Predictive Learning from Data

LECTURE SET 9-2

SVM Practical Issues and Application Studies

Cherkassky, Vladimir, and Filip M. Mulier. *Learning from data: concepts, theory, and methods*. John Wiley & Sons, 2007.

Source: Dr. Vladimir Cherkassky (revised by Dr. Hsiang-Han Chen)

OUTLINE

- **Practical issues for SVM classifiers**
  - input scaling
  - unbalanced settings
  - multi-class problems
  - SVM software implementations
- Univariate histograms for SVM classifiers
- SVM model selection
- Application studies
- Summary

SVM Practical Issues

- **Understand assumptions** for classification setting & relate them to applic. requirements, i.e. *performance indices*
- **Data Scaling** scale all inputs to [0,1] range
- **Type of SVM problem**
  - classification (binary, multi-class, …)
  - regression
  - single-class learning
  - etc.
- **Implementations of SVM Algorithm**

Unbalanced Settings for Classification

- **Unbalanced Data:** relative size of +/- class encoded as prior probabilities for:

$$training \sim \pi_t^+ / \pi_t^-$$

$$test \sim \pi^+ / \pi^-$$

- **Misclassification Costs:** FP vs FN errors
- **(linear) SVM Classification Formulation:**

$$C^+ \sum_{i \in +class} \xi_i + C^- \sum_{i \in -class} \xi_i + \frac{1}{2}\|\mathbf{w}\|^2$$

where $$C^+ = Cost(false\,neg)\pi^+\pi_t^-$$
$$C^- = Cost(false\,pos)\pi^-\pi_t^+$$

Multi-Class SVM Classifiers

- **Multiple Classes:** **J** output classes

- **Problems:** usually unbalanced;
  misclassification costs (unknown)

- **Approaches for Multi-Class Problems:**
  - **J** *one-vs-all* binary classifiers
  - **J(J-1)/2** *pairwise* binary classifiers

SVM Implementations

- **General-purpose quadratic optimization**

  - for small data sets (~1,000 samples)

When the kernel matrix does not fit in memory, use:

- **Chunking methods**

  - apply QP to a manageable subset of data

  - keep only SV's

  - add more data, etc

- **Decomposition methods** (SVMLight, LIBSVM)

  - split the data (and parameters) in a number of sets, called *'working sets'*

  - perform optimization separately in each set

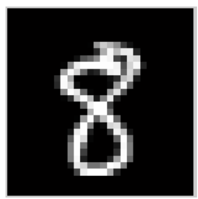
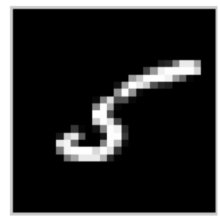
  - **Sequential Minimal Optimization** (SMO) uses working set of just two points (when analytic solution is possible)

OUTLINE

- Practical issues for SVM classifiers
- **Univariate histograms for SVM classifiers**
- SVM model selection
- Application studies
- Summary

Interpretation of SVM models

Humans can not provide interpretation of high-dimensional data, even when they can make good prediction

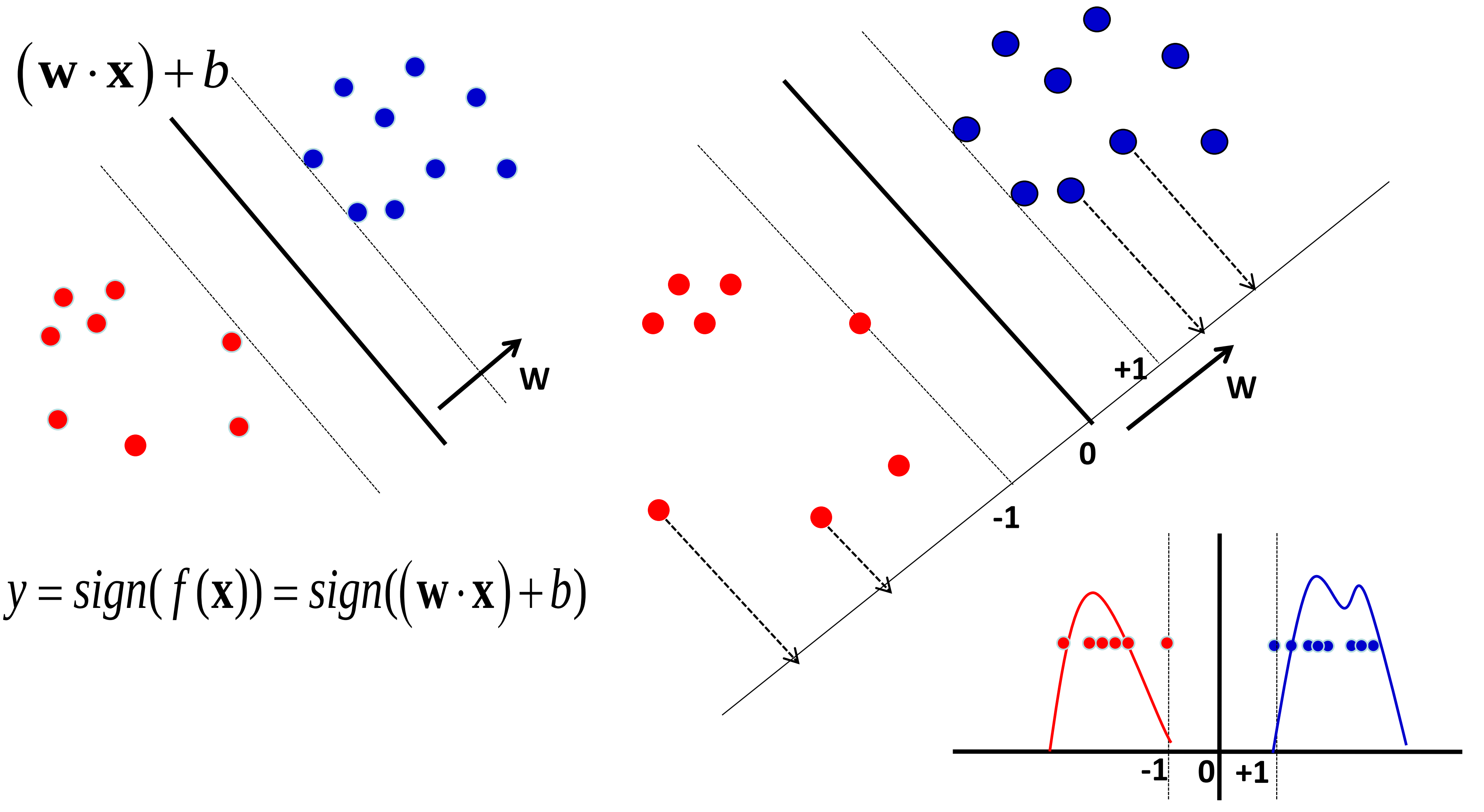
Example:  **vs** 

How to interpret high-dimensional models?

- Project data samples onto normal direction **w** of SVM decision boundary D(**x**) = (**w x**) + b = 0
- Interpret univariate histograms of projections

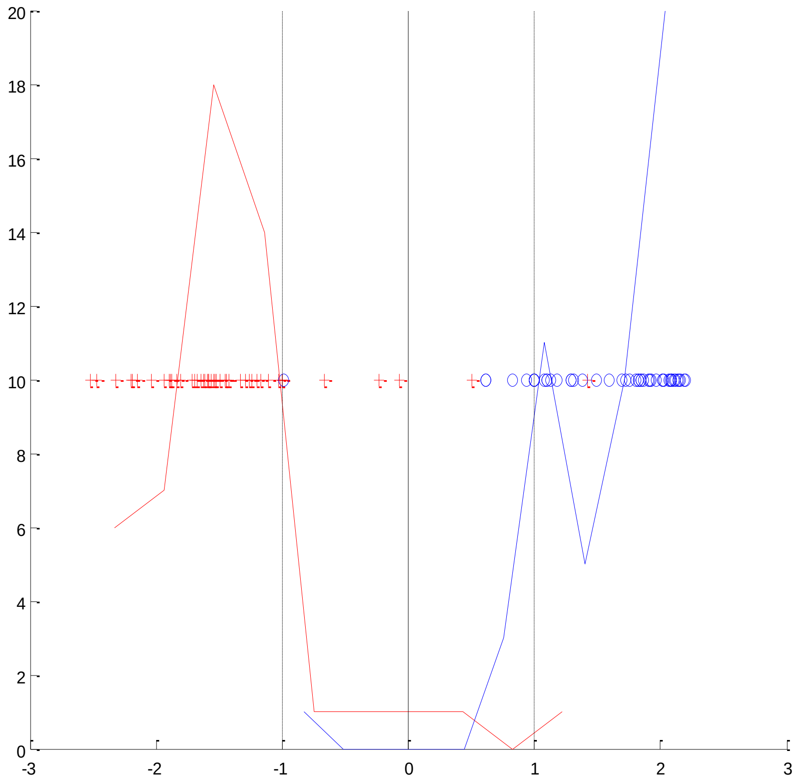
Univariate histogram of projections

Project training data onto normal vector **w** of the trained SVM

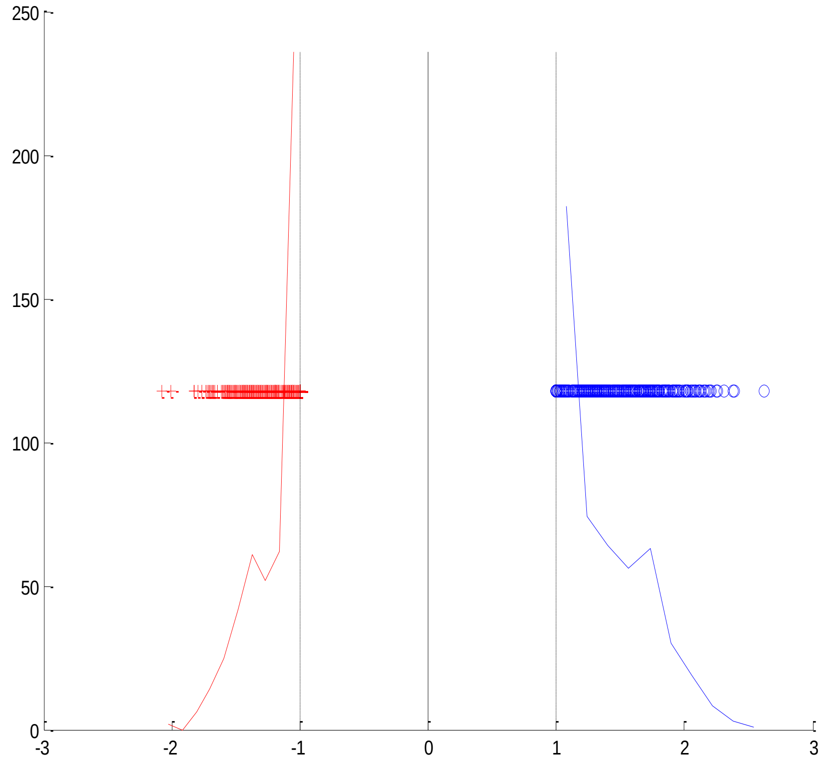


$$y = sign(f(\mathbf{x})) = sign((\mathbf{w} \cdot \mathbf{x}) + b)$$

Example histograms
(for *balanced* high-dimensional training data)



Non separable data



Separable data

OUTLINE

- Practical issues for SVM classifiers
- Univariate histograms for SVM classifiers
- **SVM model selection**
  - **General strategies**
  - **Model selection for classification**
  - **Model selection for regression**
- Application studies
- Summary

Strategies for Model Selection

- Setting/ tuning of SVM hyper-parameters
  - usually performed by experts
  - more recently, by non-expert users
- Issues for SVM model selection
  (1) parameters controlling the 'margin' size
  (2) kernel type and kernel complexity
- Strategies for model selection
  - exhaustive search in the parameter space (via resampling)
  - efficient search using VC analytic bounds
  - rule-of-thumb analytic strategies (for a particular type of learning problem)

Strategies continued

- **Parameters controlling margin size**

  - for ***classification***, regularization parameter C

  - for ***regression***, the value of epsilon

  - for ***single-class learning***, the radius

- **Complexity control~ fraction of SV's** $\nu \in [0,1]$

  - for classification, replace C with ν

  - for regression, specify the fraction of points ν allowed to lie outside ε -insensitive zone

- **Very sparse data (d/n>>1) → linear SVM**

Model Selection for Classification

- **Parameters C and kernel,** via resampling:
  Training data + Validation data
  Consider RBF kernel $K(\mathbf{x},\mathbf{x}') = \exp\left(-\gamma \|\mathbf{x}-\mathbf{x}'\|^2\right)$

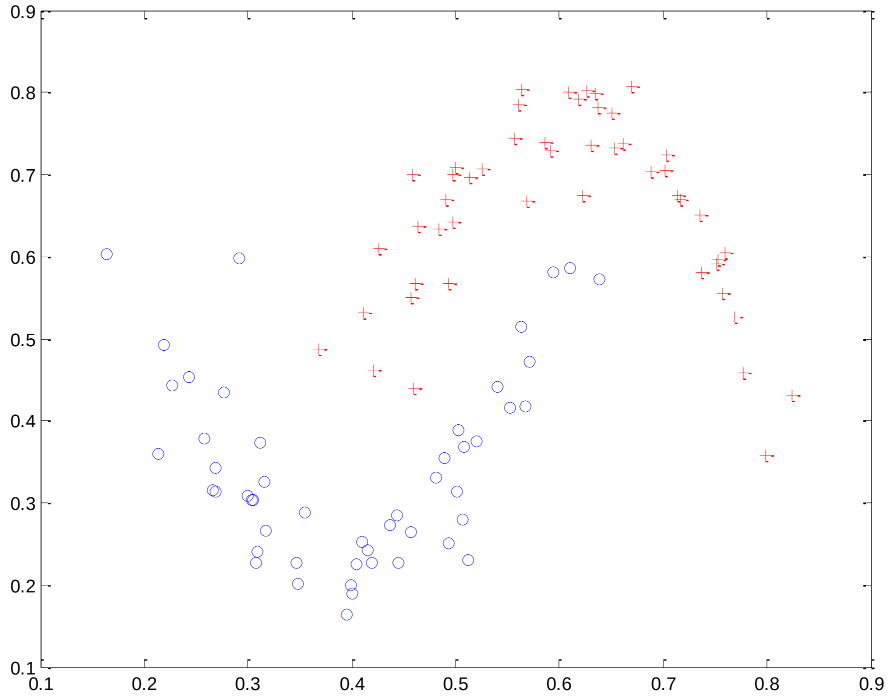
MODEL SELECTION Procedure

[**1**] Estimate SVM model for each (C, γ) values using the **training data**.

[**2**] Select the tuning parameters (C*, γ*) that provide the smallest error on the **validation data** samples.

- **In practice, use K-fold cross-validation**

Hyperbolas Data Set



$x_1 = ((t-0.4)*3)^2+0.225$
$x_2 = 1-((t-0.6)*3)^2-0.225.$

$t \in [0.2, 0.6]$ for class 1.(Uniform)
$t \in [0.4, 0.8]$ for class 2.(Uniform)

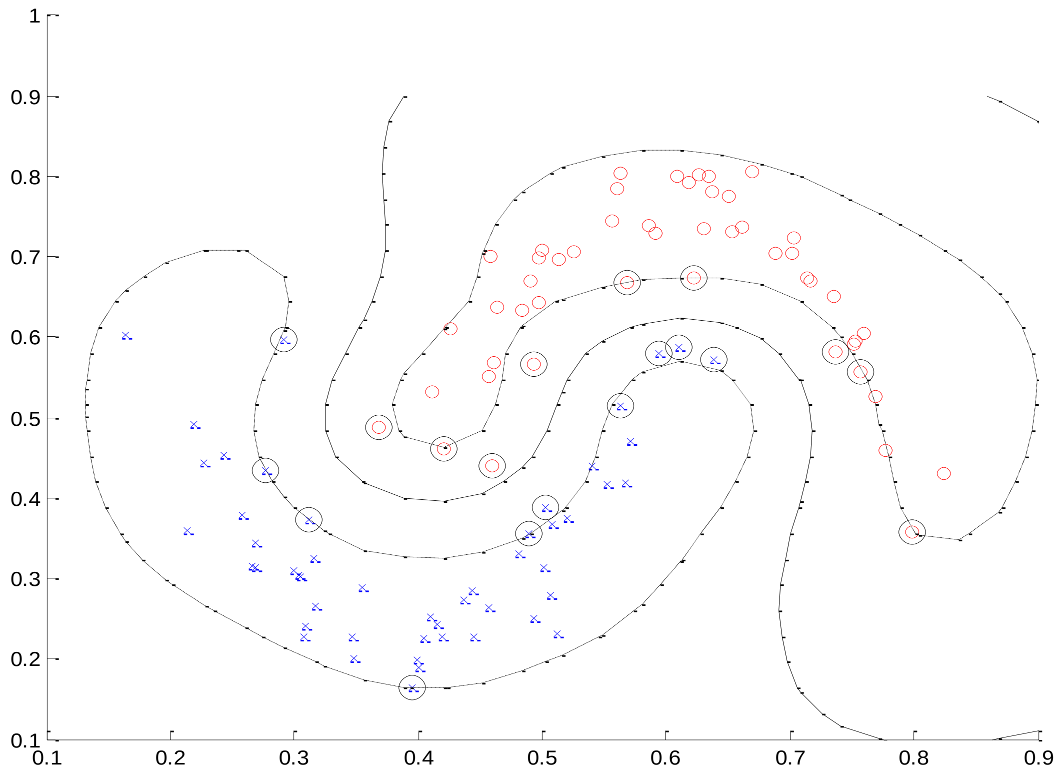
Gaussian noise with st. dev. = 0.03 added to both x_1 and x_2

- 100 Training samples (50 per class)/ 100 Validation.
- 2,000 Test samples (1000 per class).

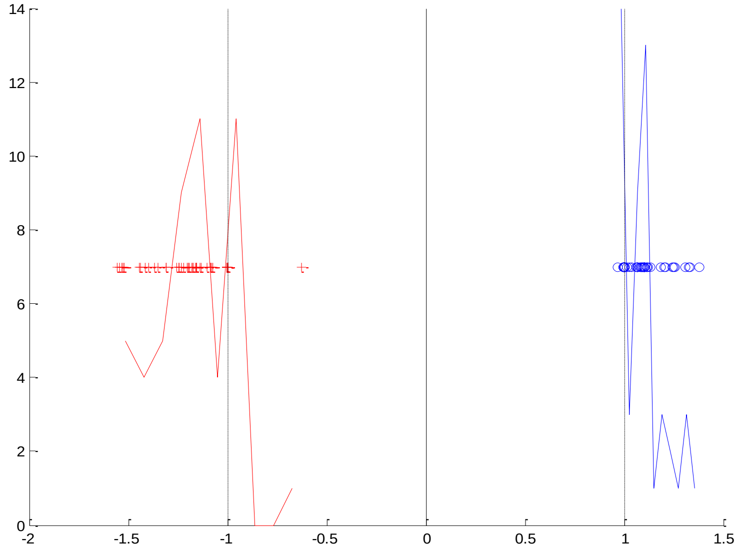
Hyperbolas Example (cont'd)

- Range of SVM parameter values: $C \sim [2^{-2}, 2^{-1}, \ldots, 2^{4}]$
- Optimal values C ~ 2 and γ ~ 64 $\gamma \sim [48, 56, 64, \ldots, 88, 96]$

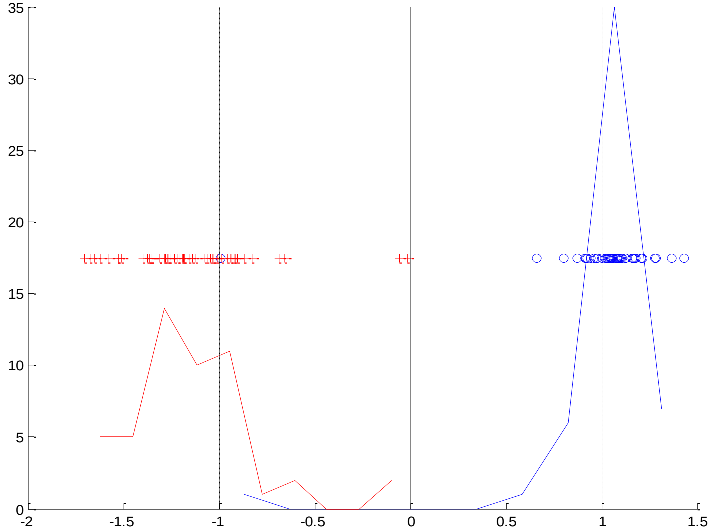
→ SVM model with training data:



TYPICAL HISTOGRAMs OF PROJECTIONS

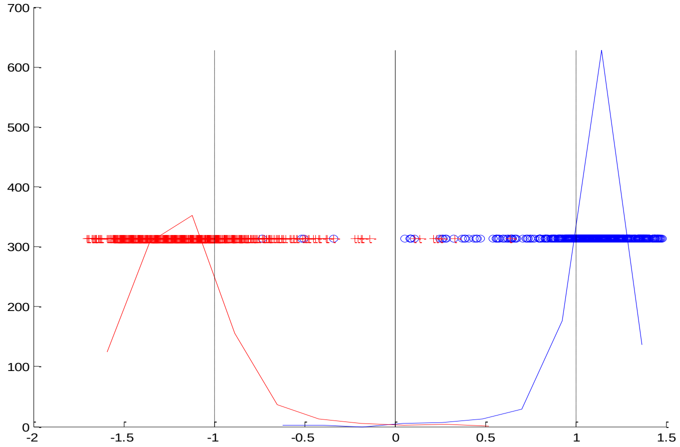


(a) Projections of training data (100 samples). *Training error=0*

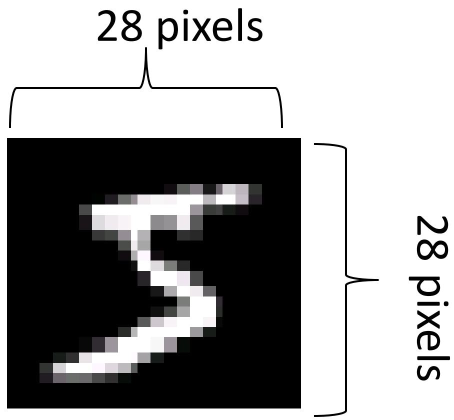


(b) Projections of validation data. *Validation error=0 %*

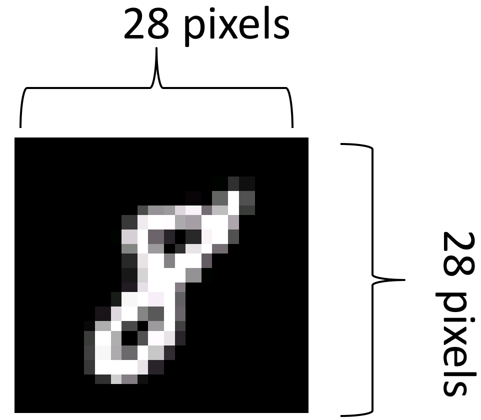
(c) Projections of test data (2,000 samples)
Test error =0.55%



MNIST Data (handwritten digits)



Digit “5”



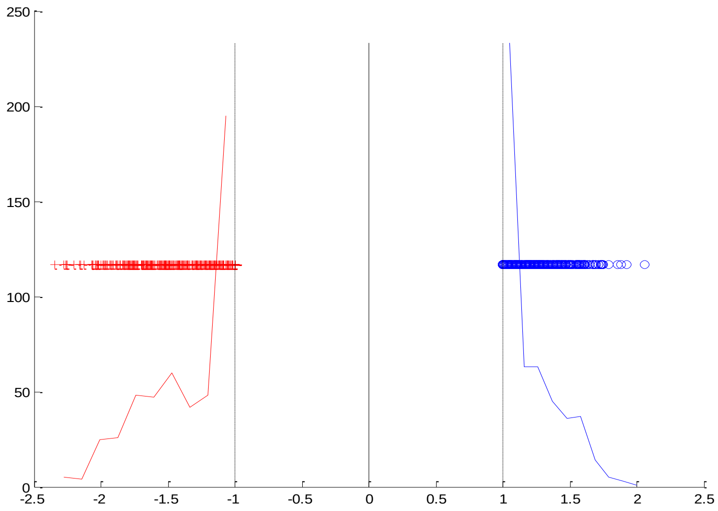
Digit “8”

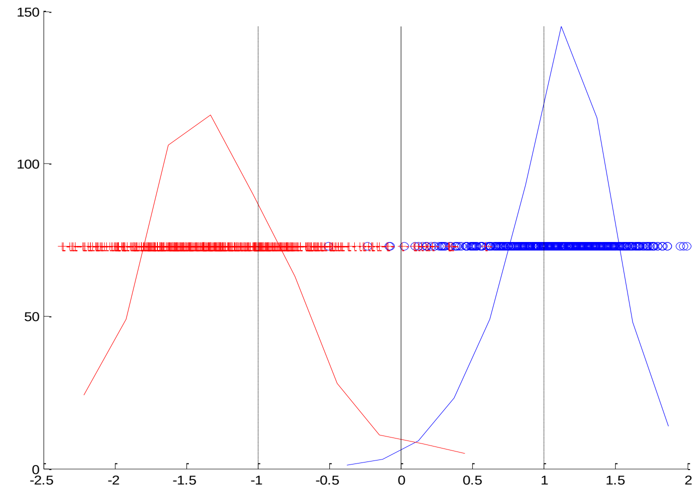
<u>Binary classification task:</u> digit “5” vs. digit “8”

- No. of Training samples = 1000. (500 per class).
- No. of Validation samples = 1000.(used for model selection).
- No. of Test samples = 1866.
- Dimensionality of each sample = 784 (28 x 28).
- Range of SVM parameters: $C \sim [10^{-2}, 10^{-1}, \ldots, 10^{3}]$

$$\gamma \sim [2^{-8}, 2^{-6}, \ldots, 2^{-2}]$$

TYPICAL HISTOGRAMs OF PROJECTIONS





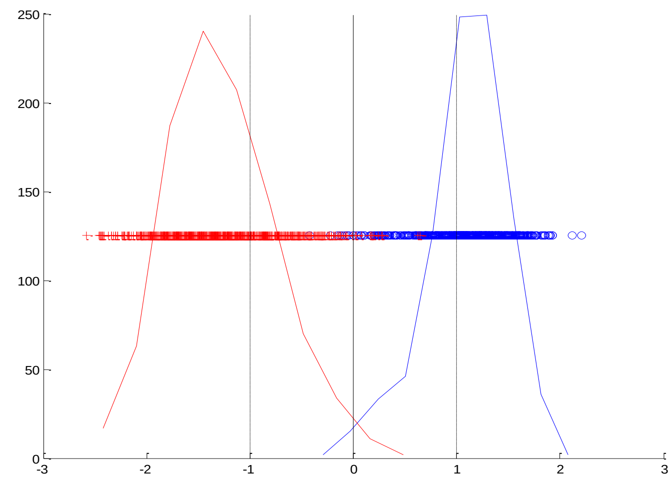
(a) Projections of training data (1000 samples). *Training error=0*

(b) Projections of validation data. *Validation error=1.7%*

- **Selected SVM parameter values**

$$C \sim 1 \ or \ 10$$

$$\gamma \sim 2^{-6}$$



(c) Projections of test data (1866 samples). *Test error =1.23%*

Model Selection for HDLSS Data

- **High-Dim. Low Sample Size** (HDLSS)

  - many applications: genomics, fMRI…

  - sample size(~10's)<<dimensionality (~1000)
- **Very Ill-Posed Problems**
- **Issues for SVM classifiers**

  (1) How to apply SVM classifiers to HDLSS?

  → use **linear SVM**

  (2) How to perform model selection?

MNIST data under HDLSS scenario

<u>EXPERIMENTAL SETUP :-</u> Binary classification digit "5" vs. digit "8"

- No. of Training samples = 20 (10 per class).
- No. of Validation samples = 20 (for model selection).
- No. of Test samples = 1866.
- Dimensionality = 784 (28 x 28).
- Model estimation method **Linear SVM** (single tuning parameter C)

TWO MODEL SELECTION STRATEGIES for linear SVM:

1. Use **independent validation set** for tuning C
2. Set C to **fixed small value** providing maximum margin

EXPERIMENTAL PROCEDURE: repeat comparison 10 times using 10 independent training/validation data sets

Model Selection for SVM Regression

- Selection of parameter *C*

Recall the SVM solution $f(\mathbf{x}) = \sum_{i=1}^{n} (\alpha_i^* - \beta_i^*) H(\mathbf{x}_i, \mathbf{x}) + b$

where $0 \leq \alpha_i^* \leq C$ and $0 \leq \beta_i^* \leq C, i = 1, \ldots, n$

→ with bounded kernels (RBF) $C = y_{\max} - y_{\min}$

- Selection of ε

in general, $\varepsilon \sim \sigma$ (noise level)

But this does not reflect dependency on *sample size*

For linear regression: $\sigma_{y/x}^2 \propto \frac{\sigma^2}{n}$ suggesting $\varepsilon \propto \frac{\sigma}{\sqrt{n}}$

- Final prescription $\varepsilon = 3\sigma \sqrt{\frac{\ln n}{n}}$

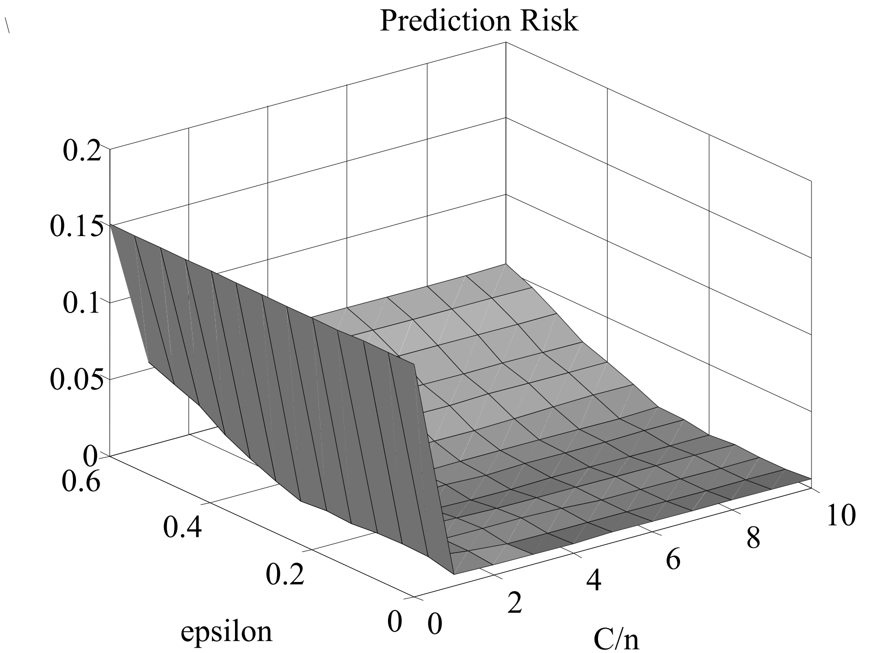
Effect of SVM parameters on test error

- Training data

  univariate Sinc(x) function $t(x) = \dfrac{\sin(x)}{x} \qquad x \in [-10,10]$

  with additive Gaussian noise (sigma=0.2)

  (a) small sample size 50 (b) large sample size 200

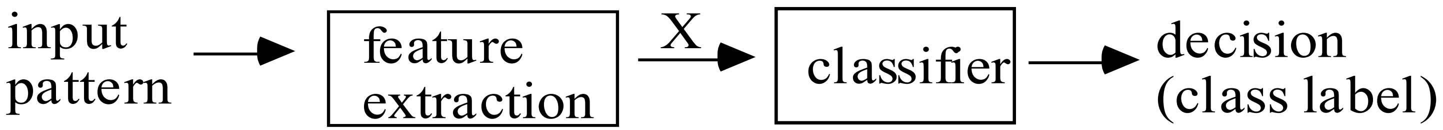


OUTLINE

- Practical issues for SVM classifiers
- Univariate histograms for SVM classifiers
- SVM model selection
- **Application studies**
  - Modeling issues preceding learning
  - Prediction of transplant-related mortality
  - Prediction of epileptic seizures from EEG
  - Online fraud detection
- Summary

Data Modeling Issues *Outside SVM*

- Generic system (for classification)



- Important modeling issues:
  - *formalization:* application → learning problem
  - *pre-processing + data encoding* (Section 2.1)
  - *preliminary data analysis* (~univariate boxplots…)
  - *feature selection (extraction)*

Note 1: these steps precede learning/ model estimation

Note 2: these steps are often neglected, even though they account for 80-90% of success

Note 3: some methods incorporate feature selection into learning algorithm (~ deep learning NNs)

Prediction of TRM

- ***Graft-versus-host disease* (GVHD)** is a common side effect of an allogeneic bone marrow or cord blood transplant.
- **High Transplant-Related Mortality (TRM):** affects ~ 25- 40% of transplant recipients
- **Hypothesis:** specific genetic variants of donor/recipient genes have strong association with TRM
- **Two data sets:** UMN and Mayo Clinic (from the ‘same distribution’)
- **Problem Formalization**: prediction of TRM is modeled via binary classification (SVM)

Available Data for Modeling (UMN)

- 301 samples (donor/recipient pairs)

  - all donor sources: sibling, unrelated, cord

  - all stem cell sources: peripheral blood, bone marrow, cord blood

  - variety of conditioning regimens

  - demographic variables (i.e., Age, Race)

  - 136 SNPs for each patient

- Unbalanced class distribution

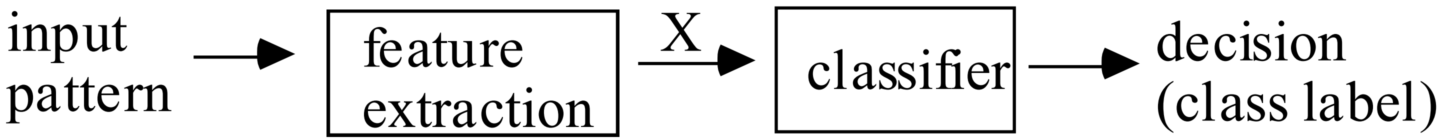
  genetic + **clinical** + **demographic** inputs

Goal predicting TRM in the **first year post-transplant**

~ **binary classification**: **alive(+)** vs **dead(-)**

Data Modeling Issues

- Generic classification system



- Specific issues:
  - unbalanced data set
  - unequal miscassification costs
  - **genetic** + **clinical** + **demographic** inputs

- Goals of modeling
  - prediction of TRM in the first year post transplant can be formalized as *binary classification* : **alive(+)** vs **dead(-)**
  - identification of reliable biomarkers and high risk groups for TRM and GVHD.

Data Modeling Approach (cont'd)

- **Feature selection via**

  (1) classical statistical methods

  (2) machine learning methods (information gain ranking, mutual info maximization)

- **SVM classification** (using selected features)

  Resampling is used to estimate test error

  Prior probabilities: 75% 'alive' and 25% 'dead'

  Misclassification costs: $C+/C- \sim 1/3$

  cost of false_positive vs false_negative

  Performance index (for comparing classifiers)

$$weighted\_test\_error = C^{+}P_{fp} + C^{-}P_{fn}$$

Modeling Results: Prediction of TRM

Feature Selection 1: machine learning method applied to all features (genetic and clinical) yields agetx, rs3729558, rs3087367, rs3219476, rs7099684, rs13306703, rs2279402

SVM Model 1 (with these 7 featurs)~ **test error 29%**

Feature Selection 4: Statistical Feature Selection applied to all features yields agetx, donor, cond1, race, rs167715, rs3135974, rs3219463

SVM Model (with these 7 features)~ **test error 38%**

For comparison: classification rule based on the majority class ~ **test error 48%**

Modeling Results (cont'd)

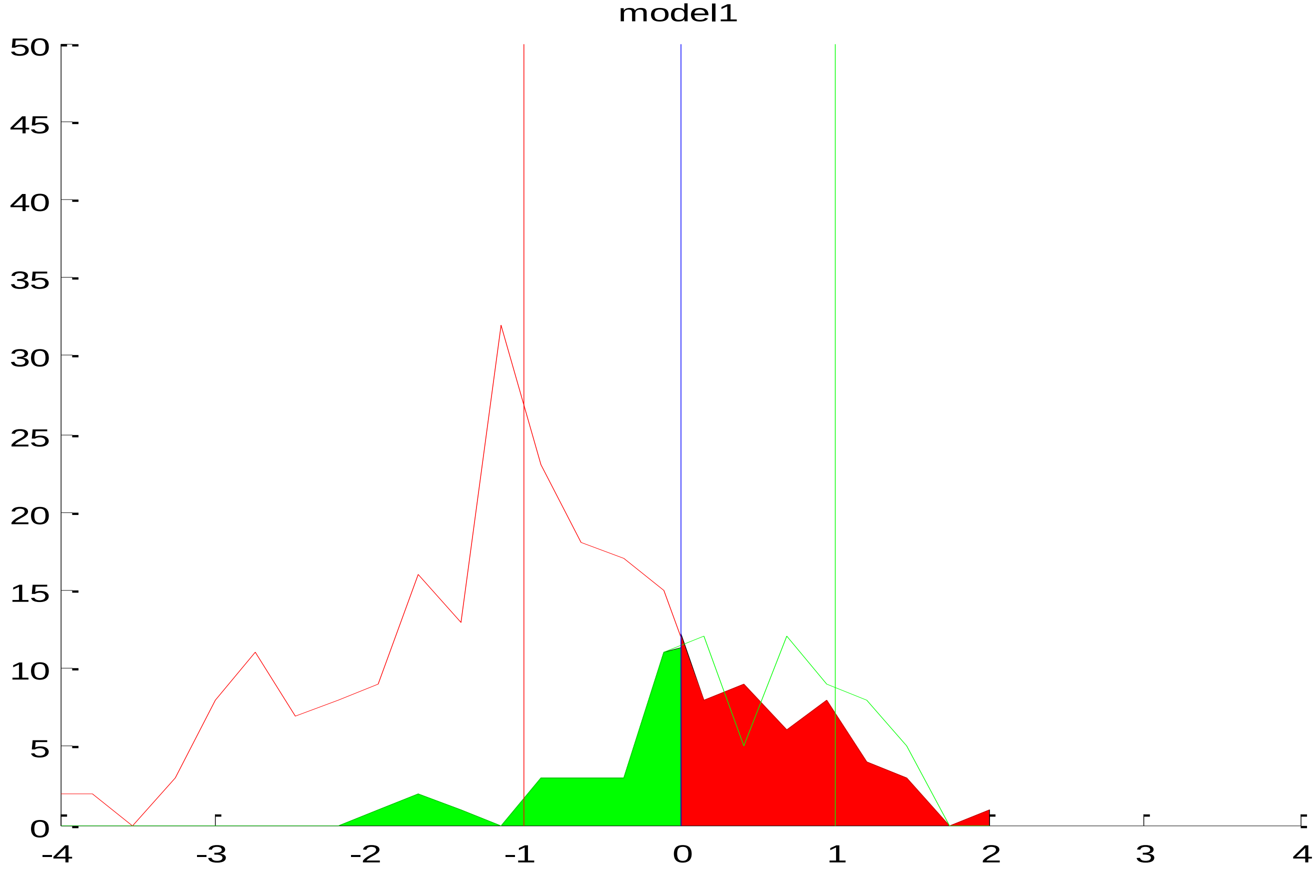
Feature Selection 3: machine learning method applied to genetic features only and then supplemented by clinical inputs provided by domain expert

rs3729558, rs3219476, rs13306703, rs2279402, rs3135974, rs3138360, Rfc5_13053, rs3213391, rs2066782, agetx, donor, cond1 and race

SVM Model 3(using these 13 inputs) ~ **test error 29%**

Note: different SVM models 1 and 3 provide *similar* prediction error. Which one to interpret?

Histogram for SVM Model 1



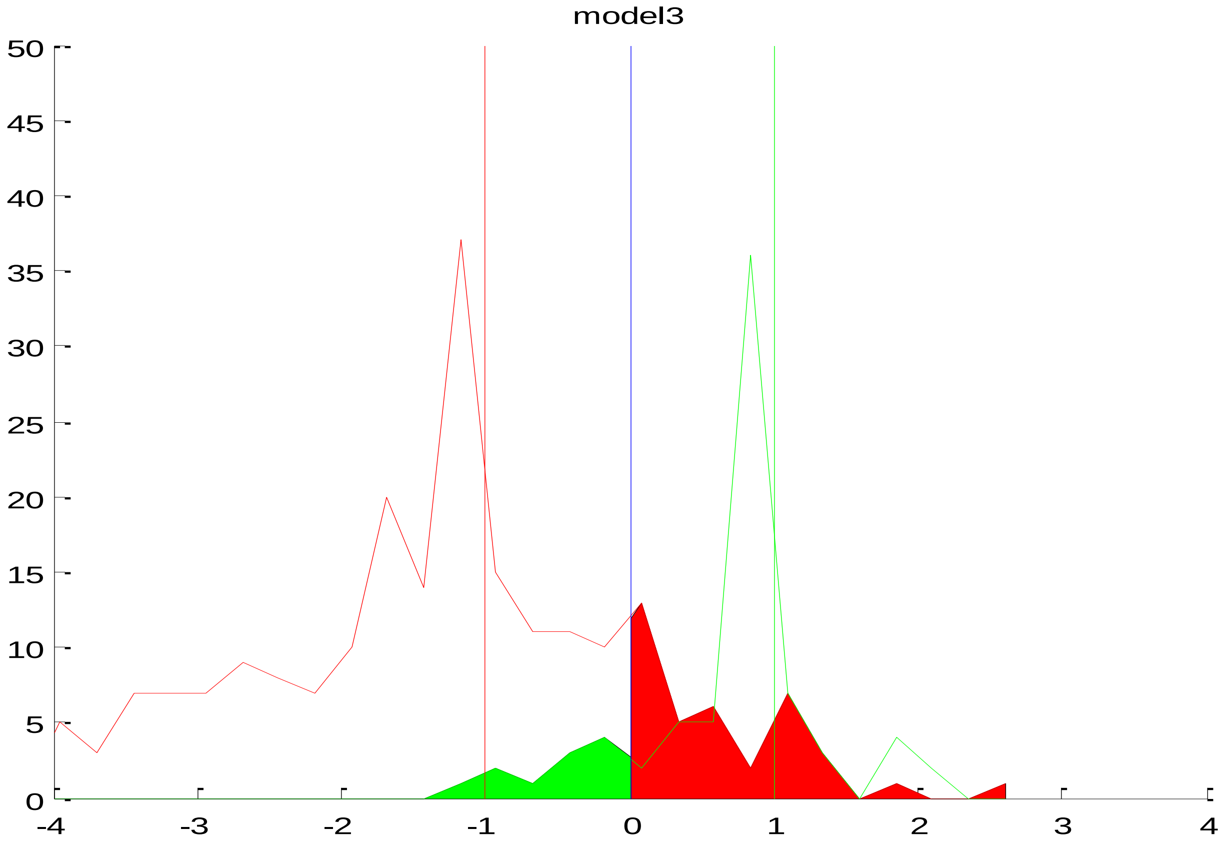
TP = 62 **FP = 56**

FN = 13 TN = 170

→ P_error_rate = FP/(TP+FP)=0.47

N_error_rate = FN/(TN+FN)=0.07

Histogram for SVM Model 3



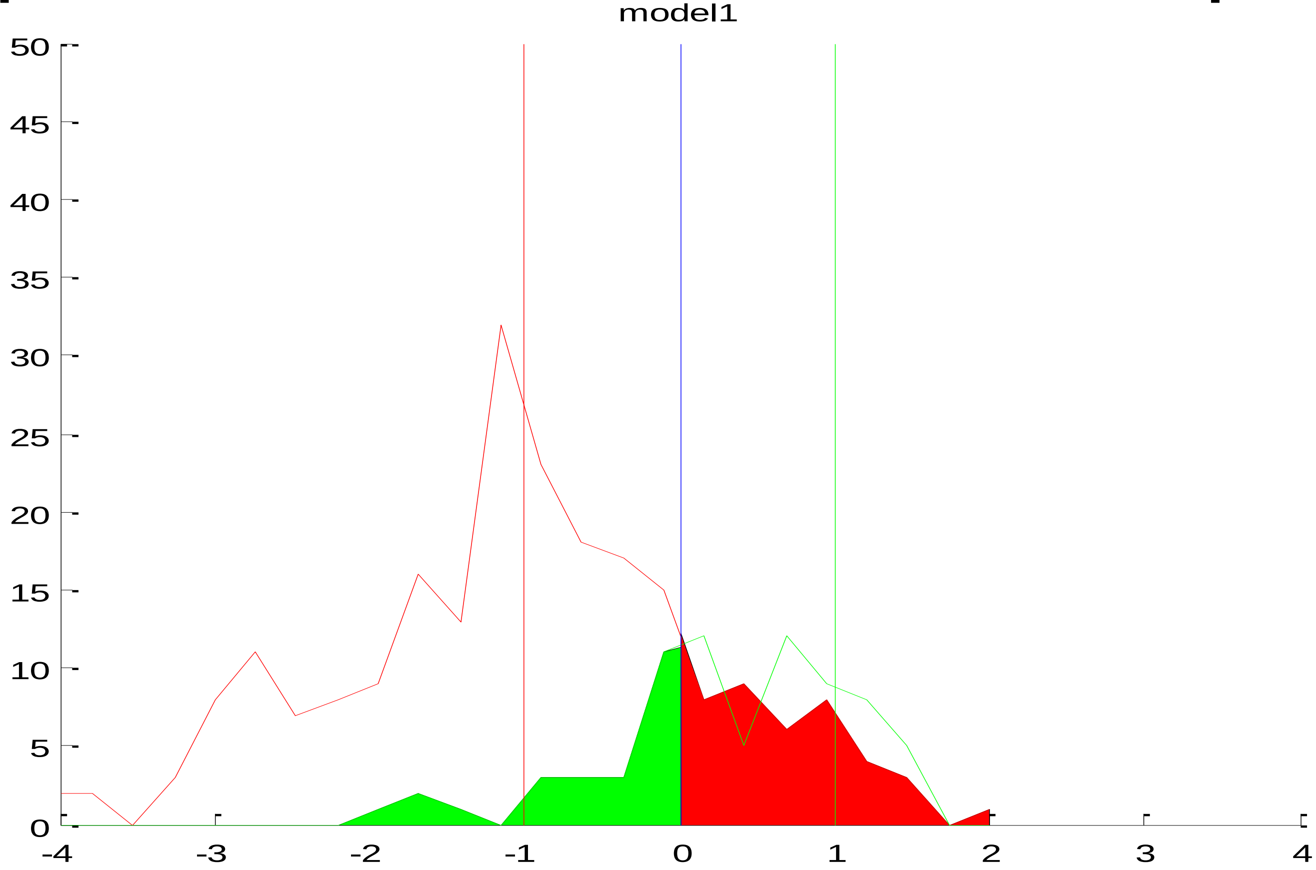
TP = 68 **FP = 45**

FN = 7 TN = 181

P_error_rate = FP/(TP+FP)=0.4

N_error_rate = FN/(TN+FN)=0.037

Interpretation of SVM Model 1: input Age



Stats for Age: Ave_AgeTP = 41 Ave_Age **FP = 44**

Ave_Age **FN = 37** Ave_Age TN = 27

Generalization for Mayo Clinic data

- UMN data set for training/ Mayo data for testing

  → very poor results ~ 46% test error

- Explanation is simple:

  - UMN data contained *more younger patients*
  - *Recipient_Age* input had most predictive value

Note: this violates the original premise that two data sets are (statistically) similar)

- Modeling results do not support the original hypothesis that genetic inputs have predictive value.

Prediction of Epileptic Seizures

(Netoff, Park and Parhi 2009)

- **Objective:** Patient-specific prediction of seizures (5 min ahead) from EEG signal (6 electrodes)
- **The main issue** is problem formalization:

  how to formalize 'good seizure prediction'?

  ~ preictal period

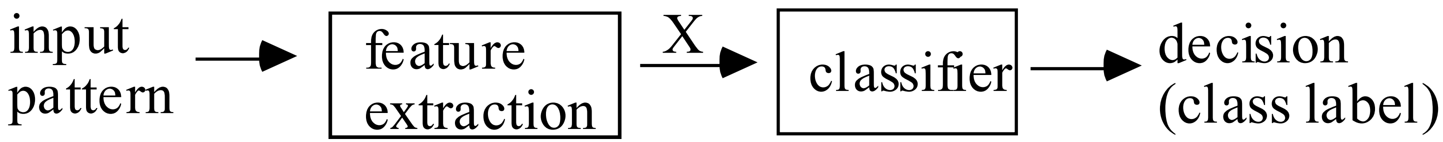
  ~ how far ahead to predict?
- **Proposed formalization:**

  - standard binary classifier for predicting 20 sec windows (interictal vs. preictal)

  - prediction is made 5 min ahead

Seizure Prediction via SVM Classifier

- **Objective:** Patient-specific prediction of seizures (5 min ahead) from EEG signal (6 electrodes)
- **Issues:** performance metrics, unbalanced data, feature selection, sound methodology for SVM

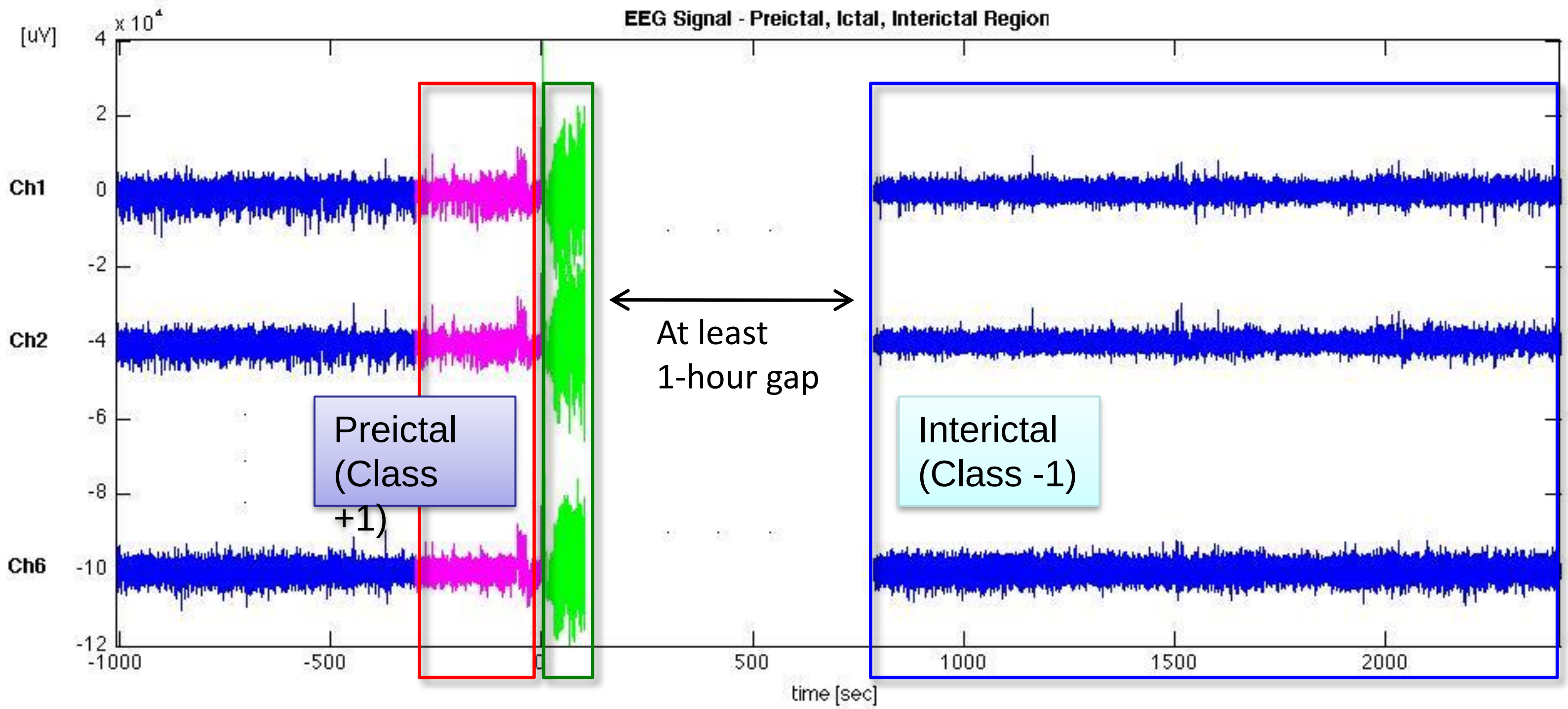


- **System implementation details:**

  - ***features*** ~ power measured in 9 spectral bands for each electrode. Total 9x6 = 54 features
  - ***classifier*** ~ SVM version with *unequal costs*
  - ***Freiburg data set***

Labeling EEG Data for SVM Classification

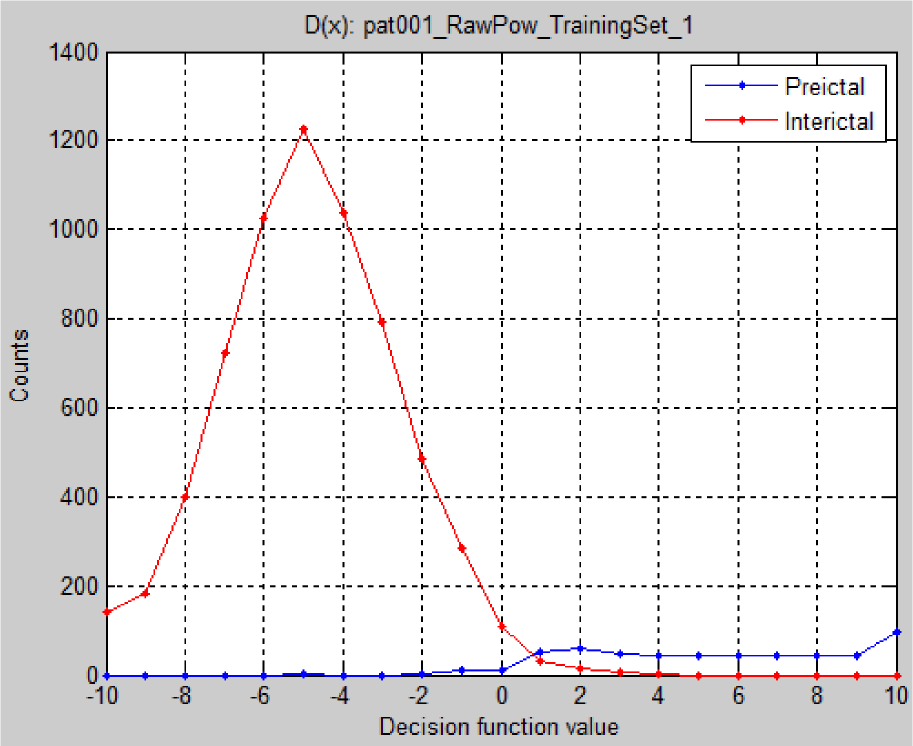
- Parts of EEG data identified by medical experts: **ictal**, **preictal (+)**, **interictal(-)**
- Preictal and interictal data used for classification
- Each data sample ~ 20 sec moving window

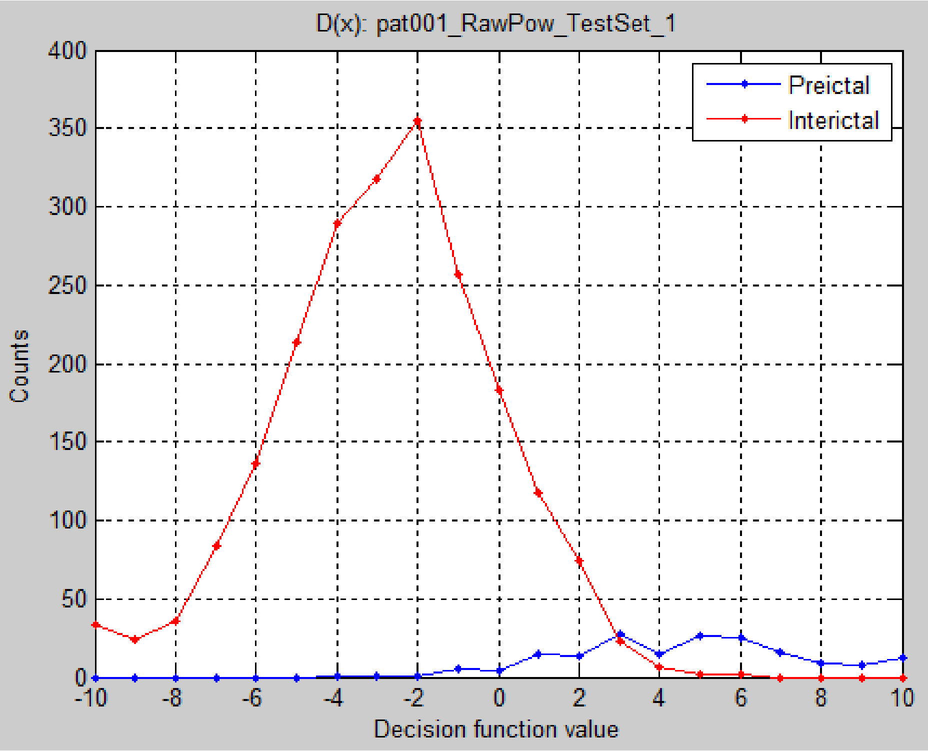


- **Unbalanced data** (patient 1):
  Total sample size: 9332
  7.7% positive (preictal), 92.3% negative (interictal)
  54 input features
- **Characterization of SVM method**
  linear SVM
  misclassification costs **Cost FN / Cost FP = 6 : 1**
- **Experimental procedure**
  Double resampling for:
  - model selection
  - estimating test error (out-of-sample)

SVM Modeling Results via projections

Patient 1: Training data and Test data





TP=552 FP=99

FN=15 TN=6363

→**NPV=TN/(TN+FN)=0.997**

PPV=TP/(TP+FP)=0.848

TP=170 FP=288

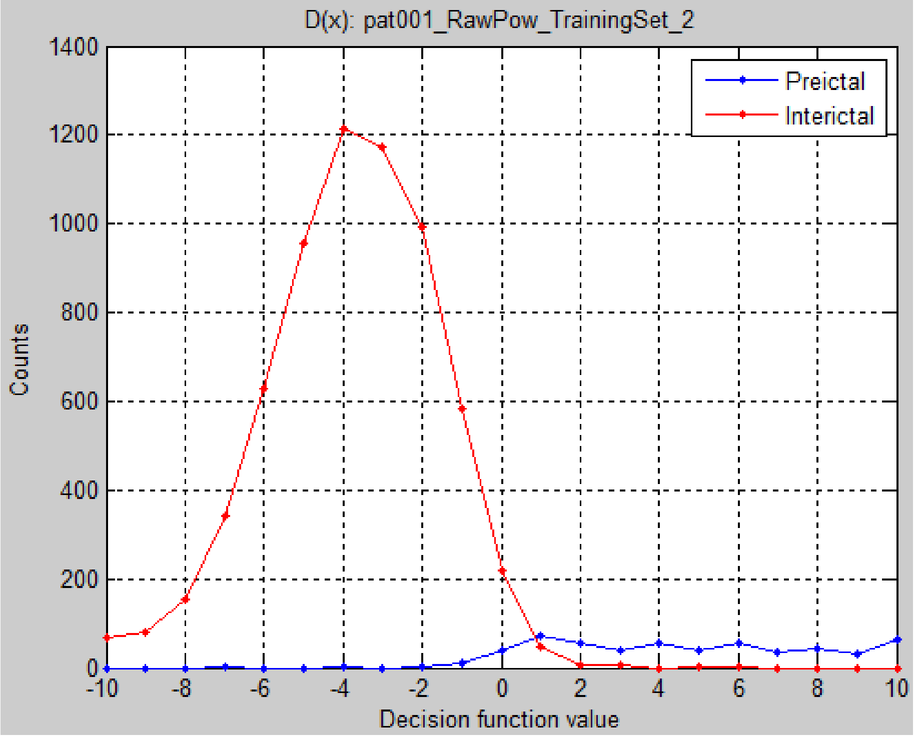
FN=9 TN=1866

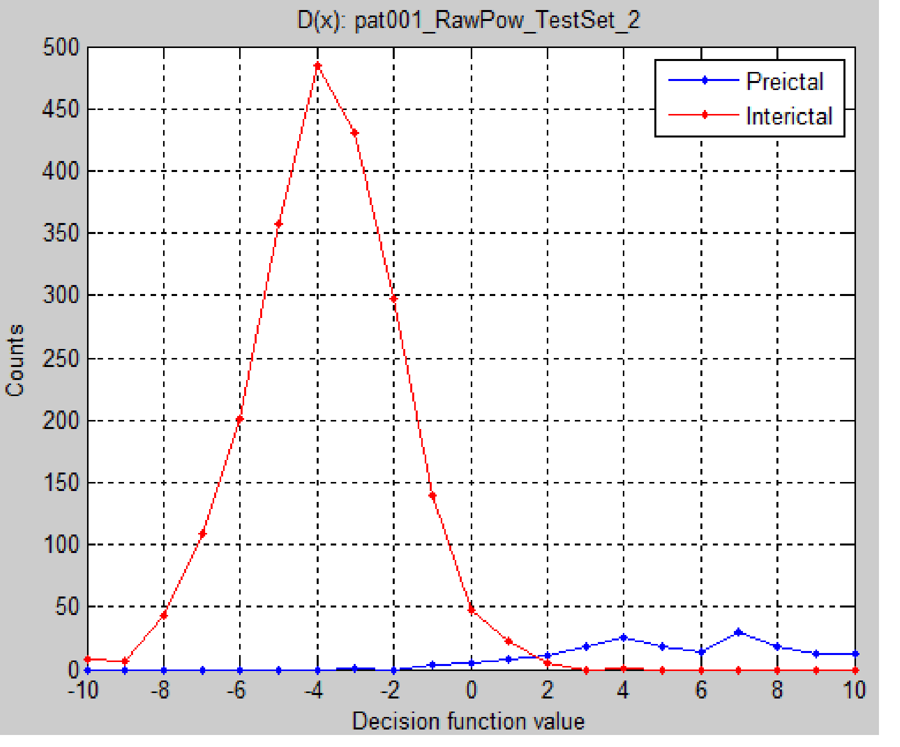
NPV=0.995

PPV= 0.371

SVM Modeling Results via projections

Patient 2: Training data and Test data





TP=500	FP=144	TP=173	FP=43
FN=37	TN=6318	FN=6	TN=2111

→**NPV=TN/(TN+FN)=0.994** **NPV=0.997**

PPV=TP/(TP+FP)=0.776 **PPV= 0.801**

Online Fraud Detection (D. Chetty 2003)

- Background on fraud detection
  - On-line transaction processing
  - Anti-Fraud strategies

Note: this description may be outdated

- Learning problem set-up (formalization)

- Modeling results

Background on fraud detection

- Historical Perspective

  - *mail order* (Sears, JC Penney catalogs)

  - *physical transactions* (using credit cards)

  - *telephone or on-line* transactions

- Legal liability due to fraud: 3 players ~ *customer, retailer, bank* (credit card issuer)
- Assumption of Risk

  traditional retail: bank is responsible

  e-commerce: e-tailer assumes the risk

Anti-Fraud Strategies

- **Balance between**

  - losing money due to fraud;

  - losing/ alienating customers;

  - increasing administrative costs

- **Two main strategies**

  - Fraud prevention (during the checkout)

  - Fraud detection (after the checkout)

Fraud Prevention

Steps during the checkout include:

- ***card authorization*** *(from a bank)*

  - ensures that the credit card has not been reported as lost or stolen
- ***cardholder authentication***
- ***address verification***

  - via Address Verification System (AVS)

  BUT AVS not effective (~ 60% mismatch rate for all transactions)

Fraud Detection (after the checkout)

Two possible approaches:

- **Rule Based Systems (RBS)**

  each transaction is compared to a number of rules. For each rule that is hit, the transaction is assigned a score. If the total fraud risk score exceeds a pre-defined threshold, the order is queued for manual review by Credit Risk Team

- **Machine Learning Approach**

  combine a priori knowledge with historical data to derive 'better rules'

Note: a priori knowledge ~ existing rules in RBS

Order processing using RBS

RBS Predicts	Credit Risk Team Decision	Final Order Status
Valid	N/A[1]	Completed[2]
Fraud	**Valid**	Completed (after reinstatement) [2]
	Fraud	Cancelled as Known Fraud[3]
	Unverifiable	Cancelled as Unverifiable[4]

1 Orders predicted as *Valid* by the RBS are fulfilled without additional review
2 True status of a completed order is known only after bank notification of settlement (*valid*) or chargeback (*fraud*).
3 *Known Fraud classification* typically occurs after the Credit Risk team communicates with the issuing bank.
4 Unverifiable orders are not relevant to this learning problem formulation as we may never know their true nature

Learning Problem Specs

Classification problem set-up includes

- ***Data selection for modeling***

  - only orders classified as fraud by current RBS system

  - orders with amount under $400

  during the period November 01 to January 02

→Total 2,331 samples selected (~0.5% of total orders)

- ***Misclassification costs***

  Good *order classified as* ***fraud*** *~ $10 (5% of ave profit margin)*

  Fraud *order classified as* ***good*** *~ $200*

Misclassification costs		Actual	
		Fraud	Valid
Predicted	Fraud	$0	$10
	Valid	$200	$0

- ***Prior probabilities***

  for training data ~ 0.5 for each class

  for future (test) data: 0.005 fraud, 0.995 valid

Feature Selection

- **Expert Domain Knowledge**

  input features ~ RBS rules (typically binary features)
- **Feature selection** (dimensionality reduction)

  via simple correlation analysis,

  i.e. pairwise correlation between each input feature and the output value (valid or fraud).
- **Common-sense encoding of some inputs**

  i.e. all email addresses email addresses aggregated into whether or not it was a popular domain (e.g., yahoo.com)
- ***All final inputs*** turned to be **binary categorical**

  (see next slide)

Feature	Description	Domain
High Risk AVS	True for an Address Verification System code of N, 11, 6, U, or U3	Yes, No
High Risk State	True for a ship-to state of CA, NY or FL	Yes, No
Popular Domain	True for a popular email domain (yahoo,hotmail)	Yes, No
High Risk Creation Hour	True for orders submitted between the hours of 10pm and 6 am.	Yes, No
High Risk Address	True for orders that have a ship-to address that is identified as high risk	Yes, No
Ship To Velocity rule	True if the same ship-to address has been used often in a time period	Yes, No
Expedited Shipping rule	True if Next Day shipping is requested for the order.	Yes, No
Customer ID Velocity rule	True if the same customer ID has been used often in a single time period.	Yes, No
High Risk Zipcode rule	True for orders that have a ship-to zip code that is identified as high risk by BestBuy.com.	Yes, No
Credit Card Velocity Rule	True if the same credit card has been used often in a single time period	Yes, No
Bill To Ship To Rule	True if the shipping address does not match the billing address on file for the credit card.	Yes, No
Subcat Rule	True if an order line item belongs to a high risk category, e.g., laptops.	Yes, No
HRS Rule	True if a BestBuy.com credit card is being used for the first time to make a purchase.	Yes, No
Order Amount Class	Range (in hundreds) within which the order total falls.	0,1,2,3
AVS Result	Code returned by the Address Verification System for the customer’s billing address.	X, Y, A, W, Z, U
Creation Hour	The hour of the day when the order was submitted on the online store.	0, 1, 2,…23

Comparison Methodology

- **Classification Methods**
  CART, k-NN, SVM classifier
- **Available Data**
  → Training(67%) + Test (33%)
- **Model selection**
  via 5-fold cross-validation on training data
- **Prediction accuracy**
  measured on the test set

Summary of Modeling Results

Test Set	Classific. Accuracy - Fraud	Classific. Accuracy - Valid	Classific. Accuracy - Overall
Rule Based System	72.43%	50.69%	59.46%
k-NN (k=13)	85.47%	83.50%	84.68%
CART (Entropy)	87.82%	82.20%	85.59%
SVM (RBF kernel, with Gamma = 0.3, C = 3)	86.38%	84.91%	85.84%

- All methods performed better than RBS
- Most important factor is feature selection rather than classification method used

OUTLINE

- Practical issues for SVM classifiers
- Univariate histograms for SVM classifiers
- SVM model selection
- Application studies
- **Summary ~ *Importance of:***
- *Sound problem formalization*
- *Data preprocessing/ encoding/ feature selection*
- *Model selection for different SVM problems*
- *Histogram of projections (for SVM classifiers)*

  - can be extended to other SVM problems