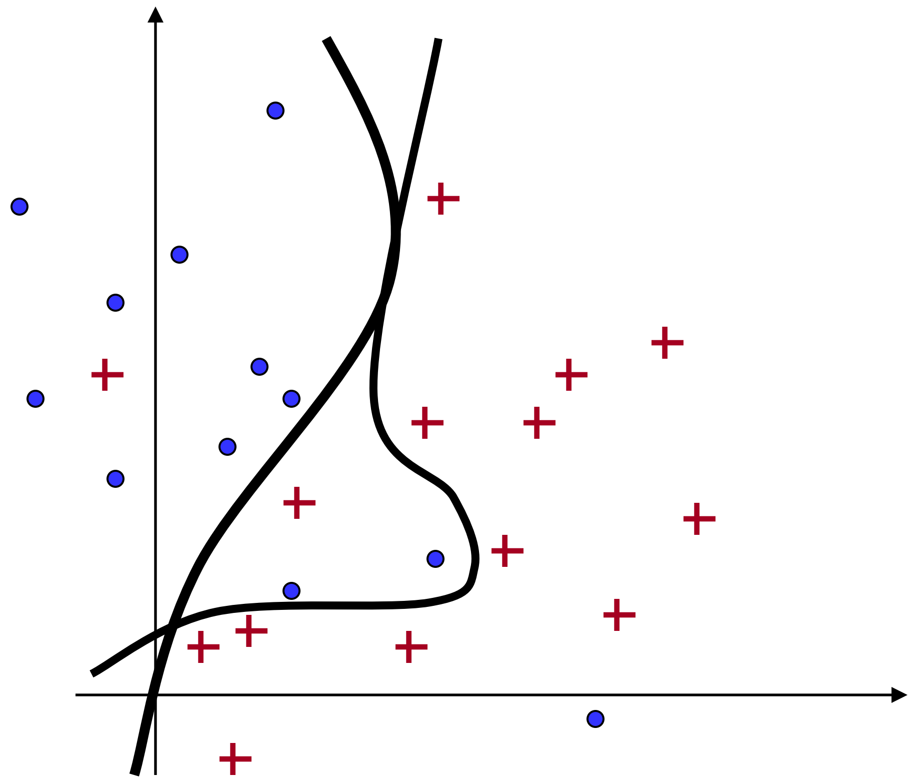
Controlling False Alarms with Support Vector Machines

Mark Davenport *Richard Baraniuk*
Clayton Scott

Rice University
dsp.rice.edu

The Classification Problem

Given some training data . . .



. . . find a classifier that *generalizes*

Conventional Classification

Signal / Pattern: $X \in R^d$

Label: $Y \in \{-1, +1\}$

Classifier: $f : R^d \rightarrow \{-1, +1\}$

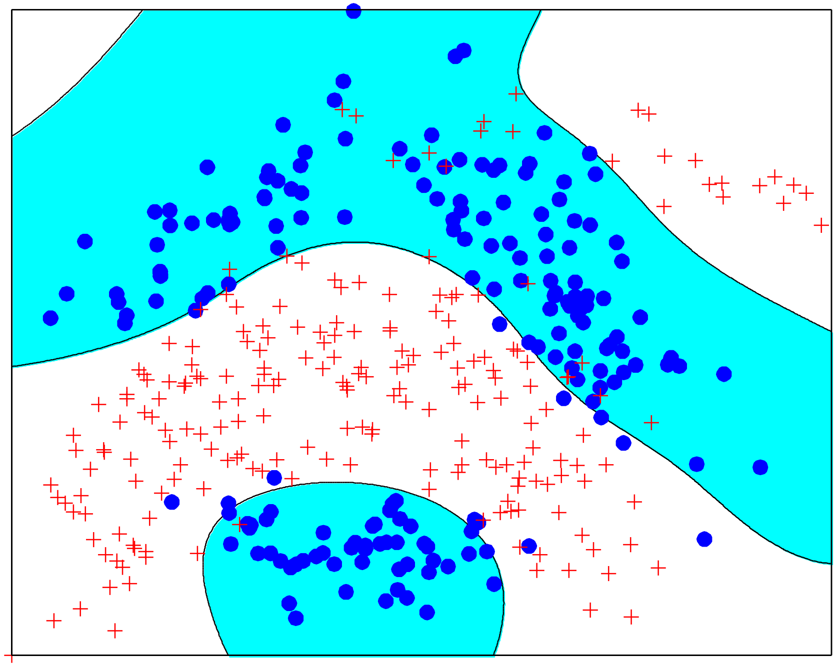
Probability of error: $P_E(f) := \text{Prob}(f(X) \neq Y)$

Goal: $$\boxed{f^* := \arg\min_f \; P_E(f)}$$

A Practical Approach

- Support Vector Machines (SVMs) offer a practical, nonparametric method for learning from data

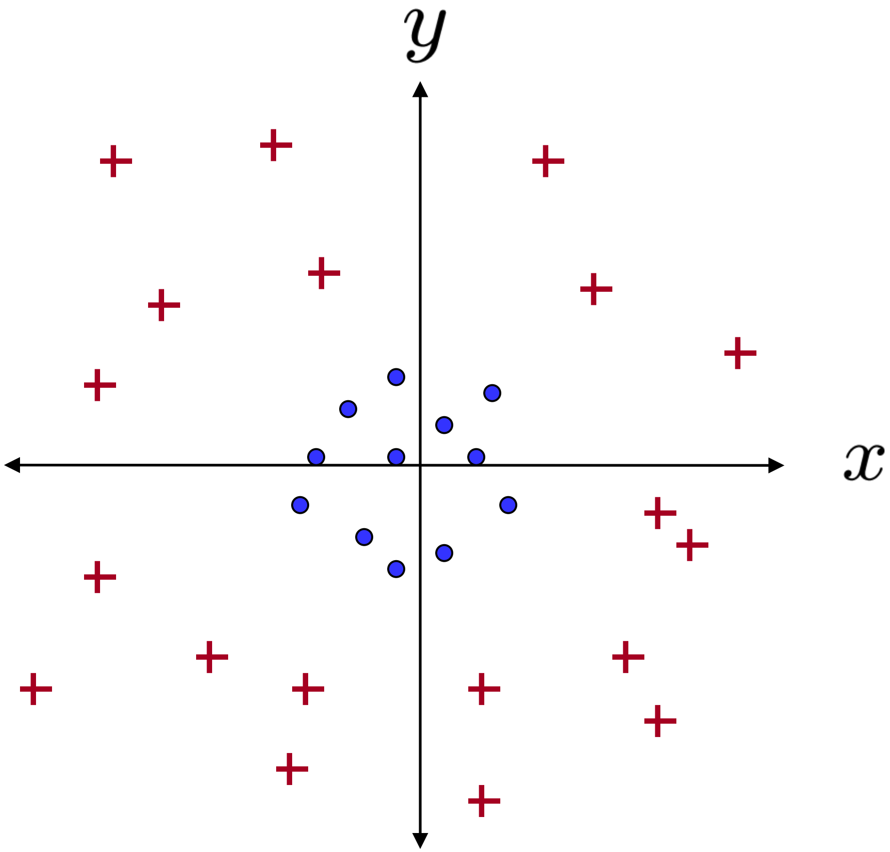
- General idea:
  - use “kernel trick”
  - hyperplane classifiers
  - maximize the *margin*



[Cortes, Vapnik (1995)]

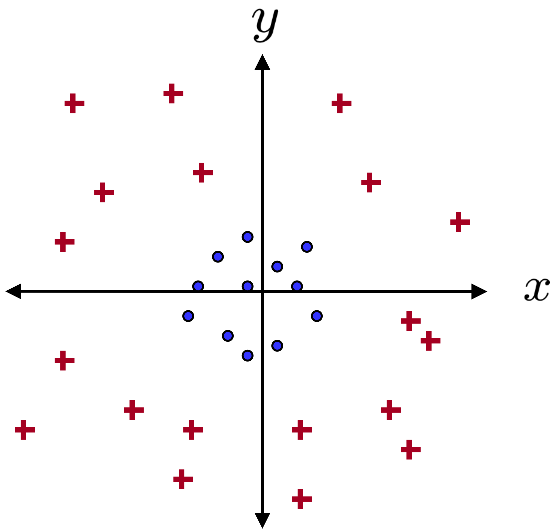
The “Kernel Trick”

- Problem: Linear classifiers perform poorly



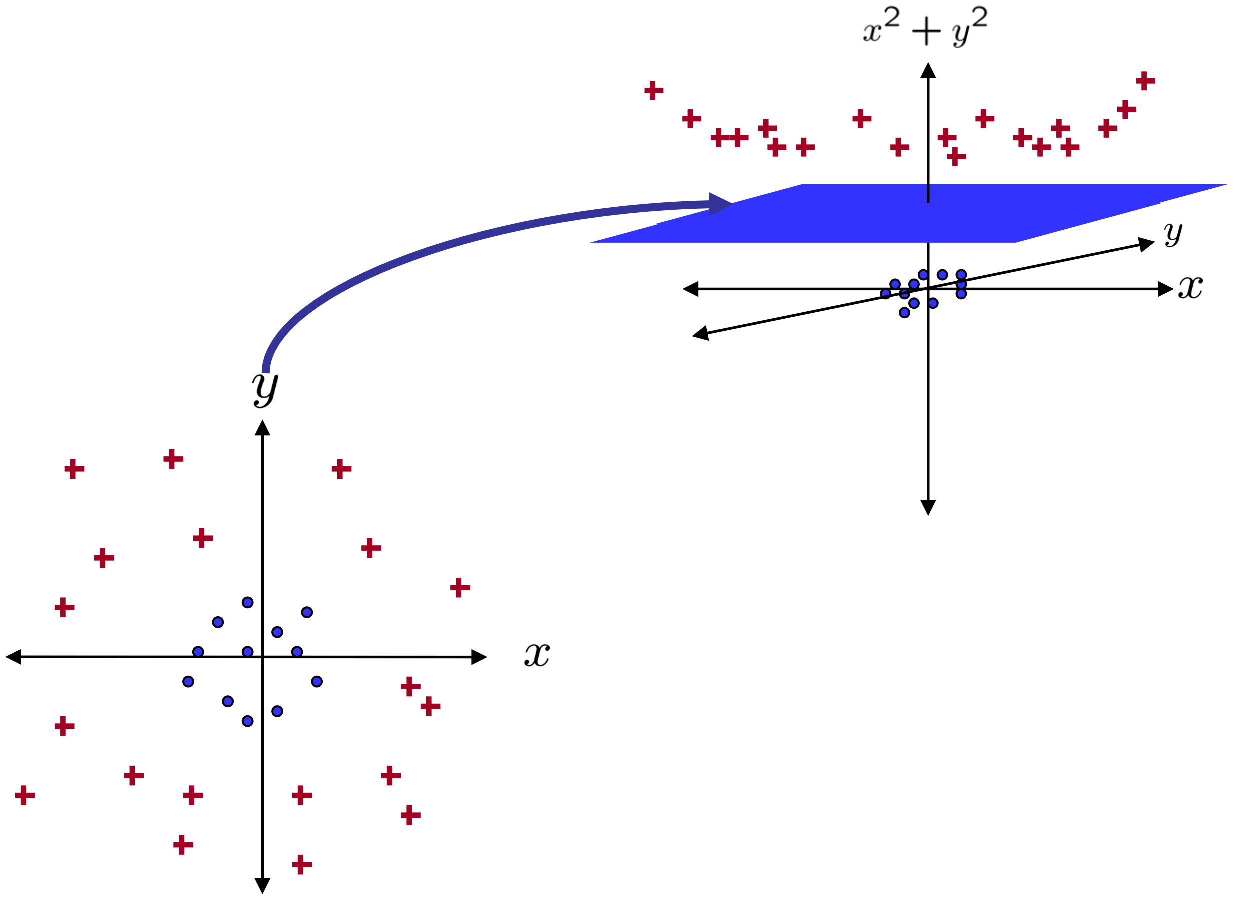
The "Kernel Trick"

- Problem: Linear classifiers perform poorly
- Solution: Map data into *feature space*



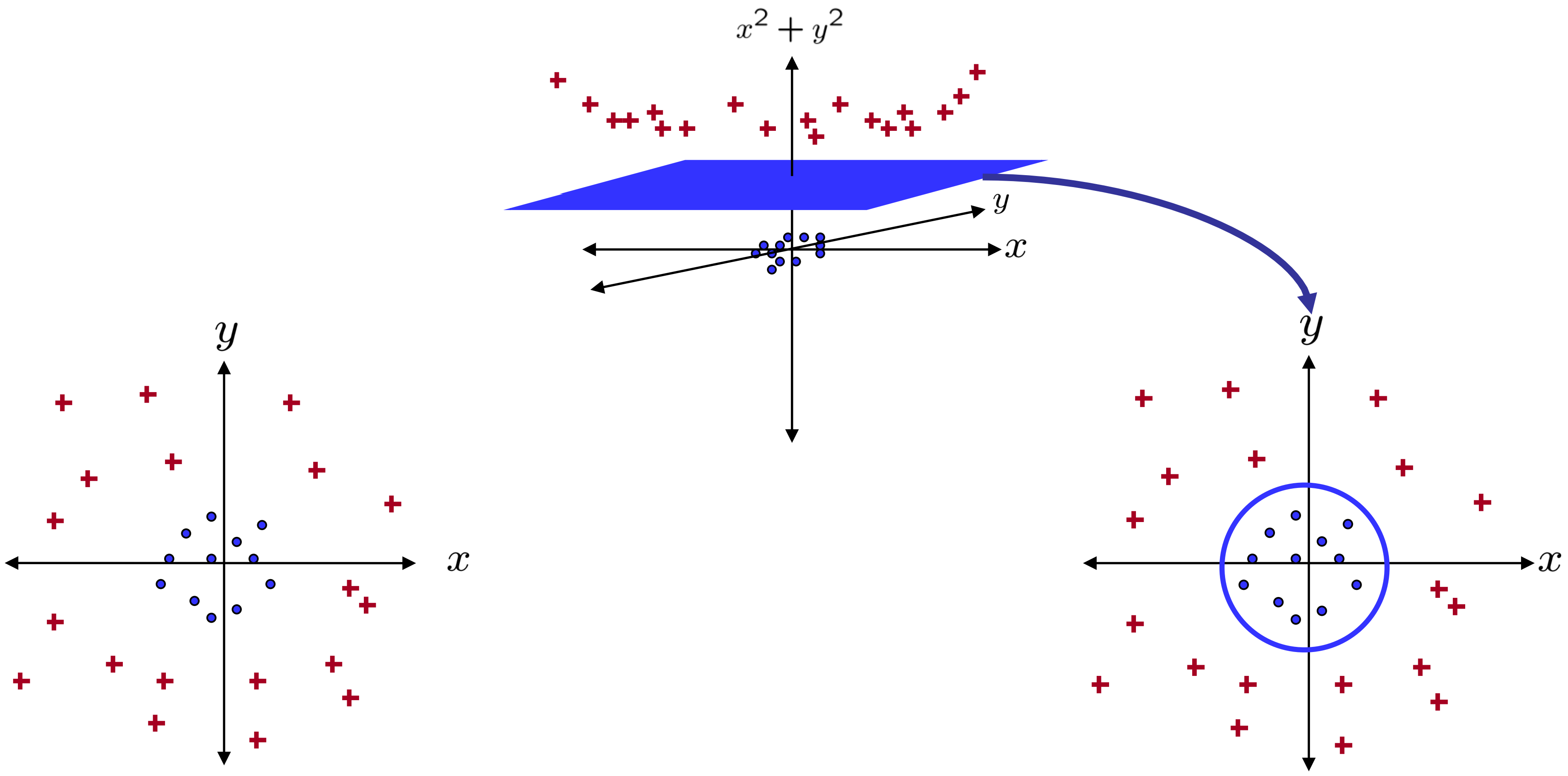
The "Kernel Trick"

- Problem: Linear classifiers perform poorly
- Solution: Map data into *feature space*



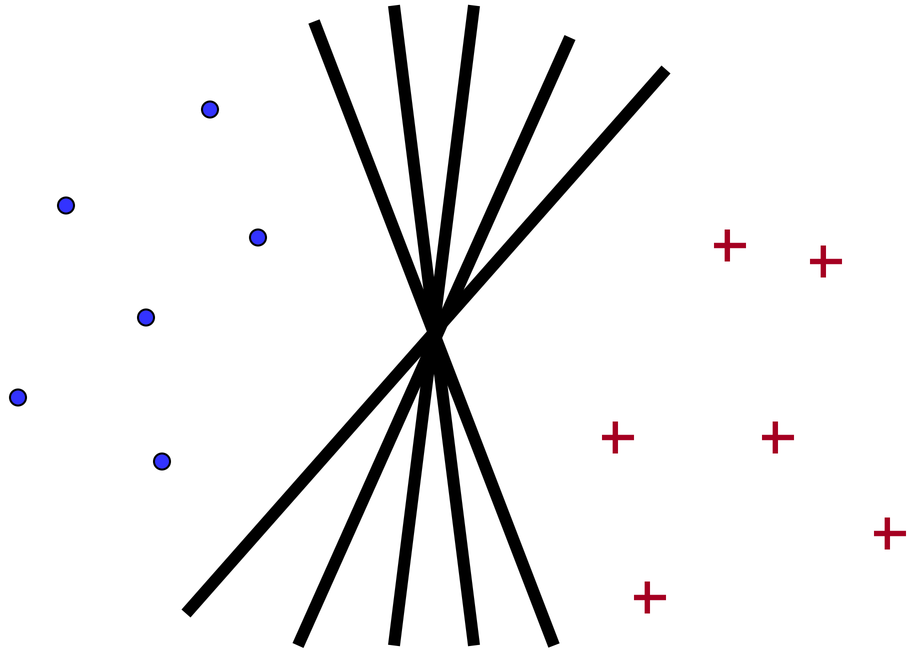
The "Kernel Trick"

- Problem: Linear classifiers perform poorly
- Solution: Map data into *feature space*



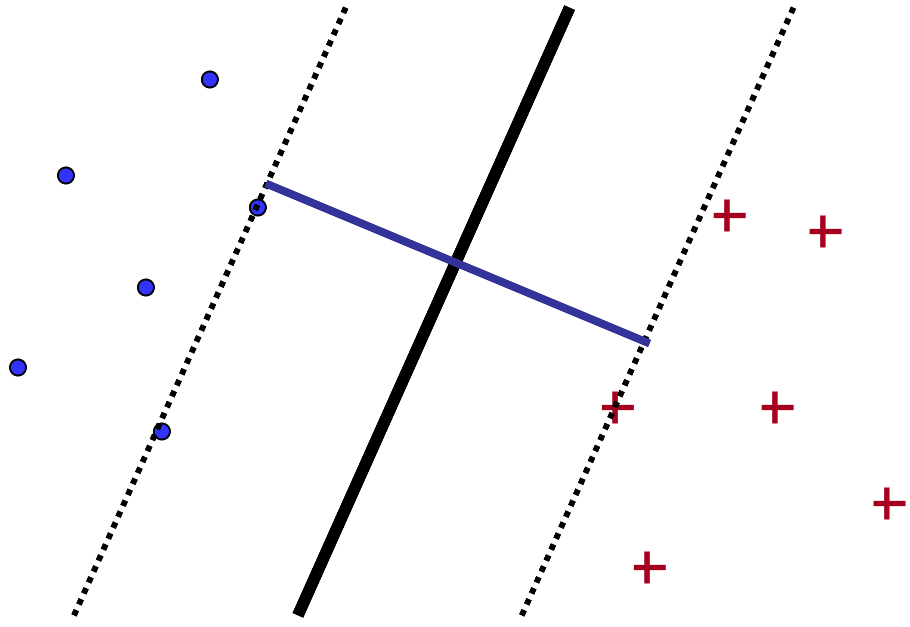
Maximum Margin Principle

- Problem: Many classifiers to choose from



Maximum Margin Principle

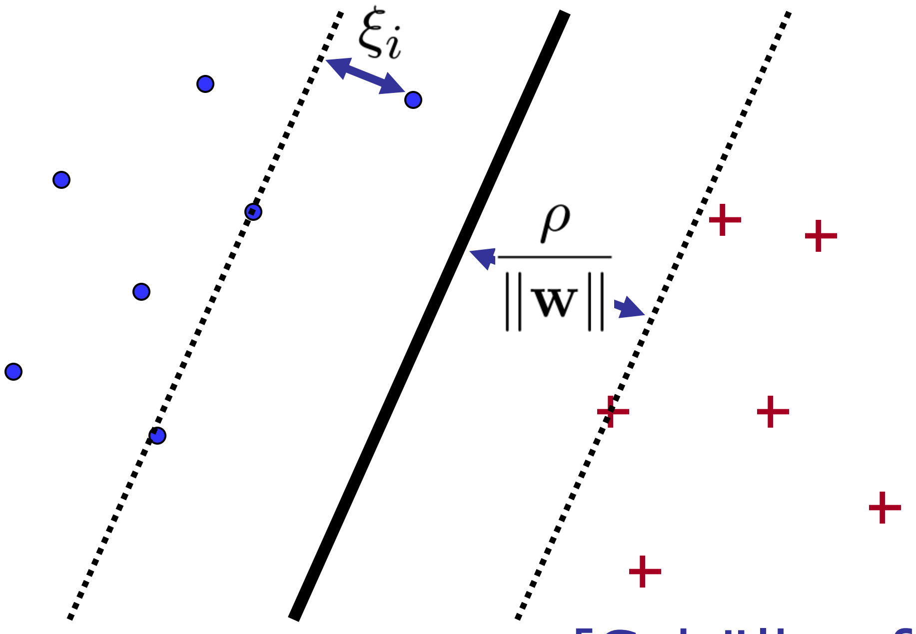
- Problem: Many classifiers to choose from
- Solution: Pick one that *maximizes margin*



ν–SVM

$$\min_{\mathbf{w},b,\xi,\rho} \quad \tfrac{1}{2}\|\mathbf{w}\|^2 - \nu\rho + \frac{1}{n}\sum_{i=1}^{n}\xi_i \qquad \nu \in [0,1]$$

$$\text{s.t.} \quad (\langle \mathbf{w}, \mathbf{X_i}\rangle + b)Y_i \geq \rho - \xi_i$$



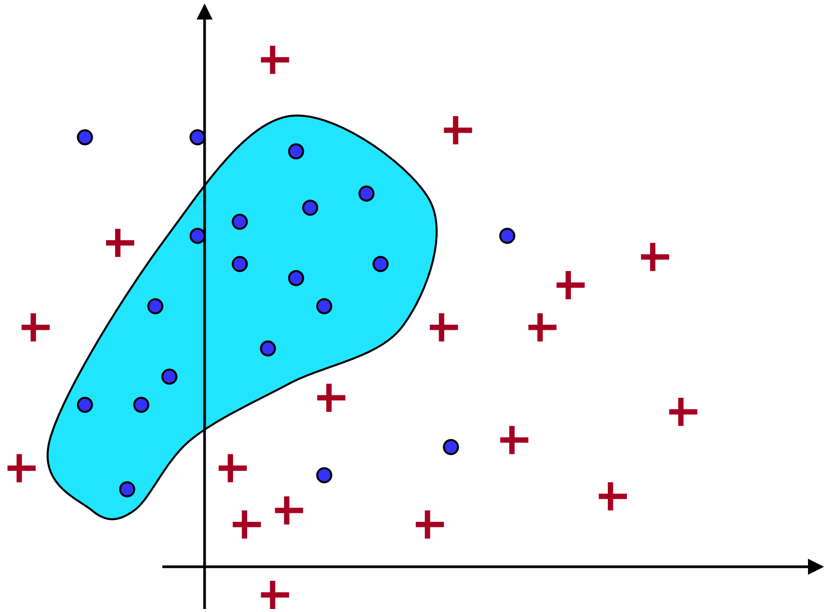
[Schölkopf et. al. (2000)]

What’s Wrong?

- Sometimes false alarms are more/less important than misses

False alarm: object detected, but not present

Miss: object present, but not detected

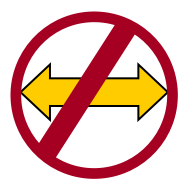


What Else?

- Class frequencies are often not represented in the training data
  - minimizing P_E can ignore smaller class

- Prior probabilities are usually unknown

- 100 training samples
- 50 have leukemia
- 50 do not



50% of population has leukemia

Neyman-Pearson Classification

- Solution: Recast the problem

False alarm: $P_F(f) := \text{Prob}(f(X) = +1 | Y = -1)$

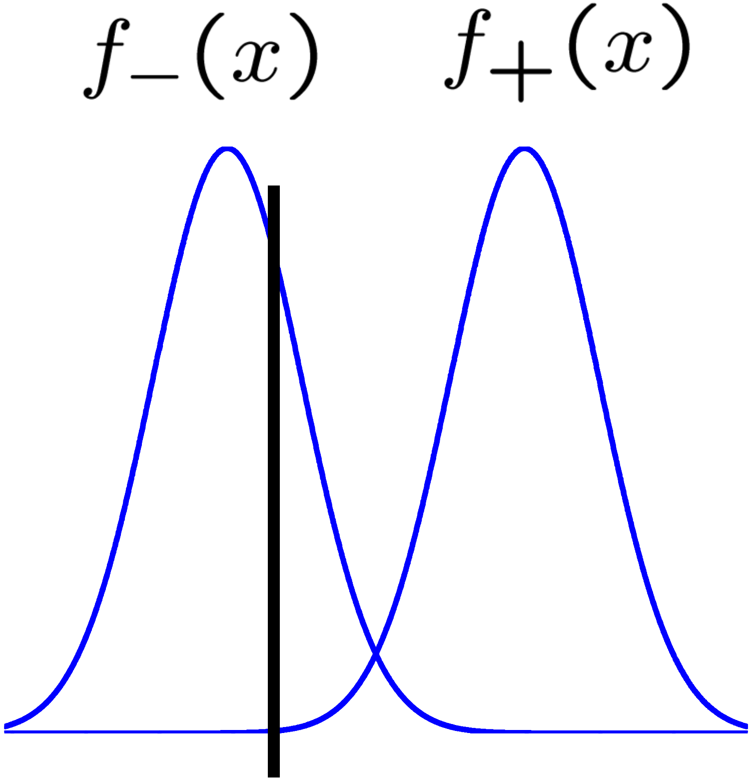
Miss: $P_M(f) := \text{Prob}(f(X) = -1 | Y = +1)$

- Goal:

$$\boxed{\begin{aligned} f^*_\alpha := \arg\min_f \; & P_M(f) \\ \text{s.t. } & P_F(f) \leq \alpha \end{aligned}}$$

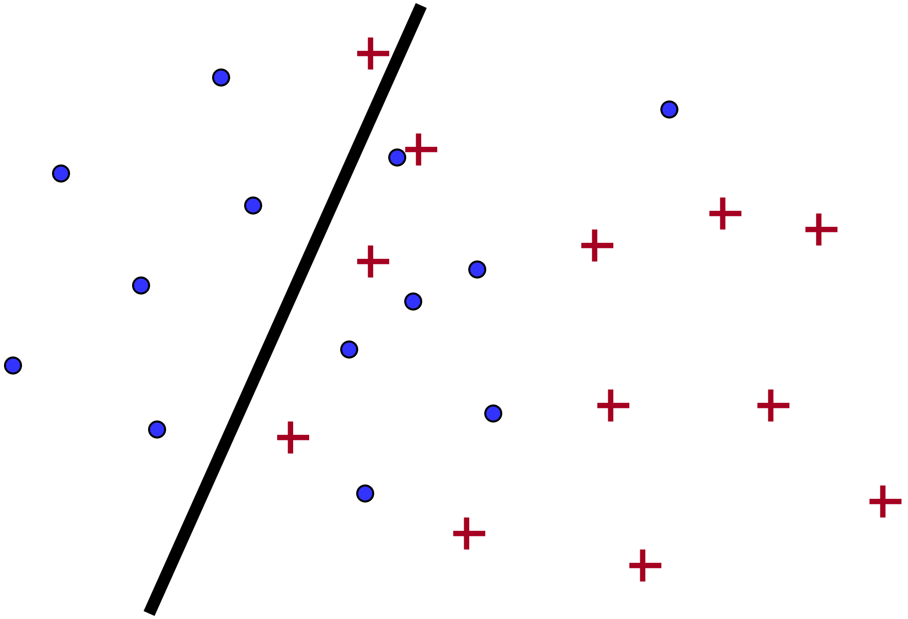
Simple Approach

- Bias-shifting
  - ad-hoc, but oft-used



Simple Approach

- Bias-shifting
  - ad-hoc, but oft-used



Cost-Sensitive SVMs

- We need to explicitly treat the classes differently during training

$$\min_{\mathbf{w},b,\xi,\rho} \quad \tfrac{1}{2}\|\mathbf{w}\|^2 - 2\nu_-\nu_+\rho + \frac{\nu_-}{n_+}\sum_{i\in I_+}\xi_i + \frac{\nu_+}{n_-}\sum_{i\in I_-}\xi_i$$

2ν–SVM

$$\text{s.t.} \quad (\langle \mathbf{w}, \mathbf{X_i}\rangle + b)Y_i \geq \rho - \xi_i$$

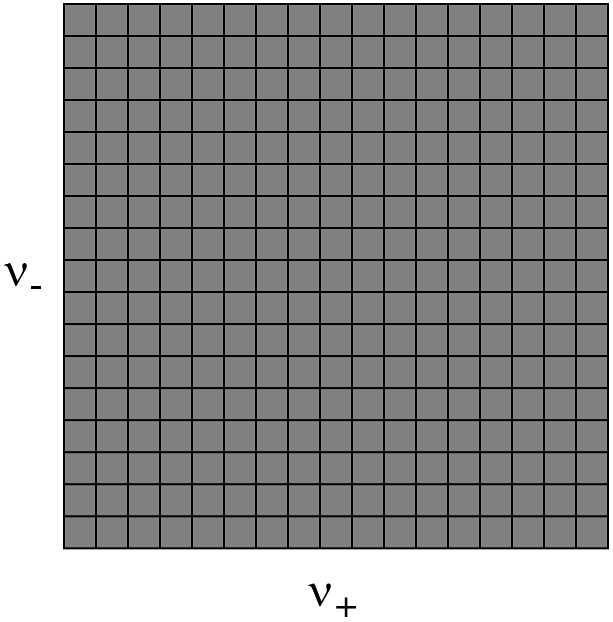
$$(\nu_+, \nu_-) \in [0,1]^2$$

- Equivalent to the 2C-SVM
- How to pick ν_+ and ν_- ?

[Chew, Bogner (2001)]
[Davenport (2005)]

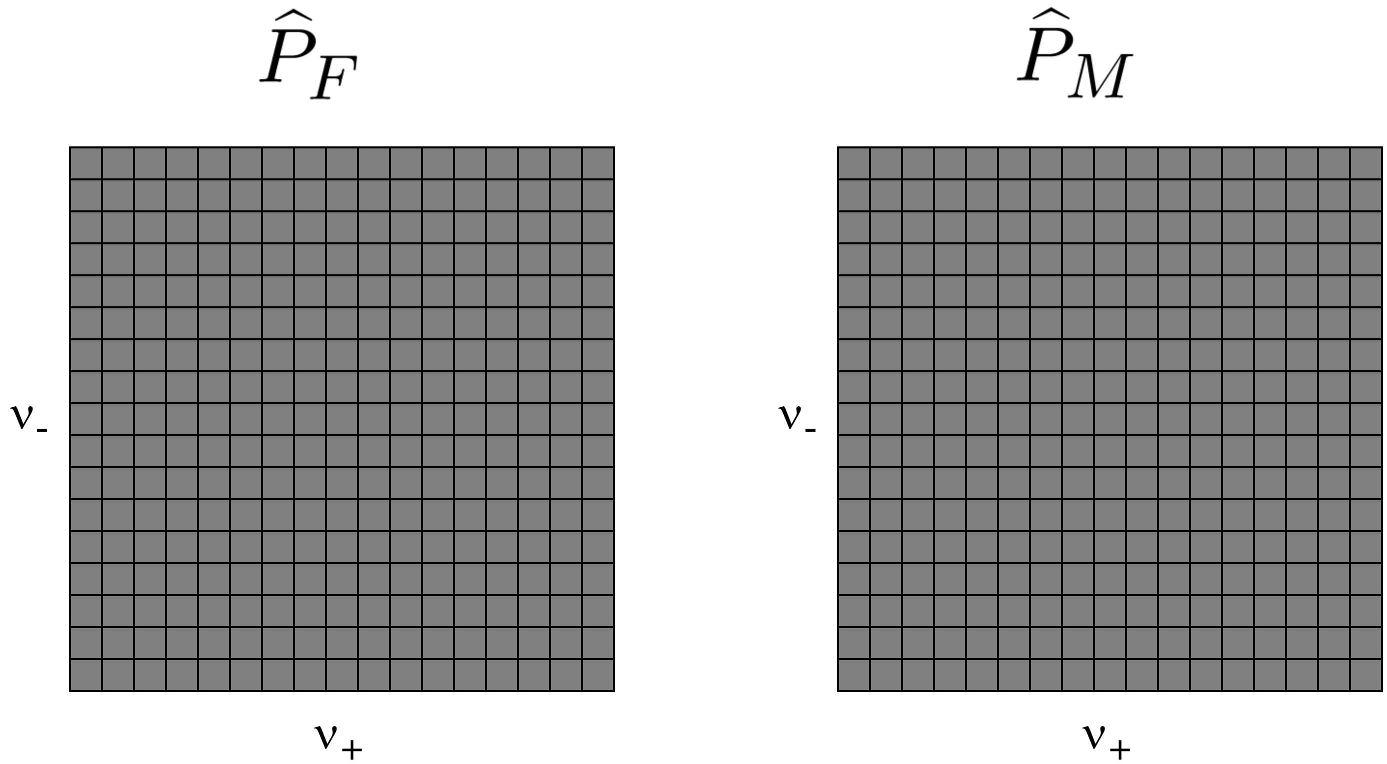
Controlling False Alarms: 2ν-SVM

- Perform grid search over parameters



Controlling False Alarms: 2ν-SVM

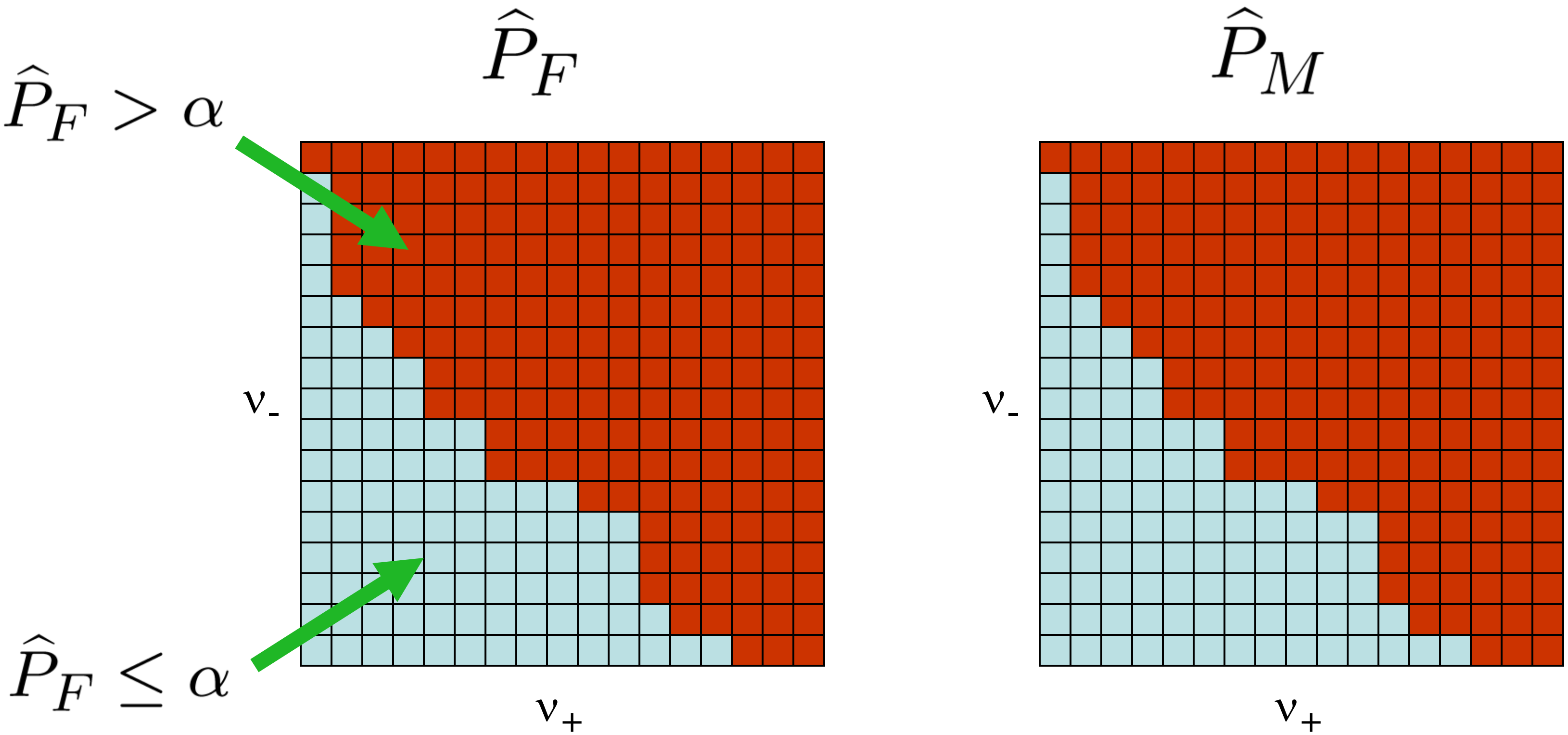
- Perform grid search over parameters
- Estimate false alarm and miss rates



Controlling False Alarms: 2ν-SVM

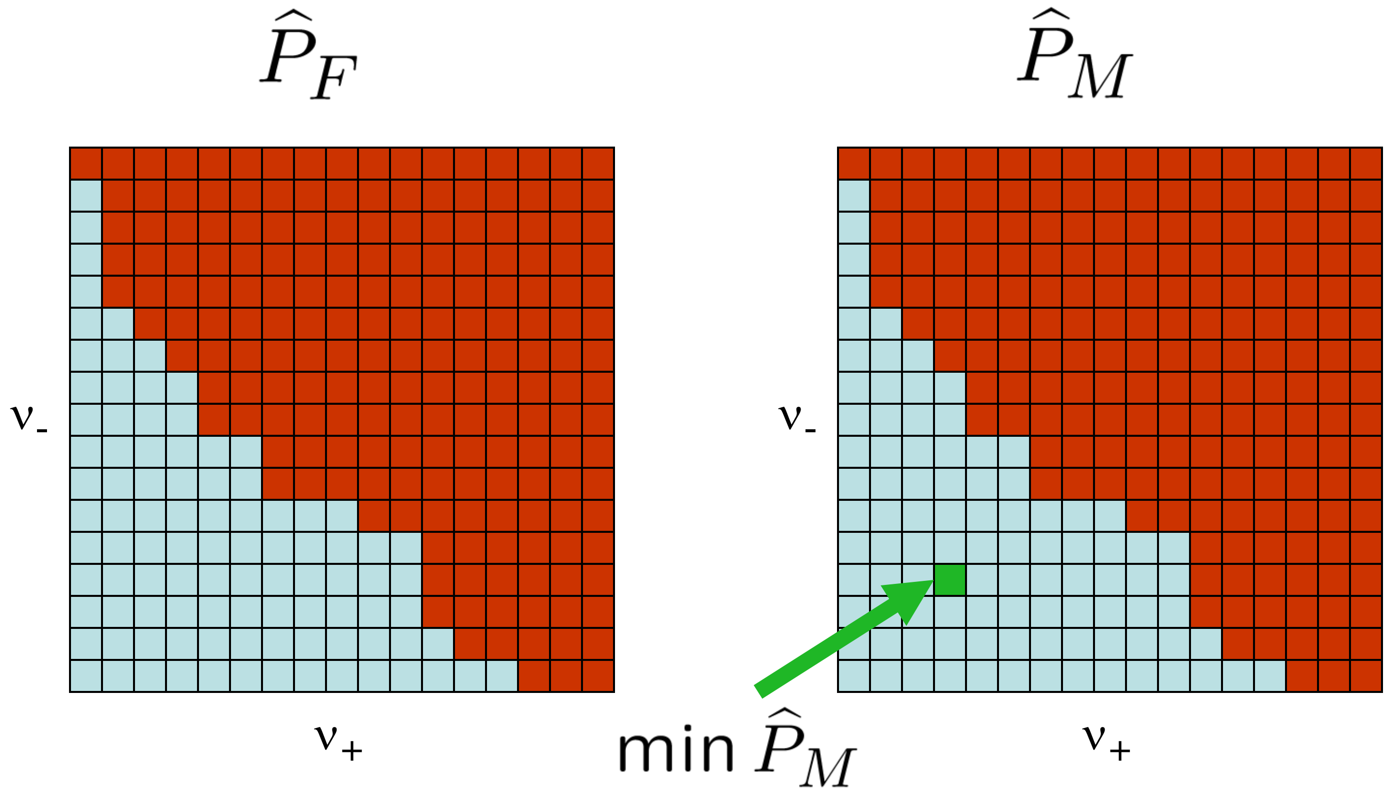
- Perform grid search over parameters
- Estimate false alarm and miss rates
- Set $(\nu_+^*, \nu_-^*, \sigma) := \arg\min \ \widehat{P}_M$

  $\text{s.t. } \widehat{P}_F \leq \alpha$

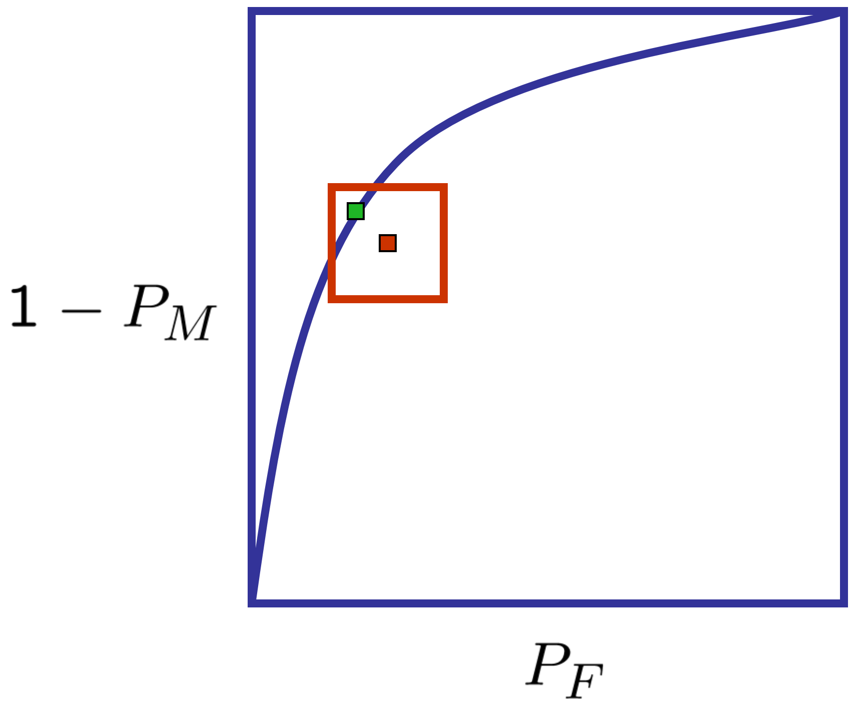


Controlling False Alarms: 2ν-SVM

- Perform grid search over parameters
- Estimate false alarm and miss rates
- Set $(\nu_+^*, \nu_-^*, \sigma) := \arg\min \; \widehat{P}_M$
  $\text{s.t.} \; \widehat{P}_F \leq \alpha$



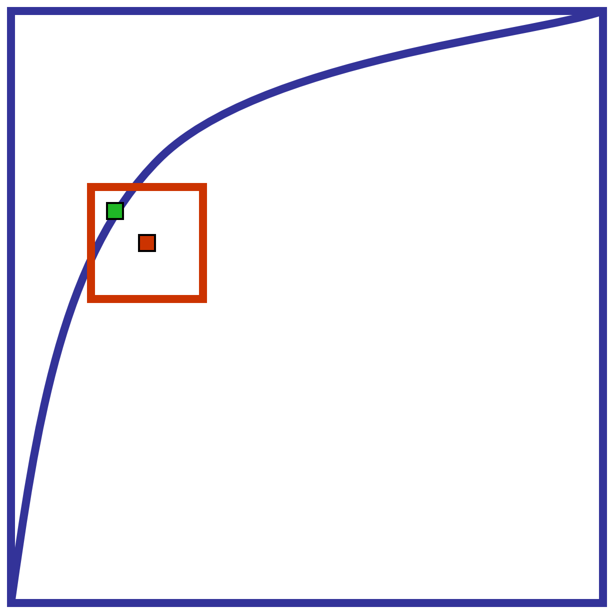
Performance Evaluation



We need a scalar measure of performance

- we want to evaluate our ability to achieve a *specific point* on the ROC

Performance Evaluation



$$\mathcal{E}(f) := \frac{1}{\alpha}\max\{P_F(f) - \alpha, 0\} + P_M(f)$$

- Theorem:

  f^*_α is the unique global minimizer of $\mathcal{E}(f)$

[Scott (2005)]

Experimental Results

- Use Gaussian kernel
- Performance averaged over 100 permutations
- 4 benchmark datasets
- We report
  - mean P_F, P_M
  - median $\mathcal{E}$
- 2ν-SVM clear winner

		P_F	P_M	$\mathcal{E}$
thyroid	BS	.06	.46	.637
	2ν-SVM	.09	.04	.051
heart	BS	.09	.55	1.000
	2ν-SVM	.11	.23	.326
cancer	BS	.00	1.00	1.000
	2ν-SVM	.11	.69	.821
banana	BS	.11	.33	.628
	2ν-SVM	.10	.12	.160

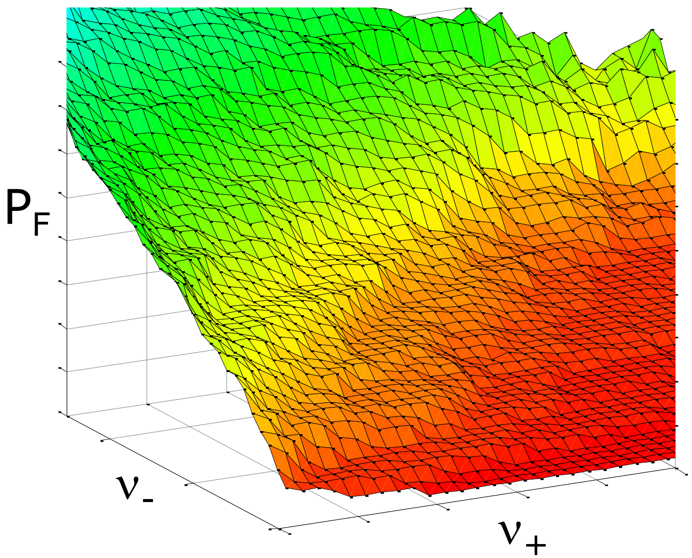
$\alpha = 0.1$

BS: bias-shifting

2ν-SVM: our approach

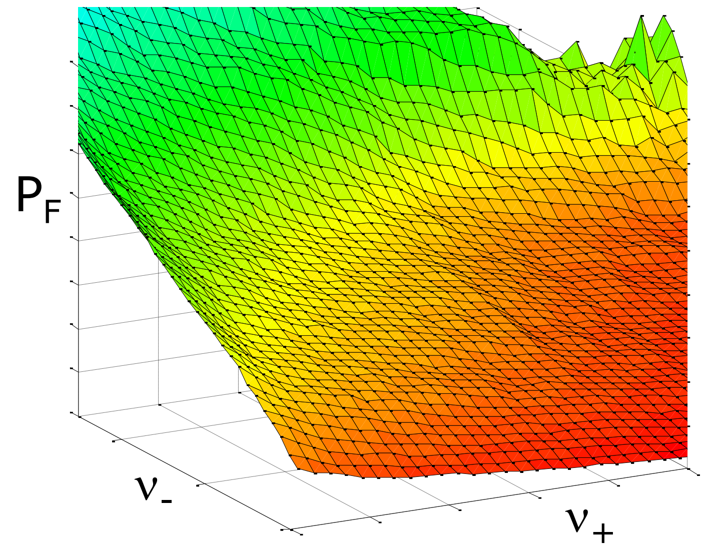
Error Estimation

Cross-Validation

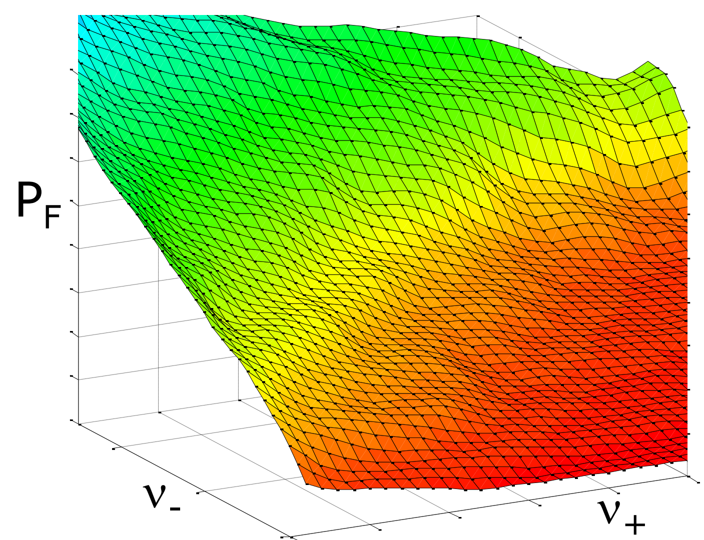


- CV has high variance
- Filtering reduces the variance and yields a better error estimate

True False Alarm Rate



Filtered Cross-Validation



Filtering Results

- Filtering provides strong performance gains
- Shape of the filter doesn't seem to matter
  - Gaussian window
  - Uniform (boxcar) filter
  - Median filter

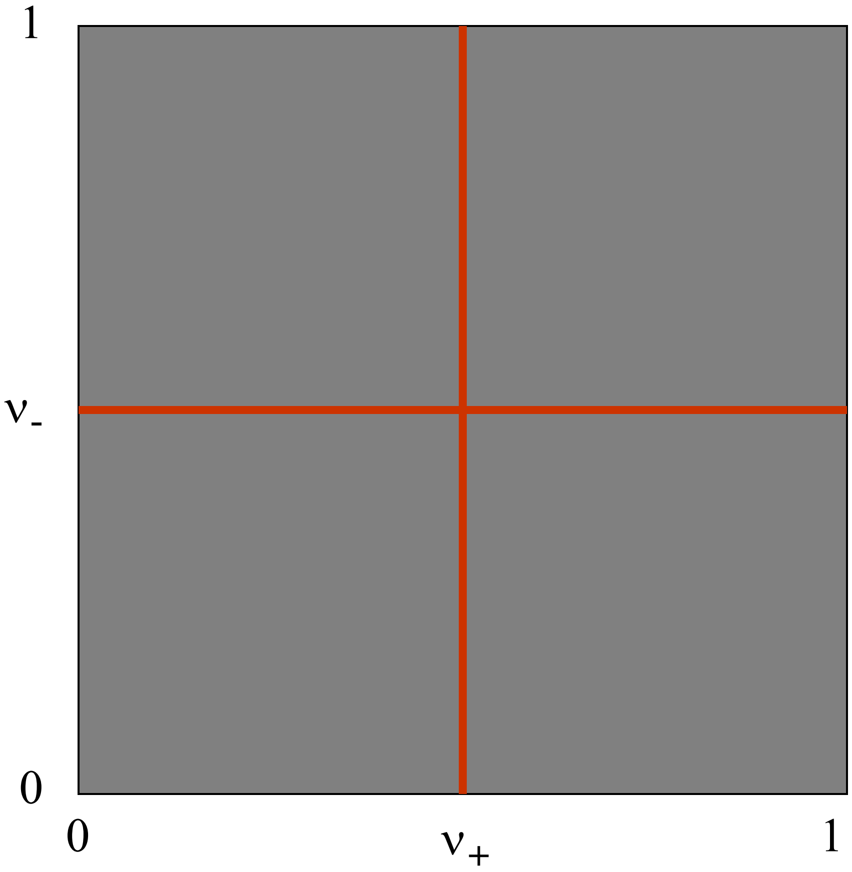
		P_F	P_M	E
thyroid	GS	.10	.06	.127
	FGS	.09	.04	.051
heart	GS	.12	.22	.375
	FGS	.11	.23	.326
cancer	GS	.16	.67	1.122
	FGS	.11	.69	.821
banana	GS	.11	.12	.255
	FGS	.10	.12	.160

$\alpha = 0.1$

GS: grid search
FGS: filtered grid search

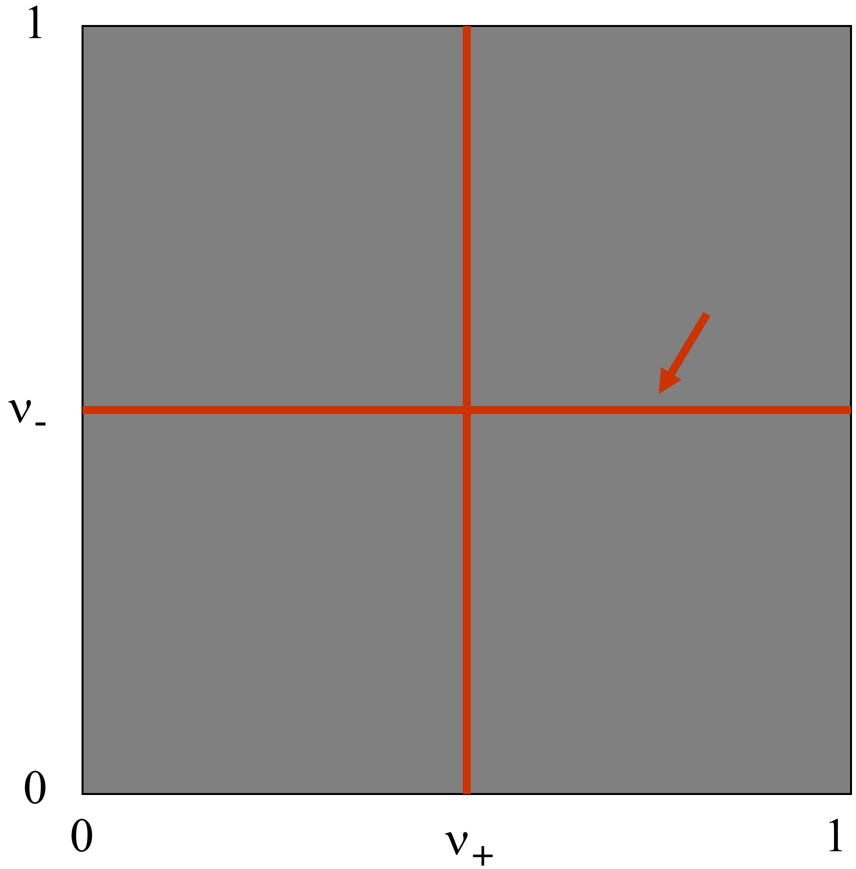
Coordinate Descent

- Technique for reducing training time
- Eliminates full grid search



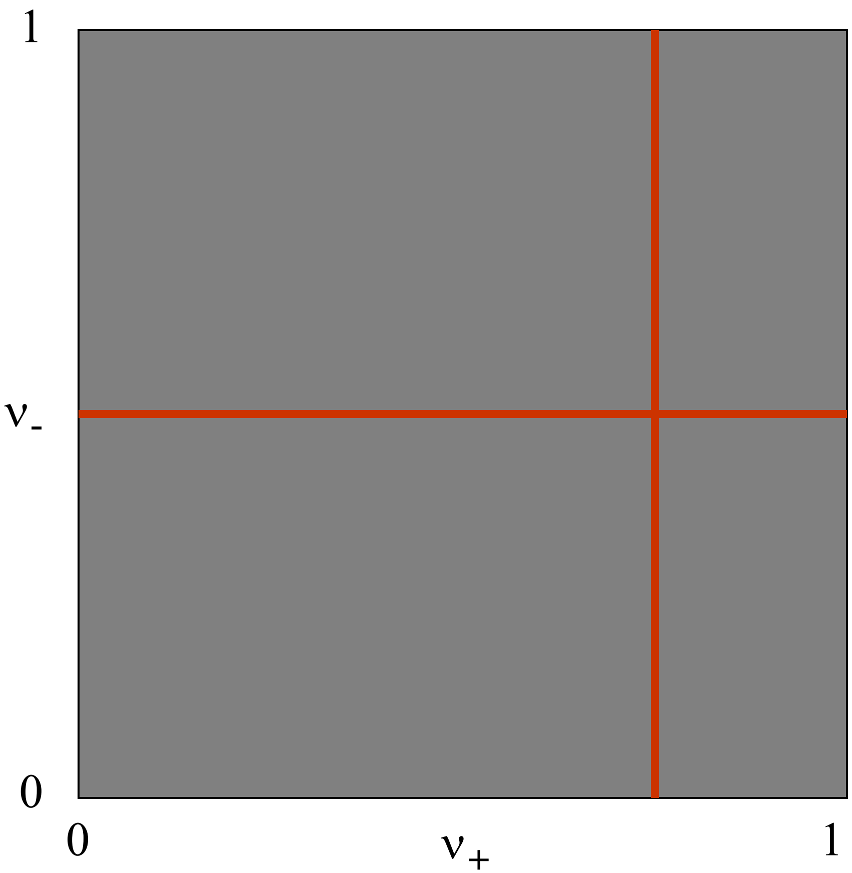
Coordinate Descent

- Technique for reducing training time
- Eliminates full grid search



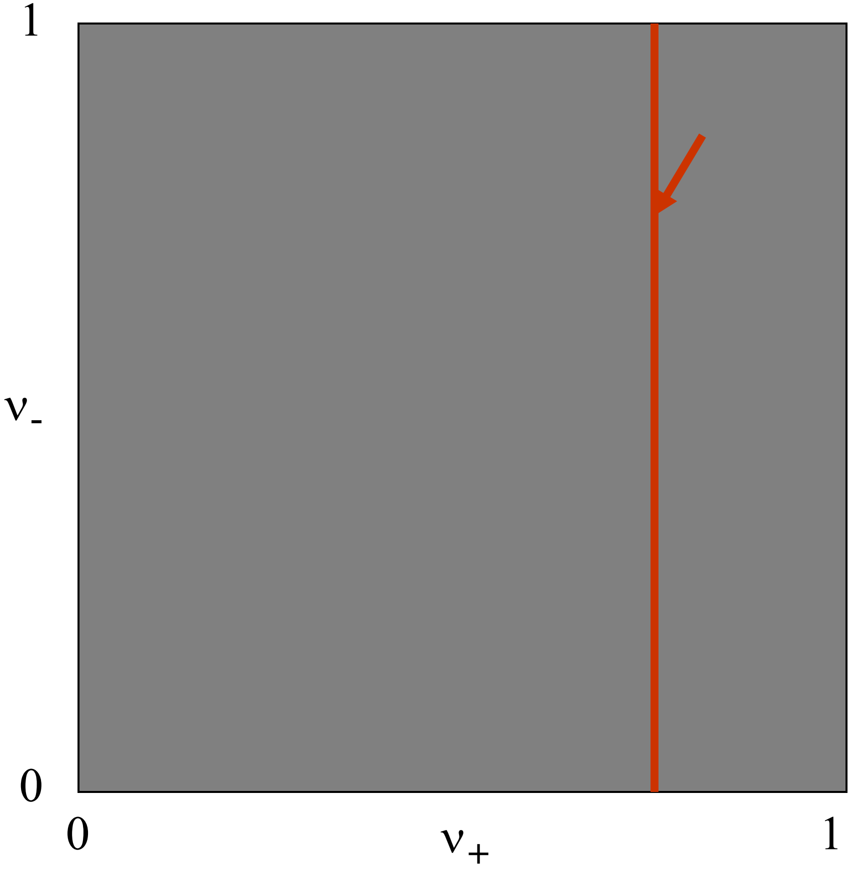
Coordinate Descent

- Technique for reducing training time
- Eliminates full grid search



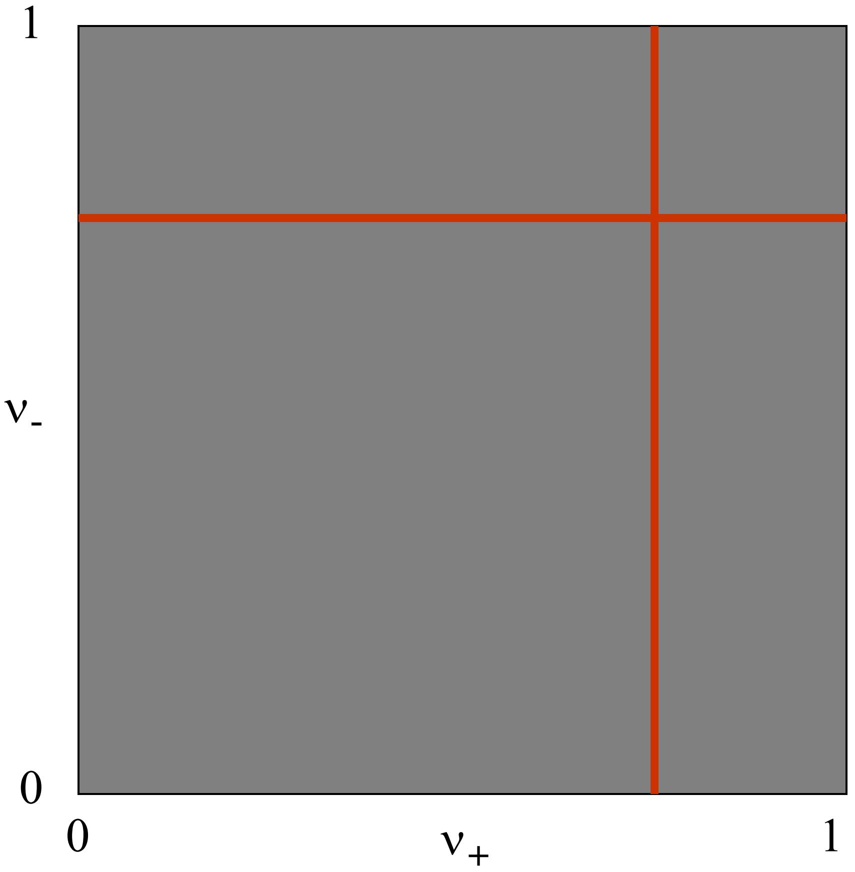
Coordinate Descent

- Technique for reducing training time
- Eliminates full grid search



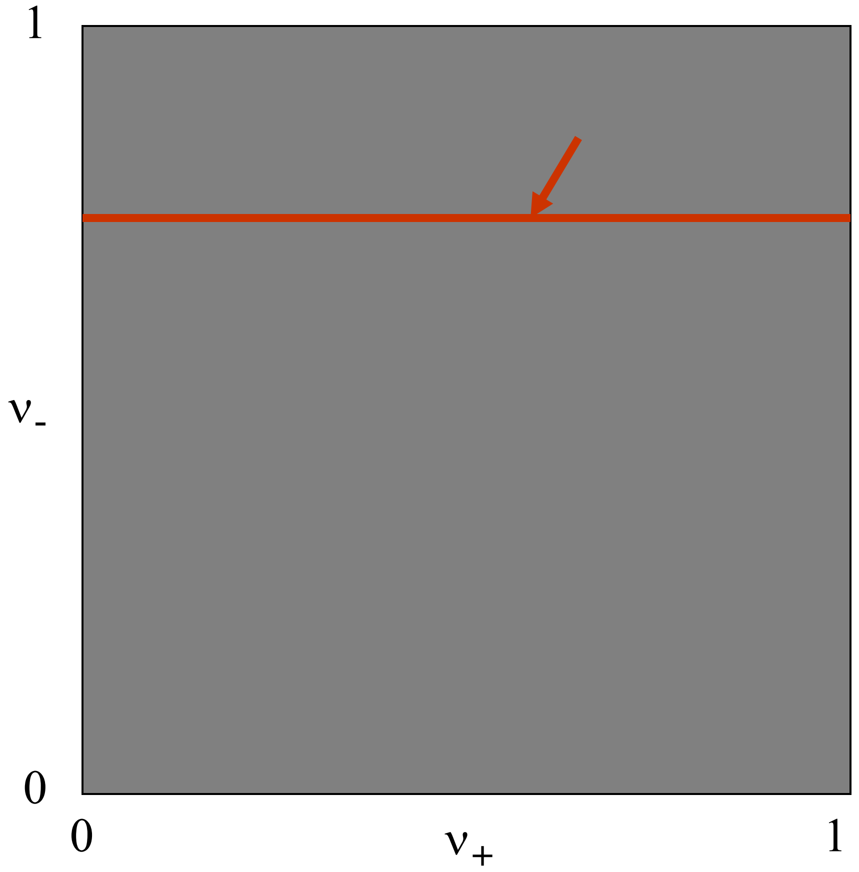
Coordinate Descent

- Technique for reducing training time
- Eliminates full grid search



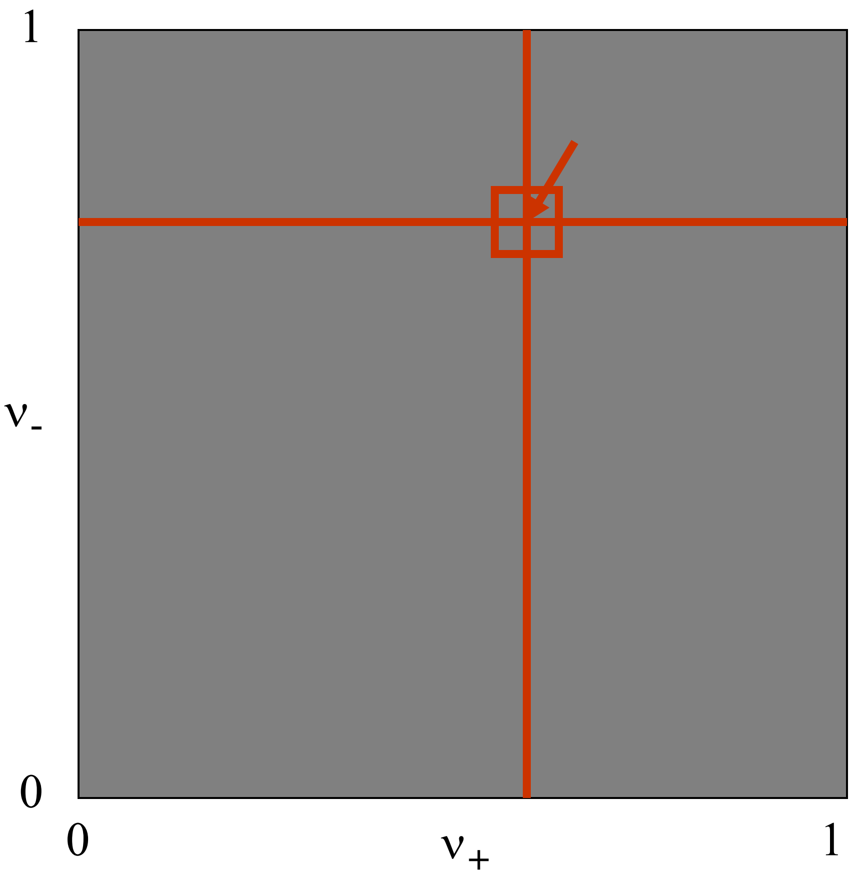
Coordinate Descent

- Technique for reducing training time
- Eliminates full grid search



Coordinate Descent

- Technique for reducing training time
- Eliminates full grid search



Coordinate Descent Results

- Training time is almost as fast as bias-shifting
- Performance comparable to full grid search
- Many more techniques for fast search possible

		P_F	P_M	E
thyroid	**CD**	**.08**	**.04**	**.066**
	FGS	**.09**	**.04**	**.051**
heart	**CD**	**.11**	**.23**	**.318**
	FGS	**.11**	**.23**	**.326**
cancer	**CD**	**.11**	**.68**	**.871**
	FGS	**.11**	**.69**	**.821**
banana	**CD**	**.10**	**.13**	**.179**
	FGS	**.10**	**.12**	**.160**

$\alpha = 0.1$

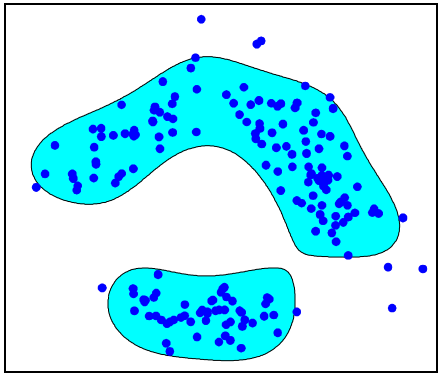
CD: coordinate descent
FGS: filtered grid search

Conclusion

- 2ν-SVM consistently outperforms bias-shifting at controlling false alarms

- Simple techniques improve performance
  - more accurate error estimation through filtering
  - faster training through coordinate descent

- Applications:
  - anomaly detection with minimum volume sets
  - minimax classification



- Code available at www.dsp.rice.edu/software

References

C. Scott and R. Nowak, "A Neyman-Pearson approach to statistical learning," *IEEE Transactions on Information Theory*, 2005.

B. Schölkopf, A. J. Smola, R. Williams, and P. Bartlett, "New support vector algorithms," *Neural Computation*, 2000.

H. G. Chew, R. E. Bogner, and C. C. Lim, "Dual-ν support vector machine with error rate and training size biasing," ICASSP 2001.

C. C. Chang and C. J. Lin, "Training ν support vector classifiers: Theory and algorithms," *Neural Computation*, 2001.

M. Davenport, "The 2ν-SVM: A cost-sensitive extension of the ν-SVM," *http://www.ece.rice.edu/~md*.

C. Scott, "Performance measures for Neyman-Pearson classification," *http://www.stat.rice.edu/~cscott*.