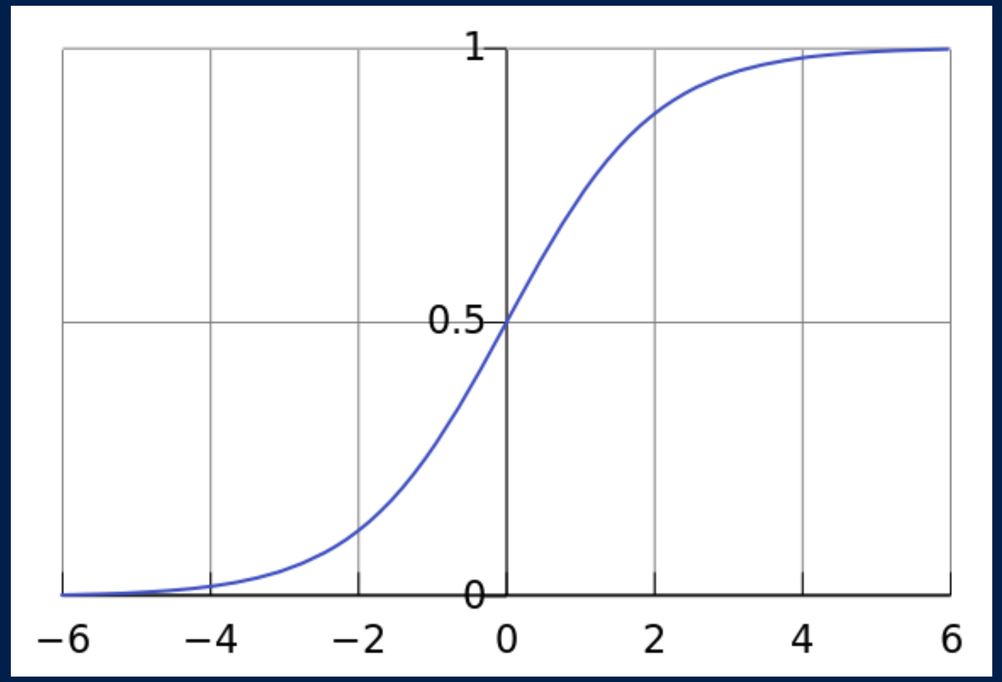
Equation 4-3. Log-odds linear function

$$\log\left(\frac{p_+(\mathbf{x})}{1 - p_+(\mathbf{x})}\right) = f(\mathbf{x}) = w_0 + w_1 x_1 + w_2 x_2 + \cdots$$

Equation 4-4. The logistic function

$$p_+(\mathbf{x}) = \frac{1}{1 + e^{-f(\mathbf{x})}}$$

$P_+(x)$: probability that a data item represented by feature x belongs to class +

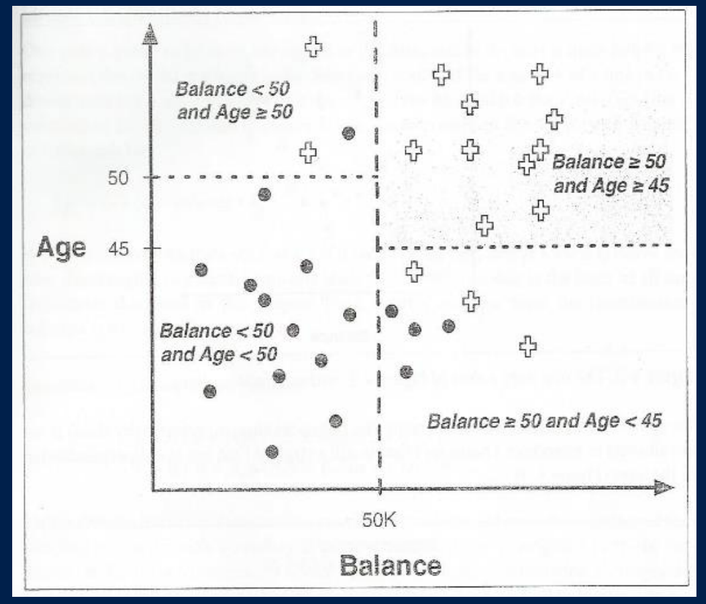


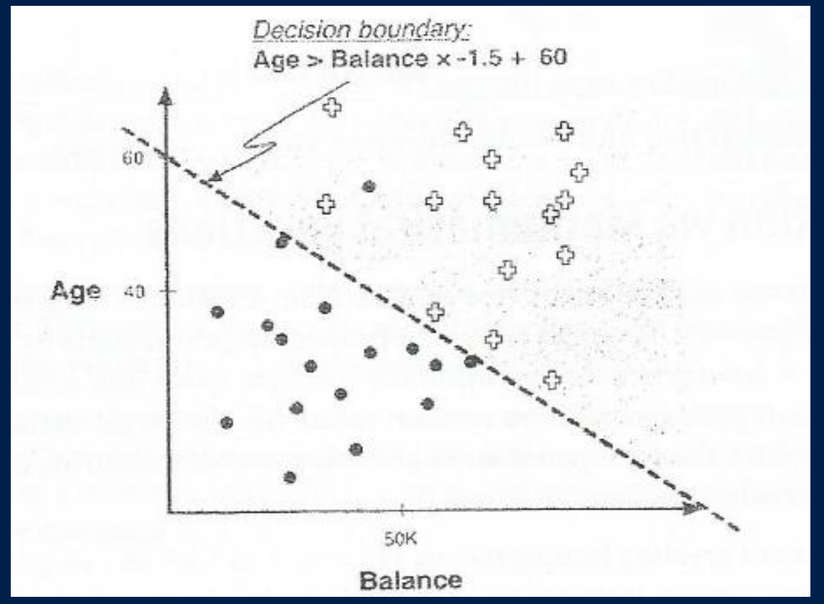
Distance from the decision boundary

Logistic Regression vs. Tree induction

A classification tree uses decision boundaries that are perpendicular to the instance-space axes, whereas the linear classifier can use decision boundaries of any direction or orientation.

Tree Induction: instance-space axes





Logistic Regression: any direction or orientation

Logistic Regression vs. Tree induction

- A classification tree is a “piecewise” classifier that segments the instance space recursively when it has to, using a divide-and-conquer approach. In principle, a classification tree can cut up the instance space arbitrarily finely into very small regions.
- A linear classifier places a single decision surface through the entire space. It has great freedom in the orientation of the surface, but it is limited to a single division into two segments.

Logistic Regression vs. Tree induction

- It is usually not easy to determine in advance which of these characteristics are a better match to a given dataset.
- When applied to a business problem, there is a difference in the comprehensibility of the models to stakeholders with different backgrounds.

Logistic Regression vs. Tree induction

- What an LR model is doing can be quite understandable to people with a strong background in statistics, and difficult to understand for those who do not.
- A DT (decision tree), if it is not too large, may be considerably more understandable to someone without a strong statistics or mathematics background.

Logistic Regression vs. Tree induction

- Why is this important? For many business problems, the data science team does not have the ultimate say in which models are used or implemented.
- Often there is at least one manager who must “sign off” on the use of a model in practice, and in many cases a set of stakeholders need to be satisfied with the model.

- This meeting focused on the fundamental concept of optimizing a model's fit to data.
- However, doing this leads to the most important fundamental *problem* with data mining—if you look hard enough, you will find structure in a dataset, even if it's just there by chance.
- This tendency is known as *overfitting*. Recognizing and avoiding overfitting is an important general topic in data science.

Overfitting

- When your learner outputs a classification that is 100% accurate on the training data but 50% accurate on test data, when in fact it could have output one that is 75% accurate on both, it has overfit.

Institut Informatika & Bisnis DARMAJAYA
Yayasan Alfian Husin

SAINS DATA
Institut Informatika & Bisnis Darmajaya

Kampus Merdeka INDONESIA JAYA

MERDEKA BELAJAR

THANK YOU!!

DATA SCIENCE DARMAJAYA “YOUR BEST FUTURE IN DATA”