Support Vector Machines 1

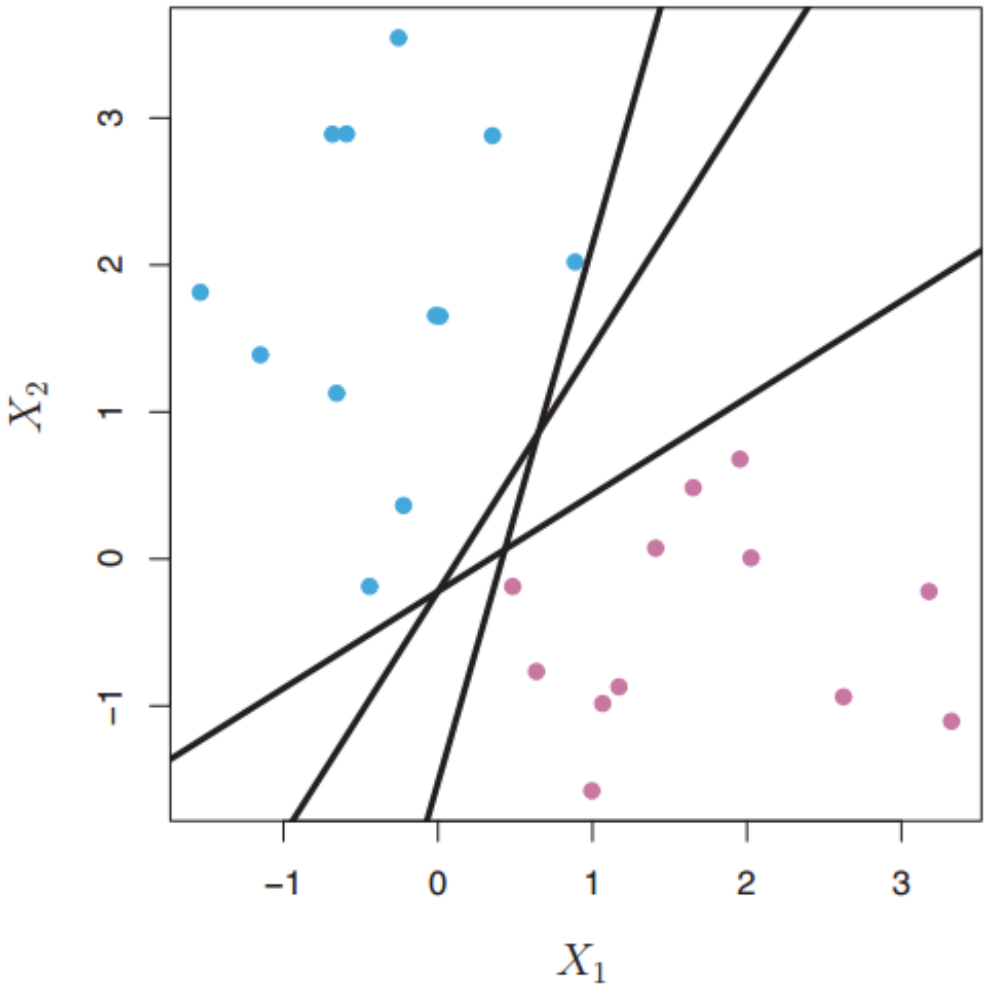
Glenn Bruns
CSUMB

Learning outcomes

After this lecture you should be able to:

- Be able to explain the basic ideas of support vector machines:
  - street
  - margin
  - support vector
- Be able to explain how support vector machines are related to logistic regression

Separating data with a line



The blue and red dots are training examples.

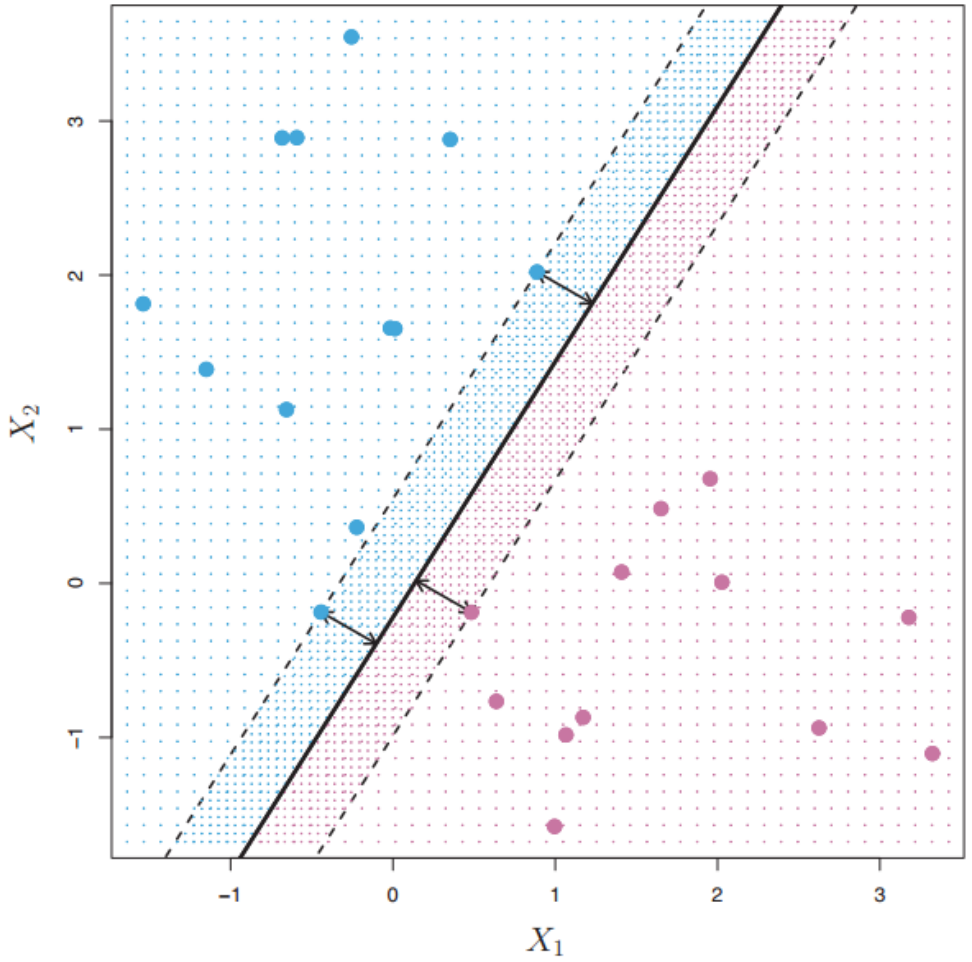
The two classes are linearly separable.

We can create a classifier by finding a line that separates the classes.

Which is the best line?

figure source: An Intro. to Statistical Learning, James et al

Hard margin classifier



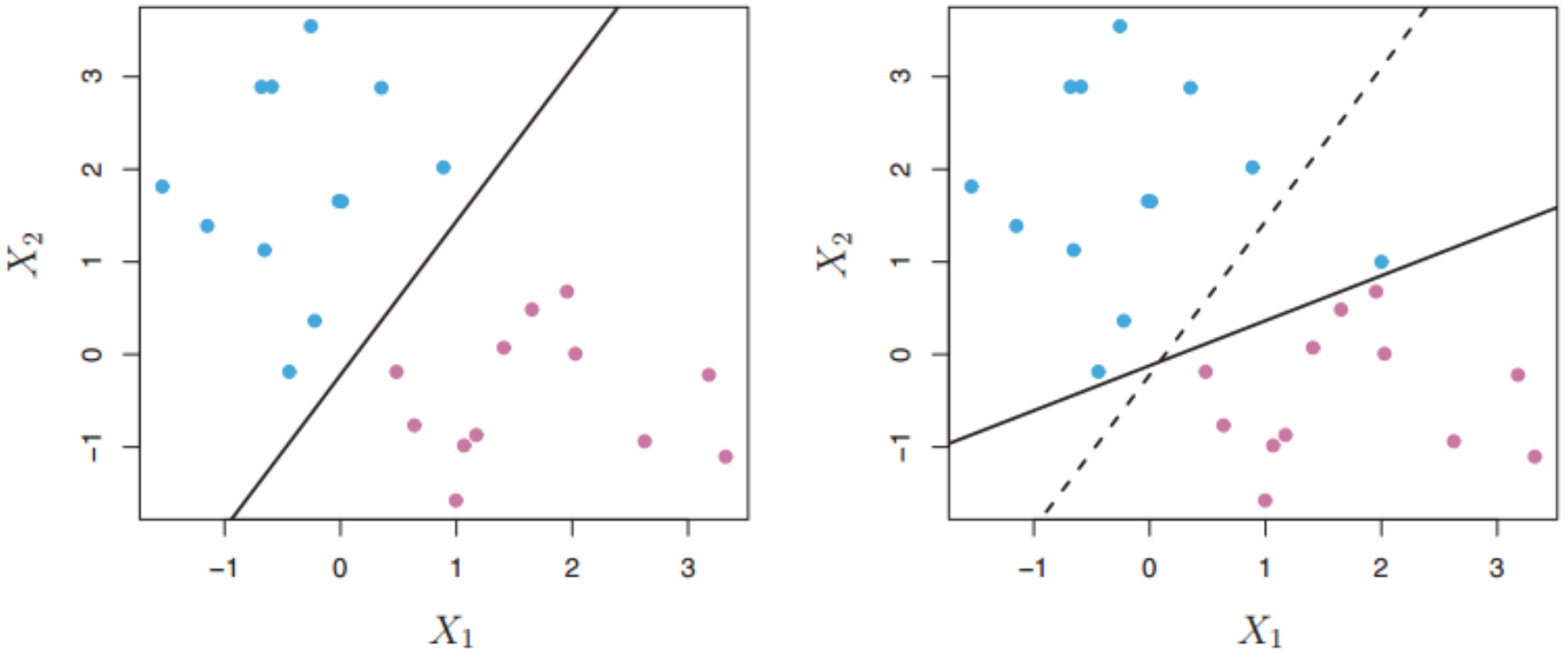
Find the widest possible "street" between the classes.

The margin is the smallest distance from any point to the separating line.

The points on the edge of the street are called support vectors

figure source: An Intro. to Statistical Learning, James et al

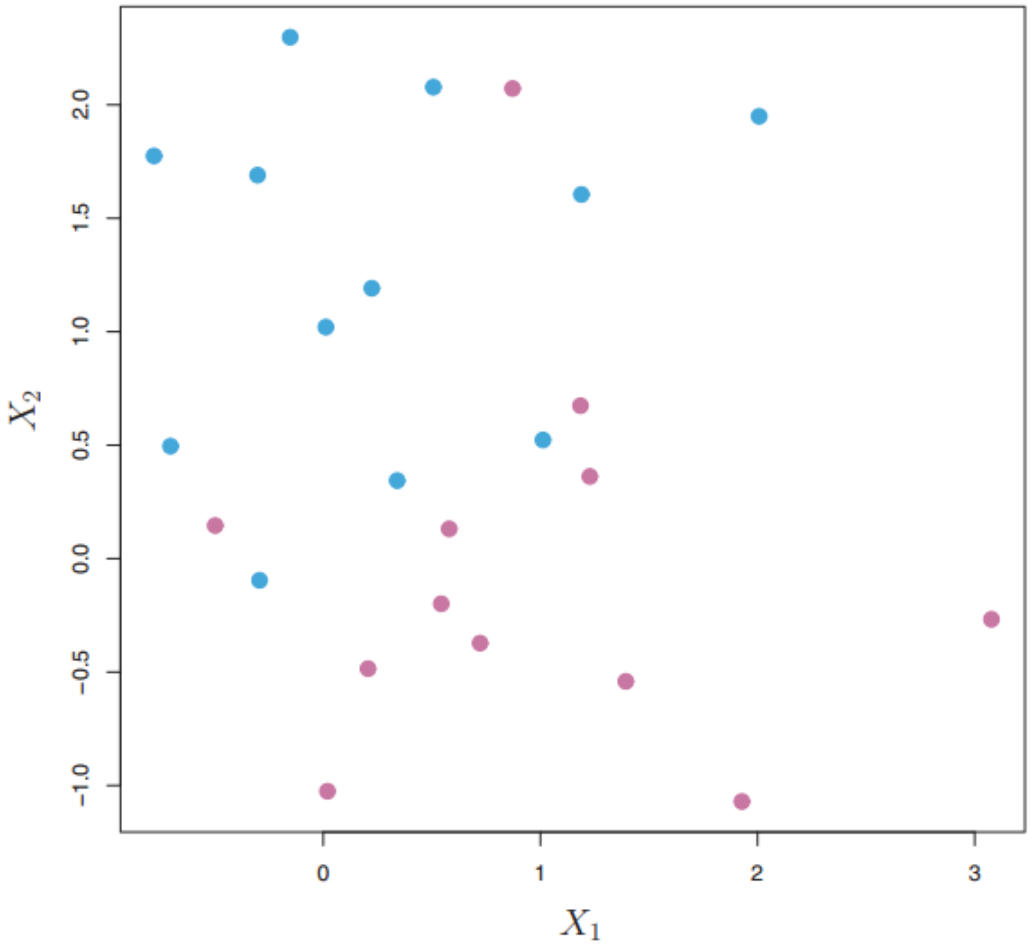
Hard margin classifier problem 1



Why are the decision boundaries so different in the two cases?

figure source: An Intro. to Statistical Learning, James et al

Hard margin classifier problem 2

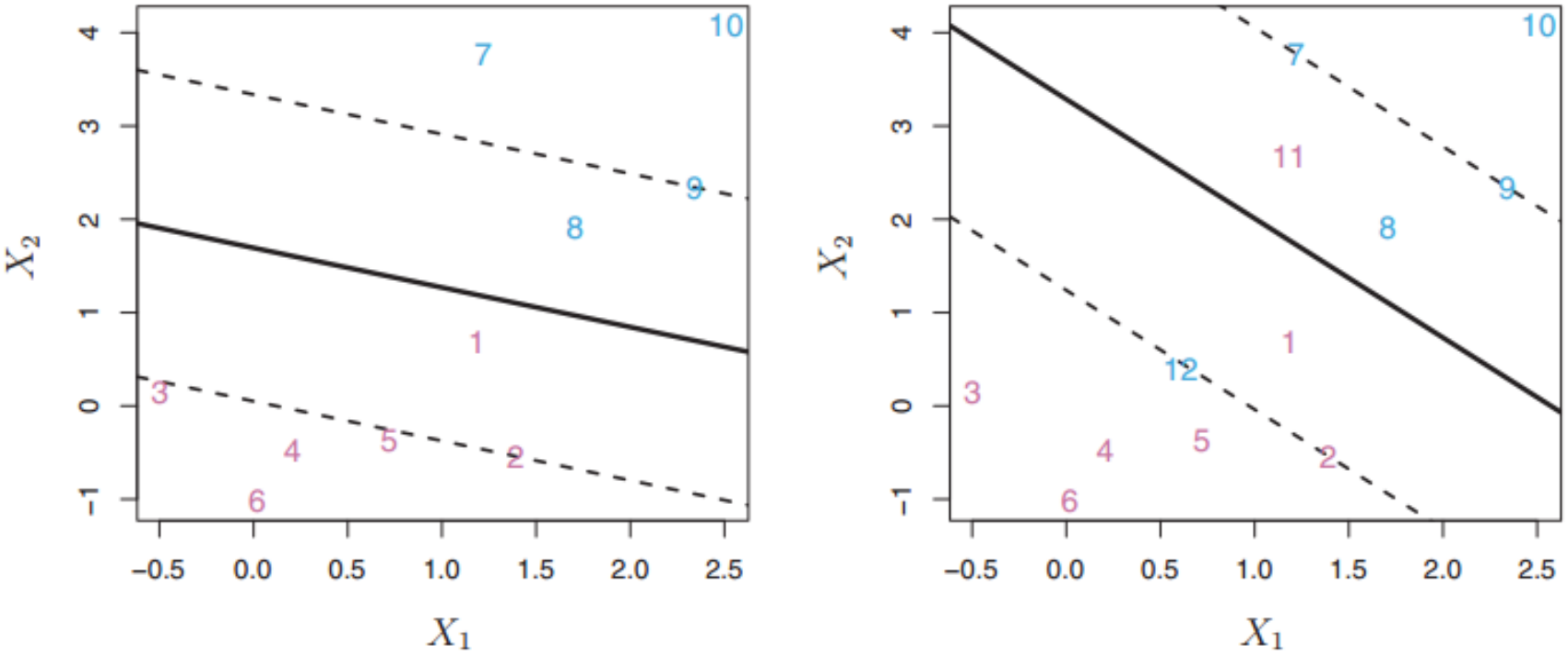


Can you find a line that perfectly separates the classes?

How to extend the maximal margin classifier to handle this case? Retain the idea of a street.

figure source: An Intro. to Statistical Learning, James et al

Soft margin (or "support vector") classifier



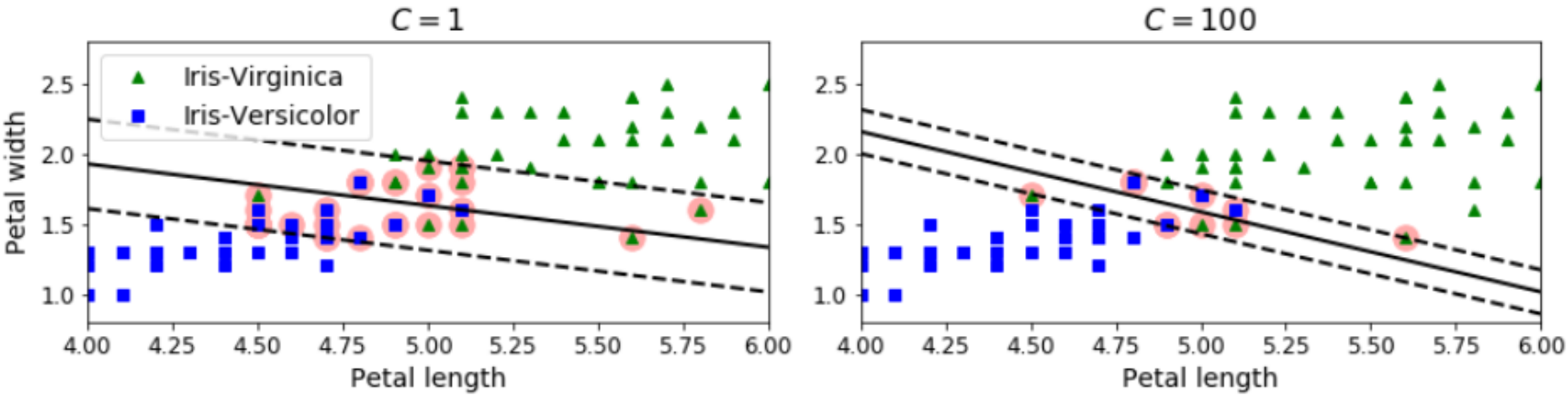
Allow training examples to be on the wrong side of the margin (even the wrong side of the line)

But … we want to limit margin violations

figure source: An Intro. to Statistical Learning, James et al

Impact of hyperparameter C

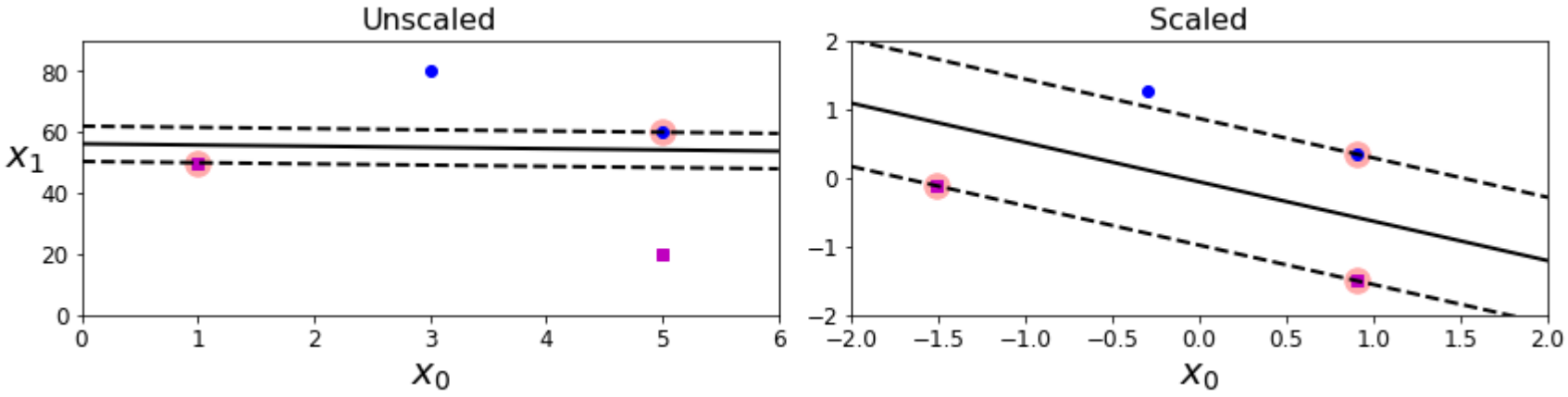
smaller C → wider street, more margin violations



Which classifier is likely to generalize better?

source: Geron, hands-on Machine Learning

Scaling data



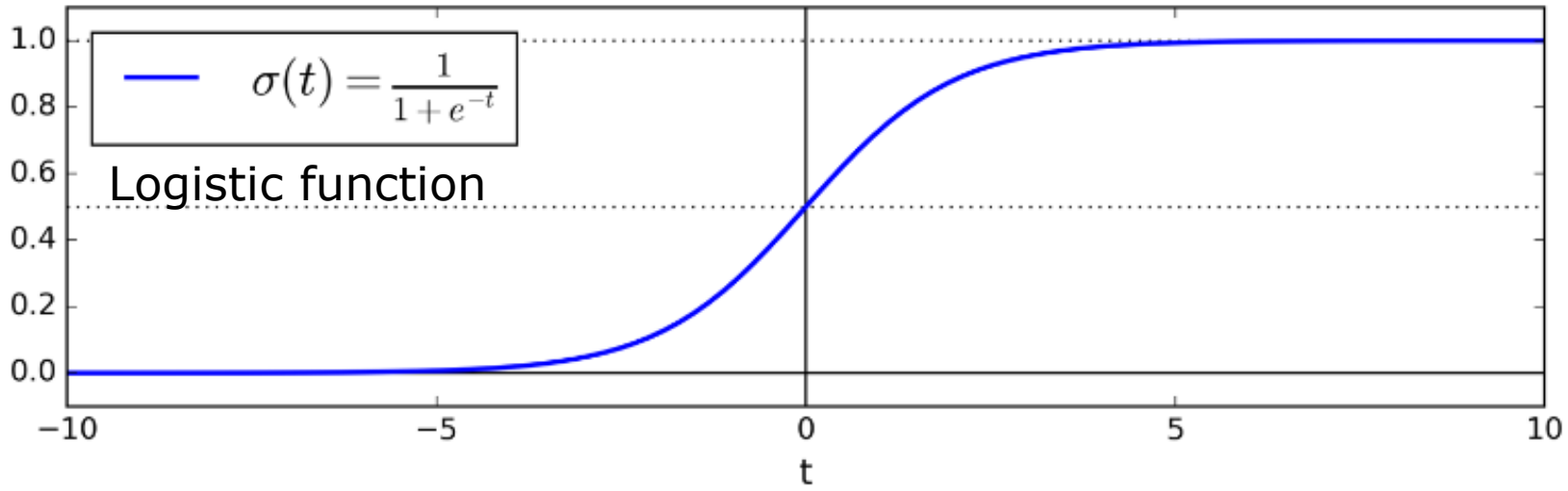
You should scale your data with SVM; use a method (like Z-score normalization) that gives a zero mean.

source: Geron, hands-on Machine Learning

Part 2: Connection to Log. Regression

To get class probabilities, first apply a linear model, then apply the logistic function:

$$\hat{p} = \sigma(\theta^T \cdot \boldsymbol{x})$$



When target value is 1, we want $\theta^T \cdot \boldsymbol{x} \gg 0$

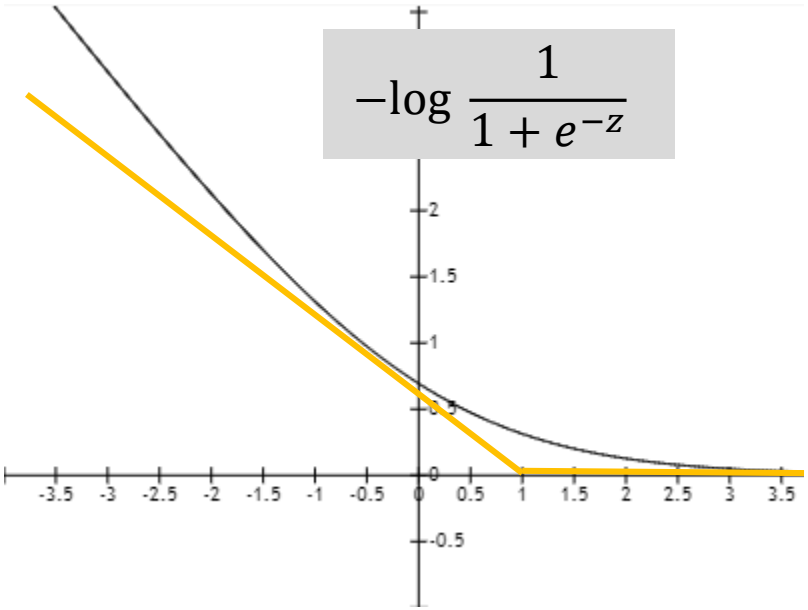
figure: Géron

Logistic Regression cost function

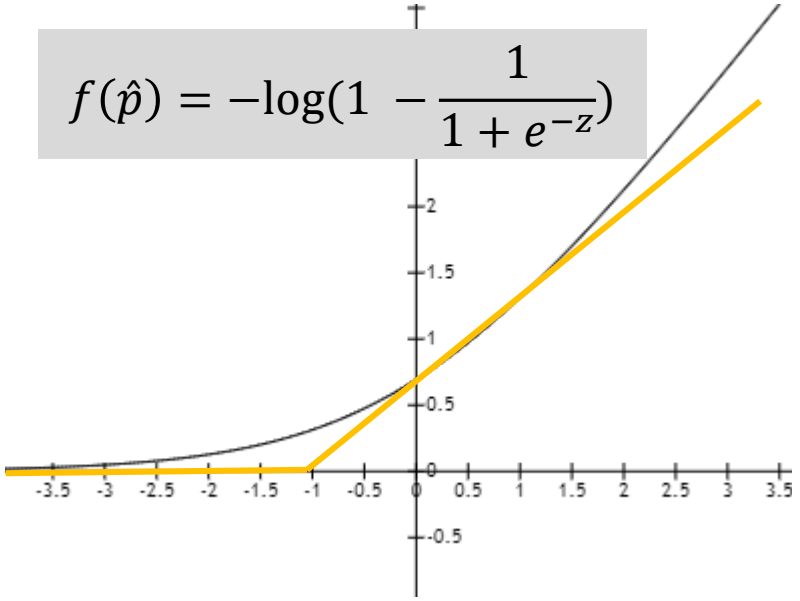
$$\text{cost} = -(y \log(\hat{p}) + (1 - y) \log(1 - \hat{p}))$$

$$= -y \log \frac{1}{1 + e^{-\theta^T \cdot x}} - (1 - y) \log(1 - \frac{1}{1 + e^{-\theta^T \cdot x}})$$

if y = 1



if y = 0



source: Andrew Ng, Coursera lecture 12.2

Transforming feature vectors

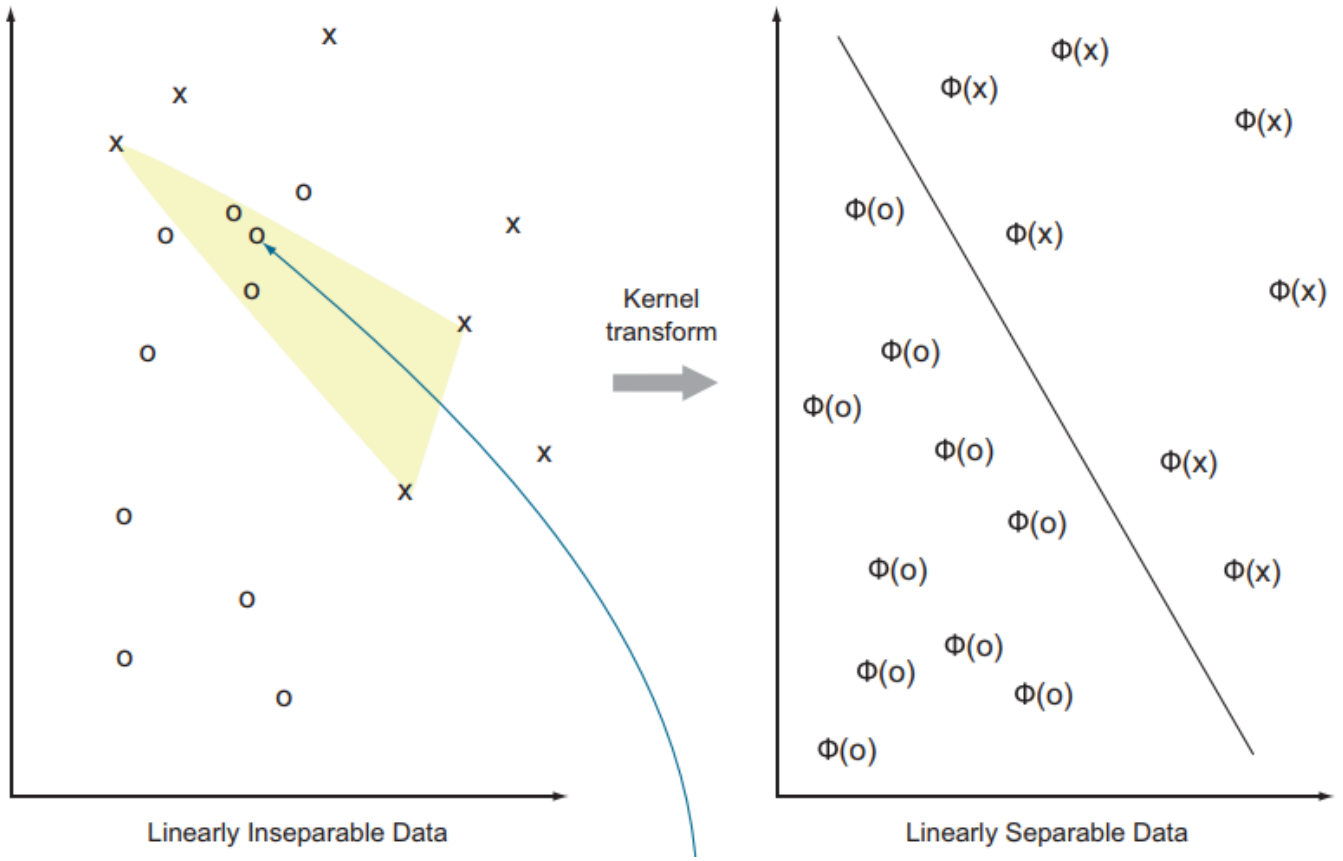
We know that linear regression can be applied to data that's not linear by using polynomial transformations on some features.

For example, instead of using cache size (to predict CPU speed) we use the square of the cache size.

In general we can transform our space of feature vectors into a new space where problems become easier.

Mapping vectors to a new space

Function ϕ transforms original feature vectors to new ones



source: Zumel and Mount, *Practical Data Science*

What is a kernel function?

For SVM we will need to be able to compute the dot product of vectors.

We don't need to explicitly transform feature vectors

Let u, v be feature vectors.

A function k is a *kernel* if there is some function ϕ that maps (u, v) to a new feature space, and such that

$$k(u, v) = \phi(u) \cdot \phi(v)$$

You never actually have to map the vectors to the new feature space.

The new feature space can even have infinitely many dimensions!

Summary

- Hard margin classifier
  - works only with linearly-separable data
  - finds widest "street" that separates the classes
- Soft margin classifier ("support vector classifier")
  - handles data that's not linearly separable
  - uses a tuning parameter C that represents the importance of margin violations