Data, Environment and Society: Lecture 20: Support Vector Machines

Instructor: Duncan Callaway
GSI: Salma Elmallah

November 7, 2019

DEPT. OF TECHNOLOGY NOVEMBER 12, 2018 ISSUE

IN THE AGE OF A.I., IS SEEING STILL BELIEVING?

Advances in digital imagery could deepen the fake-news crisis—or help us get out of it.

By Joshua Rothman

As synthetic media spreads, even real images will invite skepticism.

Illustration by Javier Jaén; photograph by Svetikd / Getty

Announcements

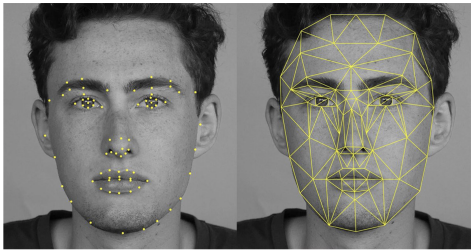
- I'll be teaching from ISLR Ch 9 today.
- HW9 due today!
- Next week lab: come with data and resource allocation question in hand.
- Thursday: Guest lectures from Diego Ponce de Leon and Grace Wu – come ready to ask questions!

Objectives for today

- Some examples
- Introduce the idea of a **hyperplane** (it's really simple)
- Figure out what a **maximal margin hyperplane** (MMH) is and why we use it
  - Note, these only work for *separable data*
- Understand how **support vector classifiers** extend the MMH to cases when the data are not separable.
  - SVCs are *linear* separations of the feature space
- Open your horizons to the **support vector machine**
  - This provides nonlinear separations of the feature space!

Motivating examples: Society

face recognition



character recognition,

Motivating examples – environment

Science of The Total Environment

Volume 395, Issues 2–3, 1 June 2008, Pages 109-116

Ground-level ozone prediction by support vector machine approach with a cost-sensitive classification scheme

Wei-Zhen Lu, Dong Wang

Motivating examples – environment

Forecasting Power Output of Photovoltaic Systems Based on Weather Classification and Support Vector Machines

Jie Shi, Wei-Jen Lee, *Fellow, IEEE*, Yongqian Liu, Yongping Yang, and Peng Wang

***Abstract*—Due to the growing demand on renewable energy, photovoltaic (PV) generation systems have increased considerably in recent years. However, the power output of PV systems is affected by different weather conditions. Accurate forecasting of PV power output is important for system reliability and promoting large-scale PV deployment. This paper proposes algorithms to forecast power output of PV systems based upon weather classification and support vector machines (SVM). In the process, the weather conditions are divided into four types which are clear sky, cloudy day, foggy day, and rainy day. In this paper, a one-day-ahead PV power output forecasting model for a single station is derived based on the weather forecasting data, actual historical power output data, and the principle of SVM. After applying it into a PV station in China (the capability is 20 kW), results show the proposed forecasting model for grid-connected PV systems is effective and promising.**

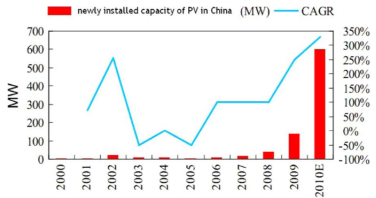
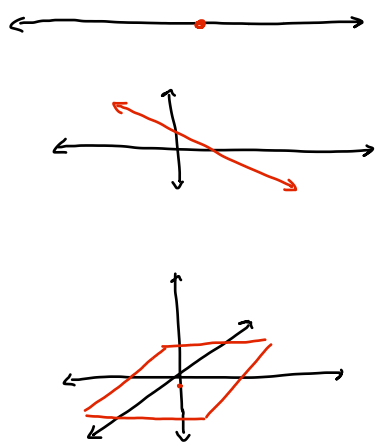


Fig. 1. Annual installation capacity of PV in China (2000–2010E, CAGR: Capacity Annual Growth Rate).

Motivating examples – environment

The hyperplane

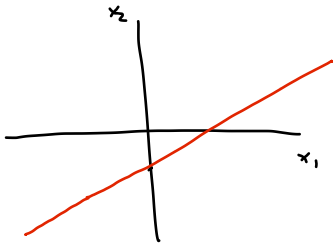
- A point splits a 1-dimensional space in two
- A line splits a 2-dimensional space in two
- A plane splits a 3-dimensional space in two
- A hyperplane splits a p-dimensional space in two.



Mathematical representation

If we have a line defined by

$$x_2 = -2 + \frac{1}{2}x_1$$



Then the equality defining the hyperplane is:

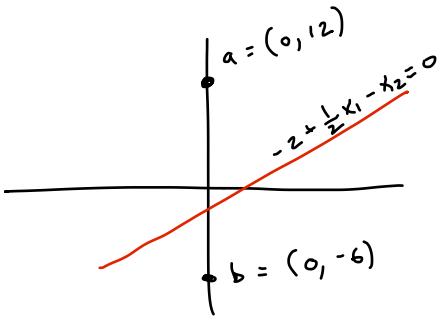
$$f(x) = -2 + \frac{1}{2}x_1 - x_2 = 0$$

- or -

$$f(x) = 2 - \frac{x_1}{2} + x_2 = 0$$

How we'll think about the hyperplane

- In a data set with p features...
- We'll conceive of it as a $p-1$ dimensional object
- ...and we'll identify the location of observations in relation to the hyperplane



On the hyperplane: $f(x) = 0$

Above the hyperplane: $f(a) = -14$

$\Rightarrow f(x) < 0$ above

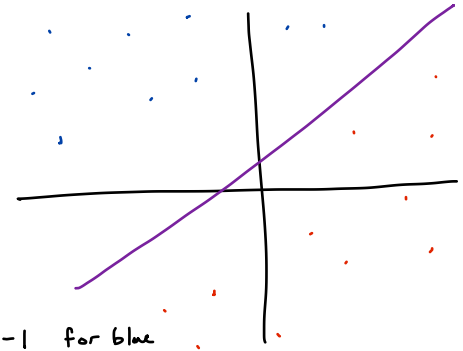
Below the hyperplane: $f(b) = 4$

$\Rightarrow f(x) > 0$ below

Theoretical example: Classification with hyperplanes

Suppose we have

- blue points
- red points



A "separating hyperplane" has the property that:

$f(x) < 0$ if blue

$f(x) > 0$ if red

Let $y_i = -1$ for blue

$y_i = 1$ for red

$\Rightarrow \quad y_i f(x_i) > 0 \quad \forall \; i$

Using the plane for predictions

This part is simple. If we have a test observation, we simply evaluate $f(x_{\text{test}})$ and assign it to a class on the basis of the sign of the result.

$f(x_{test}) > 0 \Rightarrow \hat{y}_{test} = 1 \Rightarrow$ red

$f(x_{test}) < 0 \Rightarrow$ blue

How to choose the location of the plane?

Let's pose this as a learning problem. We have data and we'd like to place the hyperplane in between the two classes.

- First: What does large $|f(x_i)|$ imply?

How to choose the location of the plane?

Let's pose this as a learning problem. We have data and we'd like to place the hyperplane in between the two classes.

- First: What does large $|f(x_i)|$ imply? $\rightarrow$ A point is far from the hyperplane.

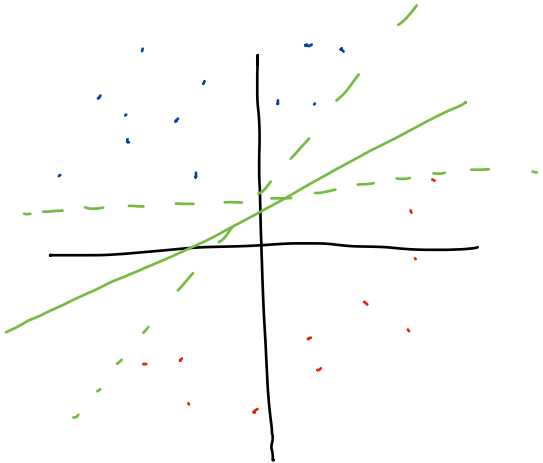
How to choose the location of the plane?

Let's pose this as a learning problem. We have data and we'd like to place the hyperplane in between the two classes.

- First: What does large $|f(x_i)|$ imply? $\rightarrow$ A point is far from the hyperplane.

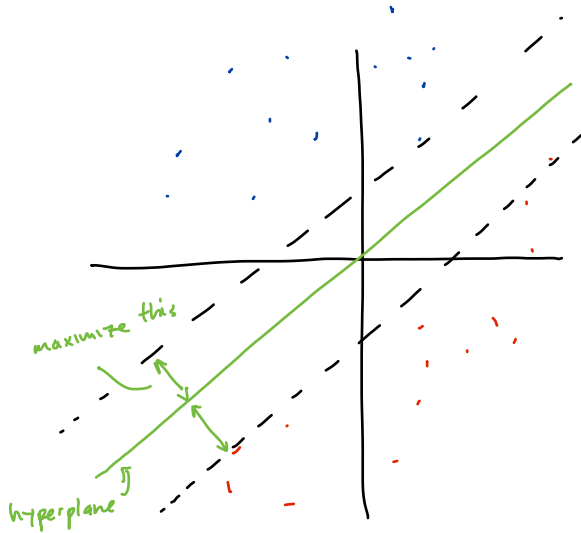
Now, how should we choose which plane to draw?

The rest of this lecture focuses on how to draw these separating geometries

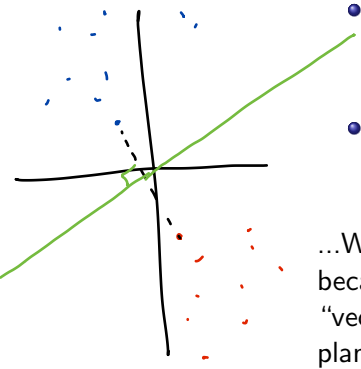


Maximal margin hyperplane (MMH)

- MMH defined: The MMH is the hyperplane that *maximizes* the *smallest distance* between the plane and all data.
  - In other words: it is the plane that is farthest from the data.
- Important: This requires that the data are *linearly separable*.



How many points define the hyperplane?



In two dimensions...

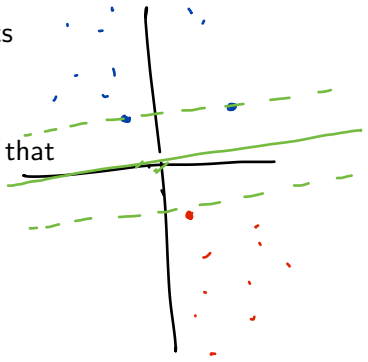
- What is the smallest number of points that could define the hyperplane?

  2

- What is the largest number of points that could define the hyperplane?

  3

...We call these points "support vectors" because each of the 2-3 observations is a "vector" of information that supports the plane.



Variance alert

In two dimensions, three observations determine the parameters of the model. Or more generally $p + 1$ parameters. *If* p is small, this leads to high variance across training data sets.

And now for the math

We'll solve for the location of the MMH using...

And now for the math

We'll solve for the location of the MMH using...you guessed it, optimization.

$$\max_{\beta_0, \beta_1 \cdots \beta_p} \quad M$$

subject to

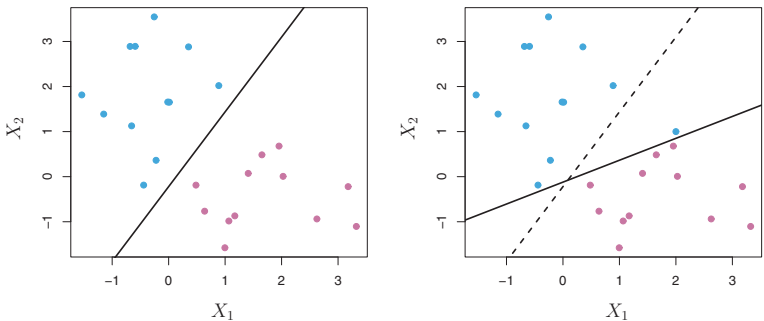
$$y_i \left(\beta_0 + \beta_1 x_{1i} + \cdots \beta_p x_{pi}\right) \geq M$$

$$\sum_{j=0}^{p} \beta_j^2 = 0$$

This constraint keeps β_j from "blowing up." Also makes hyperplane interpretable as distance.

Pause for a moment...

The location of the MMH is very sensitive to the support vectors:



Furthermore:

- Though the M objective in the MMH formulation *can* in principle be negative (which allows for non-separable data)...
- ...the problems of variance get worse as data become non-separable.

Support vector *classifiers*

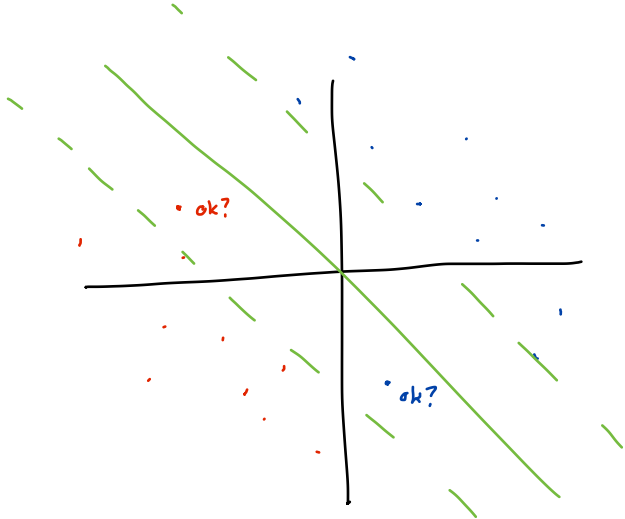
Can we get

- ...less sensitivity to individual observations, and
- ...better classification for most training data (at expense of some poor classifications)

The answer is yes – but we will

- Allow some training data to enter the "margin"
- ...and perhaps even be on the wrong side of the hyperplane.

Now we call the margin a "soft margin"



Support vector classifier details

We'll solve a slightly different optimization problem.

- Still classifies observations on the basis of what side of the hyperplane they're on
- But a few observations can be mis-classified.

The optimization problem is:

$$\max_{\beta_0, \beta_1, \ldots \beta_p, \epsilon_1, \epsilon_2 \ldots \epsilon_n} M$$

subject to

each observation gets its own!

$$y_i f(x_i) \geq M(1 - \epsilon_i)$$

$$\sum_{j=0}^{p} \beta_j^2 = 1$$

prevents really big M

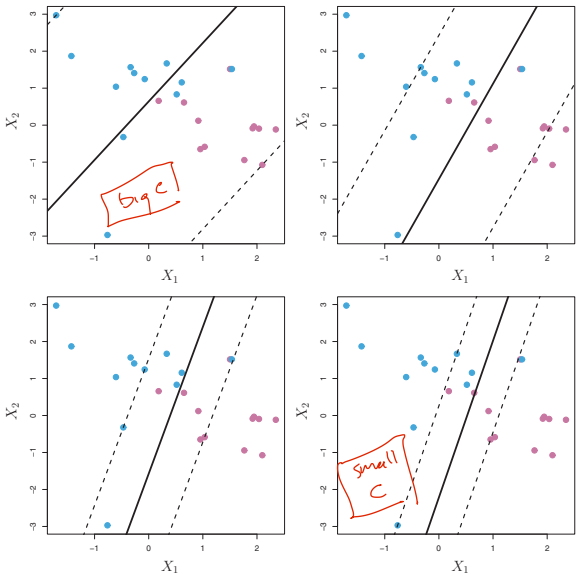
This is our budget. →

$$\sum_{i=1}^{n} \epsilon_i \leq C \quad , \quad \epsilon_i \geq 0 \ \forall \ i$$

What are the ϵ_i?

- Each *observation* gets its own ϵ_i.
- The ϵ are decision variables – the optimization problem will solve for them.
- Without C, we'd just give everyone an ϵ and go on a misclassification binge.
- But with C, the ϵ values are chosen within a budget.
  - $\epsilon_i = 0$, no margin violation
  - $0 < \epsilon_i < 1$, in the margin but on the correct side of the plane
  - $\epsilon_i > 1$, on the wrong side of the plane.

Tuning C

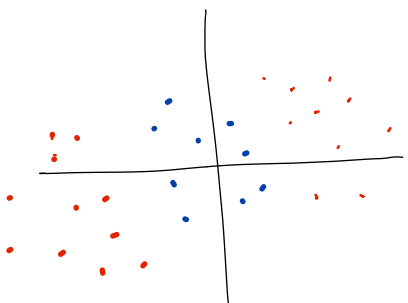


Questions

1. Which plot has large C? Which is small?
2. What's going to have the highest variance? Large or small C?

But decision boundaries aren't always so simple...

What if the boundary looked like this?



We might get better performance if we replaced the constraint in the optimization problem:

$$f'(x_i) = \beta_0 + \sum_{j=1}^{p} \beta_{1j} x_{ji} + \sum_{j=1}^{p} \beta_{2j} x_{ji}^2$$

$$y_i f'(x_i) \geq M(1 - \epsilon_i)$$

Setup for SVM "Kernels": Linear boundary

Let $\langle x_i, x_k \rangle = \sum_{j=1}^{p} x_{ij} x_{kj}$ "dot product"

One can show that:

$$\beta_0 + \sum_{j=1}^{p} \beta_j x_{ij} \Leftrightarrow \beta_0 + \sum_{k=1}^{n} \alpha_k \langle x_i, x_k \rangle$$

$$f(x_i) = \beta_0 + \sum_{k \in \mathbb{S}} \alpha_k \langle x_i, x_k \rangle$$

What's the relationship between α and β?

$$\sum_{k=1}^{n} \alpha_k \sum_{j=1}^{p} x_{ij} x_{kj}$$

$$= \sum_{j=1}^{p} x_{ij} \sum_{k=1}^{n} \alpha_k x_{kj}$$

$$\beta_j = \sum \alpha_k x_{kj} \Rightarrow \sum_{j=1}^{p} x_{ij} \beta_j$$

SVM Kernels

$$K_{\text{linear}}(x_i, x_k) = \langle x_i, x_k \rangle = \sum_{j=1}^{p} x_{ij} x_{kj}$$

$$K_{\text{polynomial}}(x_i, x_k) = (1 + \sum_{j=1}^{p} x_{ij} x_{kj})^d$$

$$K_{\text{radial}}(x_i, x_k) = \exp(-\gamma \sum_{j=1}^{p} (x_{ij} - x_{kj})^2)$$

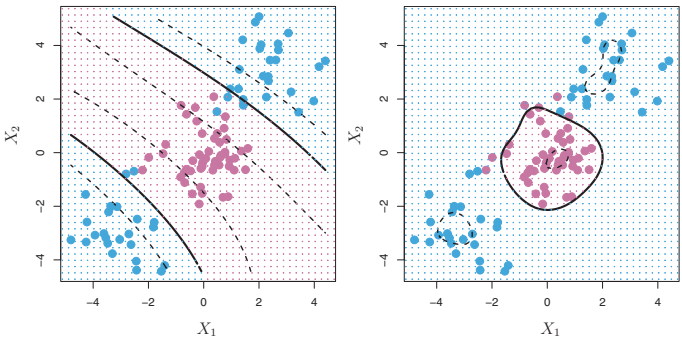
Then we can apply these kernels:

$$f(x_i) = \beta_0 + \sum_{k=1}^{p} \alpha_k K(x_i, x_k)$$

The radial kernel serves to give training observations far from a test point less weight (due to negative exponential).

For the radial kernel, all the data comprise the model – analogous to KNN.

What do the kernels look like?



Left: Polynomial, $d = 3$.
Right: Radial.

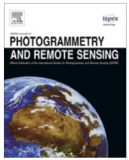
Note that d and γ are parameters to tune (via cross validation!)

SVM Example

Contents lists available at SciVerse ScienceDirect

ISPRS Journal of Photogrammetry and Remote Sensing

journal homepage: www.elsevier.com/locate/isprsjprs



Comparison of support vector machine, neural network, and CART algorithms for the land-cover classification using limited training data points

Yang Shao [a,*], Ross S. Lunetta [b]

[a] US Environmental Protection Agency, National Research Council, National Exposure Research Laboratory, 109 T.W. Alexander Drive, Research Triangle Park, NC 27711, USA
[b] US Environmental Protection Agency, National Exposure Research Laboratory, 109, T.W. Alexander Drive, Research Triangle Park, NC 27711, USA

Shao and Lunetta setup

Loads of remote sensing data (MODIS):

- 46 input features for each 250 m^2 pixel
  - 23 short wave infrared (SWIR) surface reflectance
  - 23 Enhanced Vegetation Index metrics – basically a summary of the wavelengths
- Training data from National Land Cover Dataset (NLCD), classifying land as
  - urban,
  - deciduous forest,
  - evergreen forest,
  - agricultural land, and
  - wetland
- Question: How do SVM, Neural networks (coming soon!), and regression trees perform relative to one another?

Shao and Lunetta SVM Result

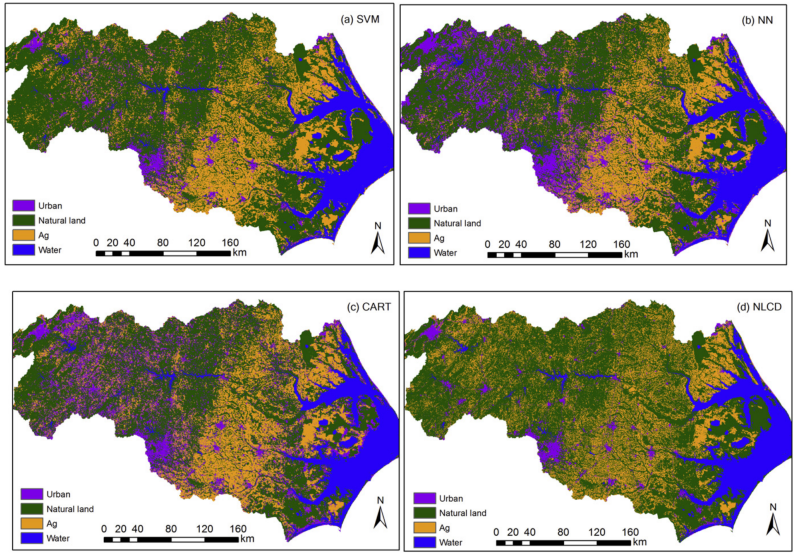
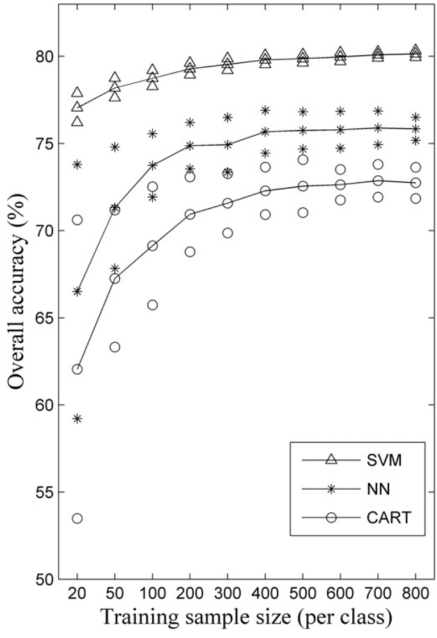


Fig. 3. Comparison of classification results for SVM (a), NN (b), and CART (c) algorithms. The NLCD 2001 (d) is also included as reference.


Overall accuracy (%)
80
75
70
65
60
55
50
20
50
100
200
300
400
500
600
700
800
Training sample size (per class)
SVM
NN
CART

Textbook example: Heart Data

First: Receiver operating characteristic (ROC)

Sensitivity = the fraction of "positives" that are correctly identified as positives

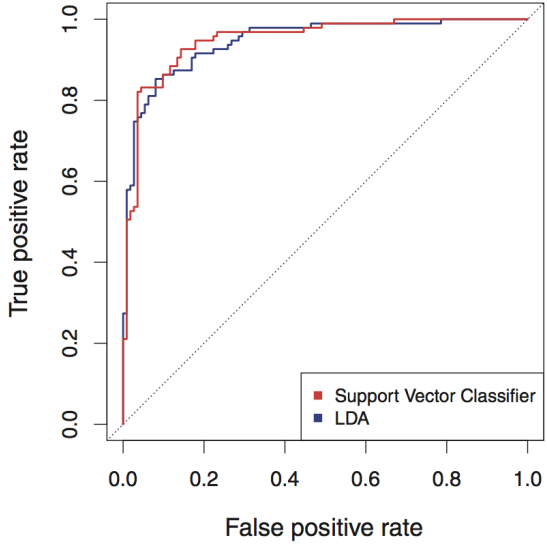
Specificity = fraction of negatives that we correctly identify as negatives

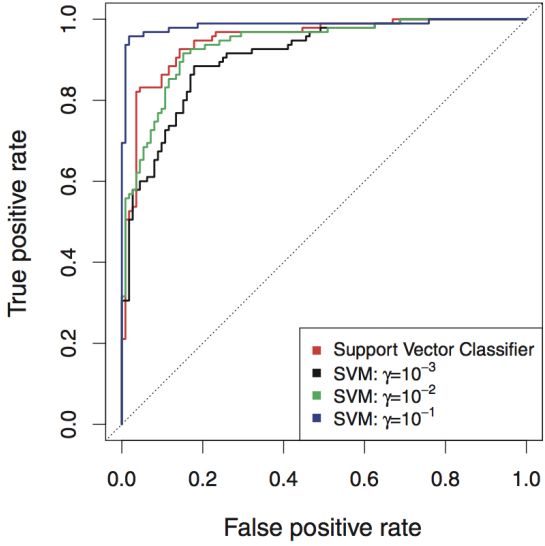
True positive rate = sensitivity

False positive rate = $1-$ specificity
Ideal classifiers have large true positive rate and low false positive.

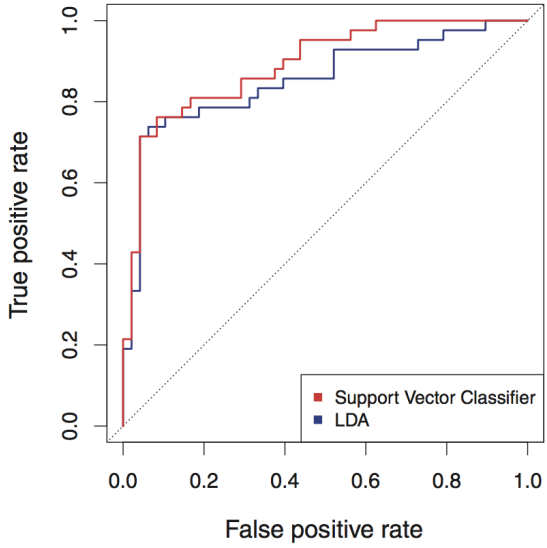
For a given method, there are usually parameters one can tune to explore the tradeoff between true and false positives.

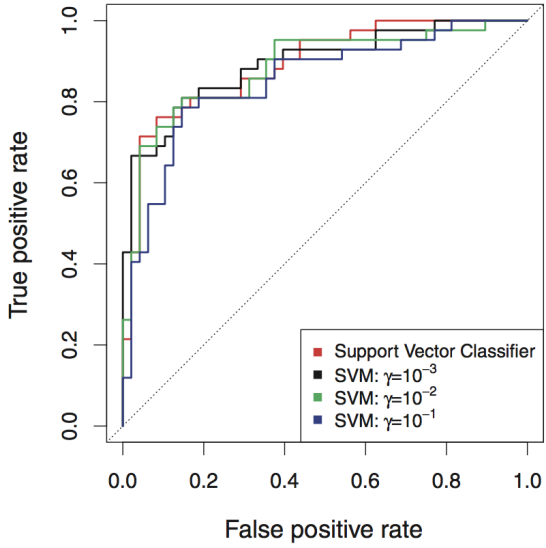
Classifying heart disease





Classifying heart disease – test data





Interpreting Heart data result

SVC is better than LDA, though surprisingly not by much if one recalls how simple the LDA approach is.

On training data, SVM with radial kernels are exceptional

...But not so much on test data. Why?

Interpreting Heart data result

SVC is better than LDA, though surprisingly not by much if one recalls how simple the LDA approach is.

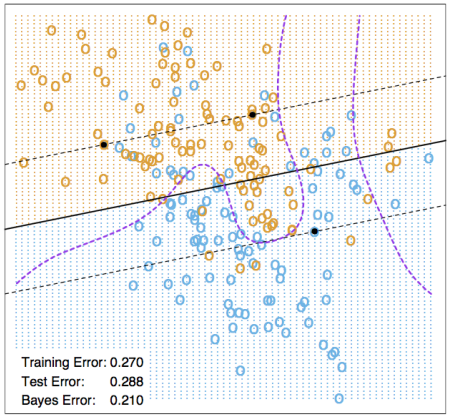
On training data, SVM with radial kernels are exceptional

...But not so much on test data. Why?

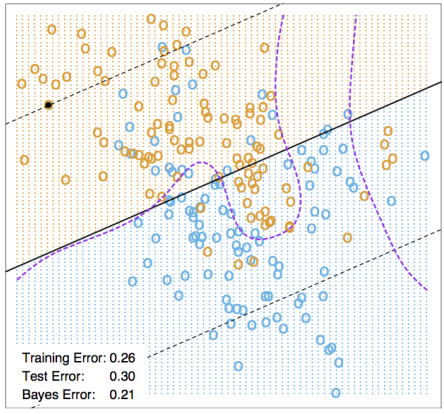
The decision boundary is really "wiggly" and prone to overfit.

It appears that SVC would have lower variance / higher bias and they balance perfectly in this particular case.

Speaking of wiggly boundaries: From Elements of Statistical Learning



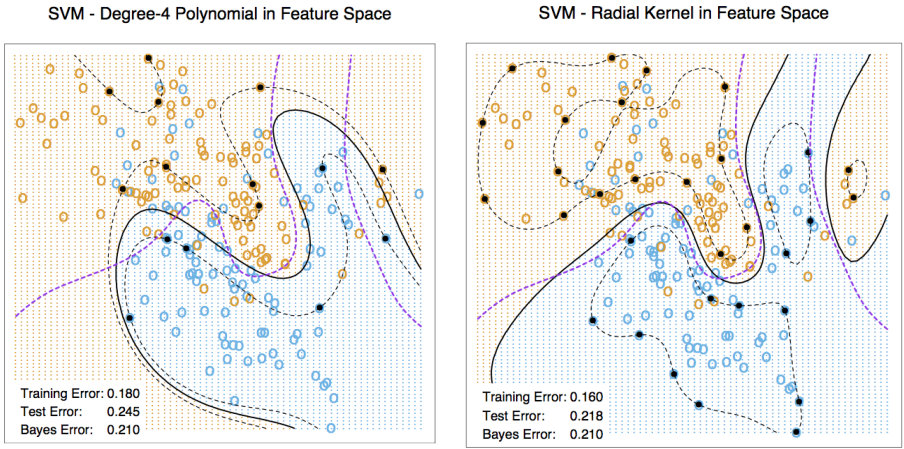
$C = 10000$



$C = 0.01$

These boundaries constructed with SVC. Purple is "truth" (what they used to generate the data). Note! In ESL C is the inverse of C in ISLR. So large C here corresponds to small C there, and vice versa.

One more example: From Elements of Statistical Learning



These boundaries constructed with SVM. Purple is "truth" (what they used to generate the data).