SVM in Oracle Database 10g: Removing the Barriers to Widespread Adoption of Support Vector Machines

Boriana Milenova, Joseph Yarmus, Marcos Campos
Data Mining Technologies
Oracle

ORACLE

Overview

- Support Vector Machines fundamentals
- Hurdles to widespread SVM adoption
  - Usability
  - Scalability
- Oracle’s solutions for productizing SVM

Data Mining in RDBMS

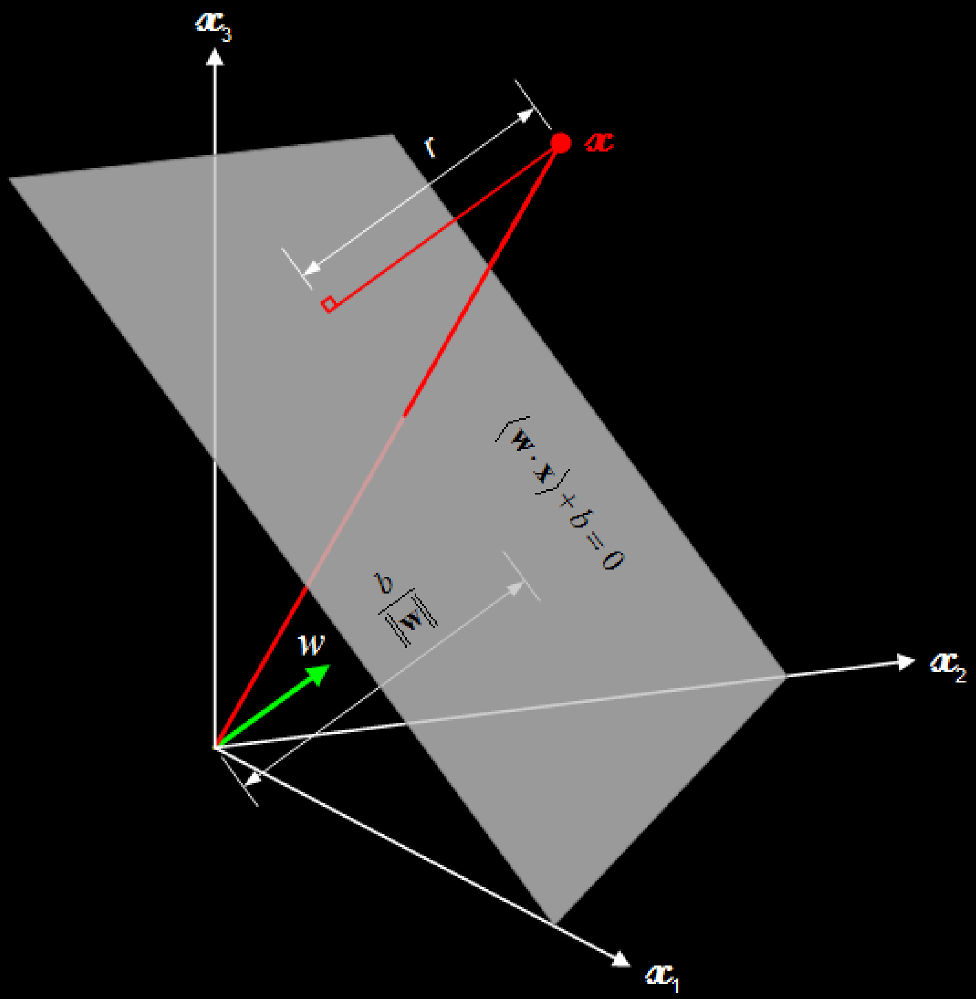
- Growing importance of analytic technologies
  - Large volumes of data need to be processed/analyzed
  - Modern data mining techniques are robust and offer high accuracy
- Challenges of data mining
  - Complex methodologies
  - Computationally intensive

Why SVM?

Powerful state-of-the-art classifier

- Strong theoretical foundations
  - Vapnik-Chervonenkis (VC) theory
- Regularization properties
  - Good generalization to novel data
- Algorithm of choice for challenging high-dimensional data
  - Text, image, bioinformatics

Conceptual Simplicity



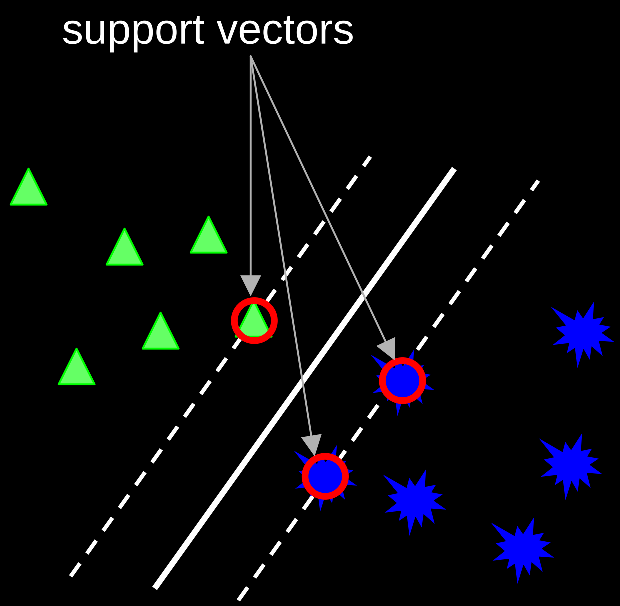
An SVM model defines a hyperplane in the feature space in terms of coefficients (**w**) and a bias term (*b*)

Prediction:

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

SVM Optimization Problem: Linearly Separable Case

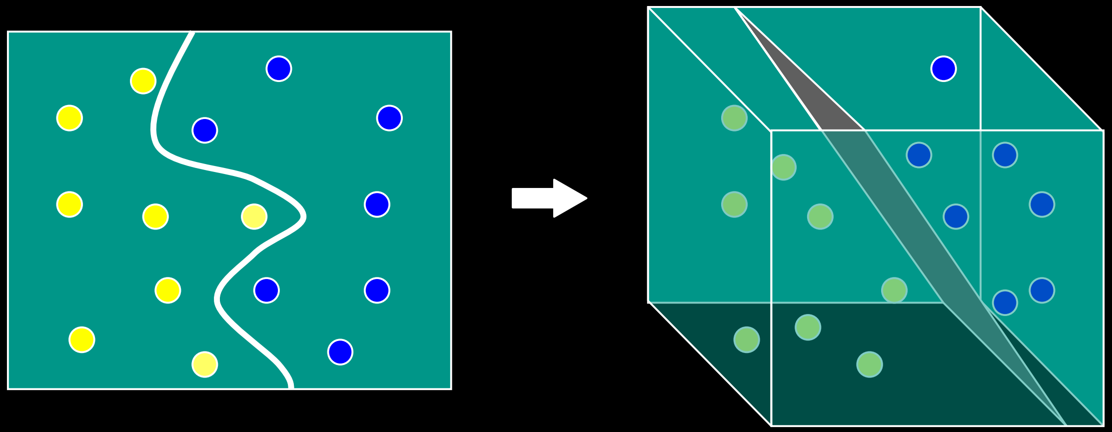
Minimize $L_p(\mathbf{w}) = \frac{1}{2}\langle \mathbf{w} \cdot \mathbf{w} \rangle$, subject to $y_i \left(\langle \mathbf{w} \cdot \mathbf{x}_i \rangle + b \right) \geq 1$



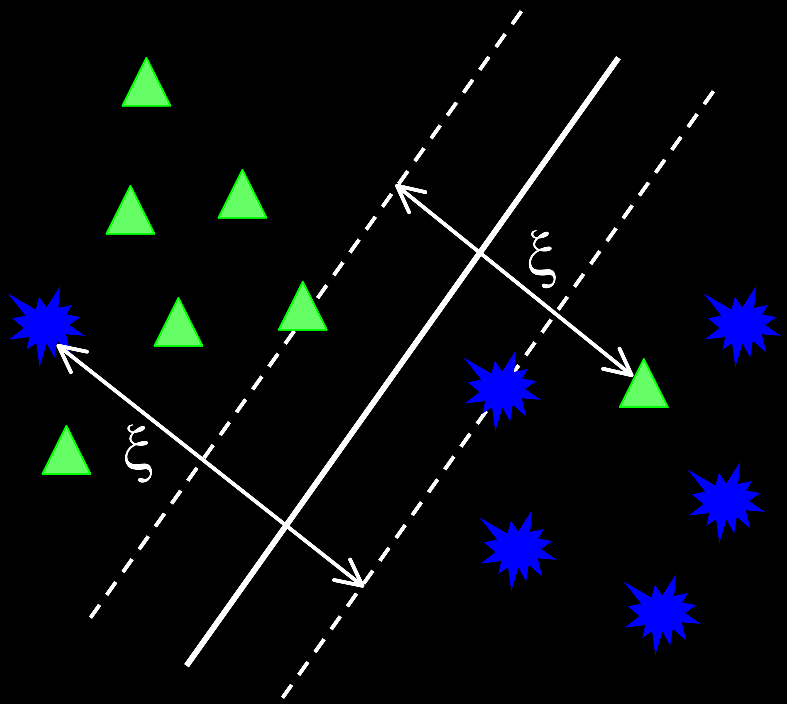
- Maximum separation between classes
- Dimensionality insensitive
- Sparse solution
- Single global minimum
- Solvable in polynomial time…

Kernel Classifiers

1. Transform data via non-linear mapping to an inner product feature space
   - Gaussian, polynomial kernels
2. Train a linear machine in the new feature space



SVM Soft Margin Optimization: Non-Separable Case

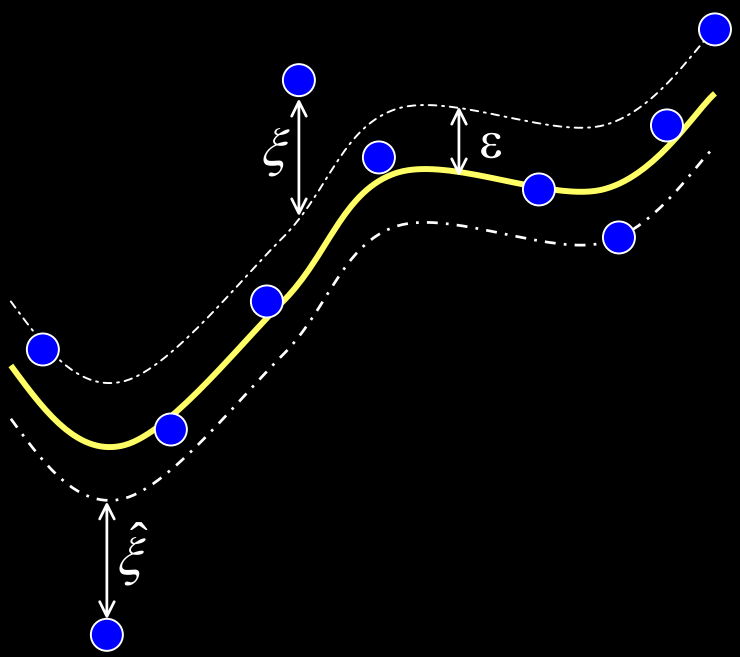


Capacity parameter *C* trades off complexity and empirical risk

$$L_p(\mathbf{w}) = \frac{1}{2}\langle \mathbf{w} \cdot \mathbf{w} \rangle + C \sum \xi^k$$

subject to $y_i\left(\langle \mathbf{w} \cdot \mathbf{x}_i \rangle + b\right) \geq 1 - \xi_i$

SVM Regression



ε-insensitive loss function

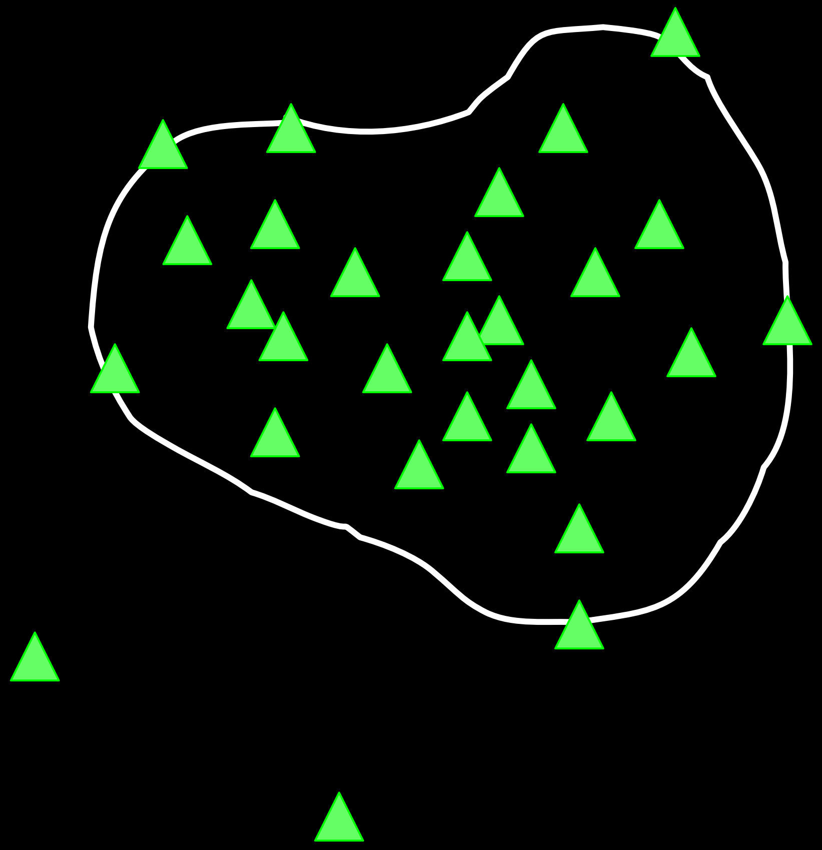
$$L_p(\mathbf{w}) = \frac{1}{2}\langle \mathbf{w} \cdot \mathbf{w} \rangle + C \sum (\xi^k + \hat{\xi}^k)$$

subject to

$$\left(\langle \mathbf{w} \cdot \mathbf{x}_i \rangle + b\right) - y_i \leq \varepsilon + \xi_i$$

$$y_i - \left(\langle \mathbf{w} \cdot \mathbf{x}_i \rangle + b\right) \leq \varepsilon + \hat{\xi}_i$$

One-Class SVM



- Outlier detection
  - Typical cases vs. outliers
- Discrimination between a known class and the unknown universe of counterexamples

SVM in the Database

- Oracle Data Mining (ODM)
  - Commercial SVM implementation in the database
  - Product targets application developers and data mining practitioners
  - Focuses on ease of use and efficiency
- Challenges
  - Good out-of-the-box accuracy
  - Good scalability
    - large quantities of data, low memory requirements, fast response time

SVM Accuracy: User Impact

Inexperienced users can get dramatically poor results

	Naive user accuracy	Expert user accuracy
Astroparticle Physics	0.67	0.97
Bioinformatics	0.57	0.79
Vehicle	0.02	0.88

Tricks of the Trade for Improving SVM Accuracy

- Data preparation
  - Outlier removal
  - Scaling
  - Categorical to numeric attribute recoding
- Parameter estimation (model selection)
  - Grid search
  - Cross-validation
  - Heuristics
  - Gradient descent optimization

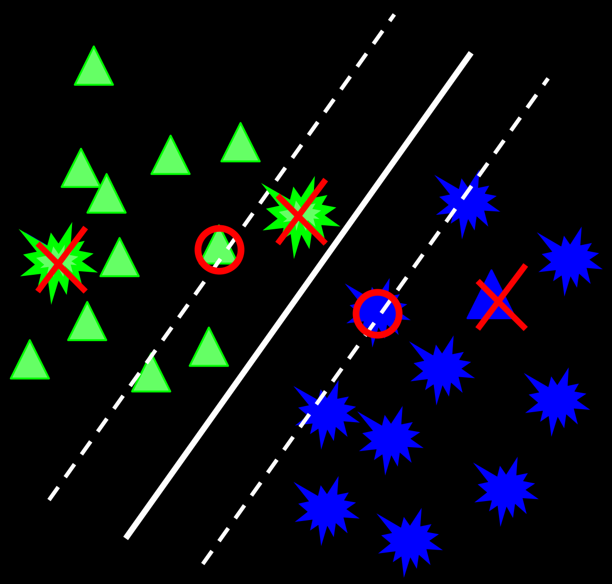
Oracle's Data Preparation Support

- Automatic data preparation
  - Outlier removal
  - Scaling
  - Categorical to numeric attribute recoding
- Supported by
  - `dbms_data_mining_transform` package
  - Oracle Data Miner

Oracle's On-the-Fly SVM Parameter Estimation

- Data-driven
- Low computational cost
- Ensure good generalization
  - Avoid overfitting
    - model is too complex and data is memorized
  - Avoid underfitting
    - model is not complex enough to capture the underlying structure of the data

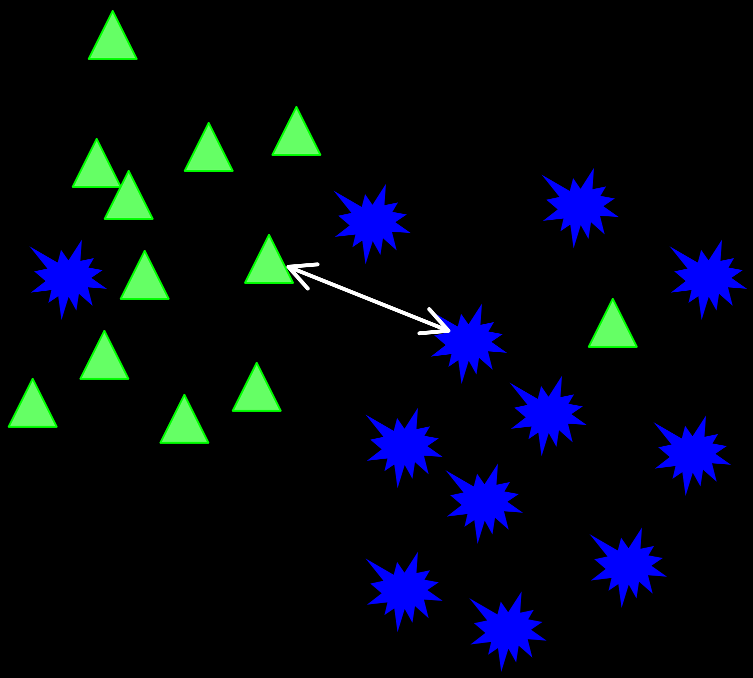
Classification Capacity Estimate



Goal: Allocate sufficient capacity to separate typical examples

1. Pick *m* random examples per class
2. Compute f_i assuming $\alpha = C$
   $$f_i = \sum_{j=1}^{2m} C y_j K(\mathbf{x}_j, \mathbf{x}_i)$$
3. Exclude noise (incorrect sign)
4. Scale *C*, $f_i = \pm 1$ (non bounded sv)
   $$C^{(i)} = sign(f_i) / \sum_{j=1}^{2m} y_j K(\mathbf{x}_j, \mathbf{x}_i)$$
5. Order ascending
6. Select 90th percentile

Classification Standard Deviation Estimate



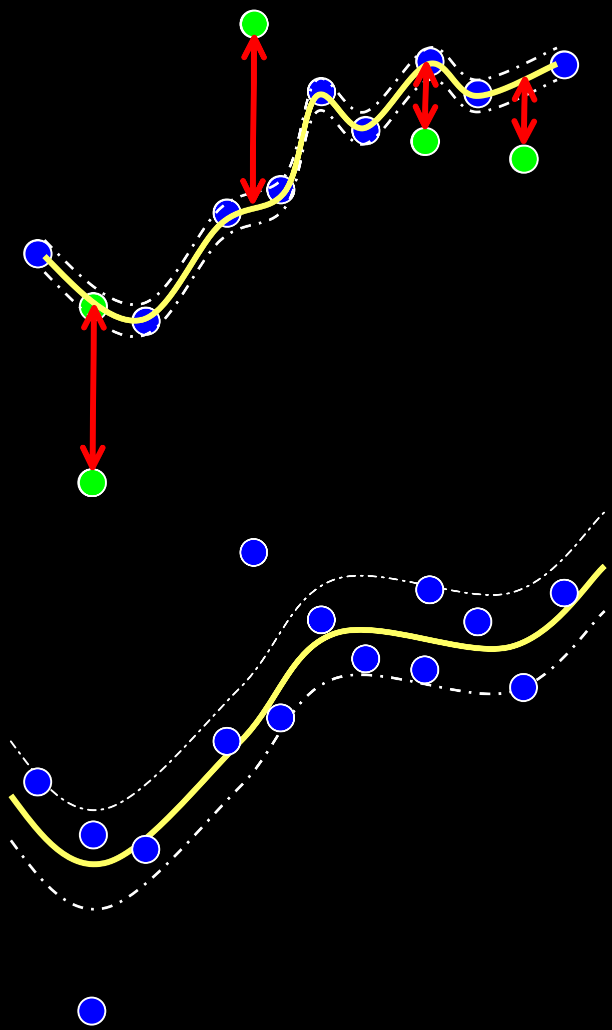
Goal: Estimate distance between classes

1. Pick random pairs from opposite classes
2. Measure distances
3. Order descending
4. Select 90th percentile

Classification Comparison

	Naive user	Grid search + xval	Oracle
Astroparticle Physics	0.67	0.97	0.97
Bioinformatics	0.57	0.85	0.84
Vehicle	0.02	0.88	0.71

Epsilon Estimate



Goal: estimate target noise by fitting preliminary models

1. Pick small training and held-aside sets
2. Train SVM model with $\varepsilon = 0.01 * \mu_y$
3. Compute residuals on held-aside data
4. Update $\varepsilon^{new} = \left(\varepsilon^{old} + \mu_r\right)/2$
5. Retrain

Regression Comparison

	Grid search RMSE	Oracle RMSE
Boston housing	6.26	6.57
Computer activity	0.33	0.35
Pumadyn	0.02	0.02

SVM Scalability Issues

- Build scalability
  - Quadratic scalability with number of records
  - Feasible for small/medium datasets
- Scoring scalability
  - Large model sizes (non-linear kernels) make online scoring impractical

Scalability Improvements

- Popular build scalability techniques
  - Chunking and decomposition
  - Working set selection
  - Kernel caching
  - Shrinking
  - Sparse data encoding
  - Specialized linear model representation

However, these standard techniques are usually not sufficient…

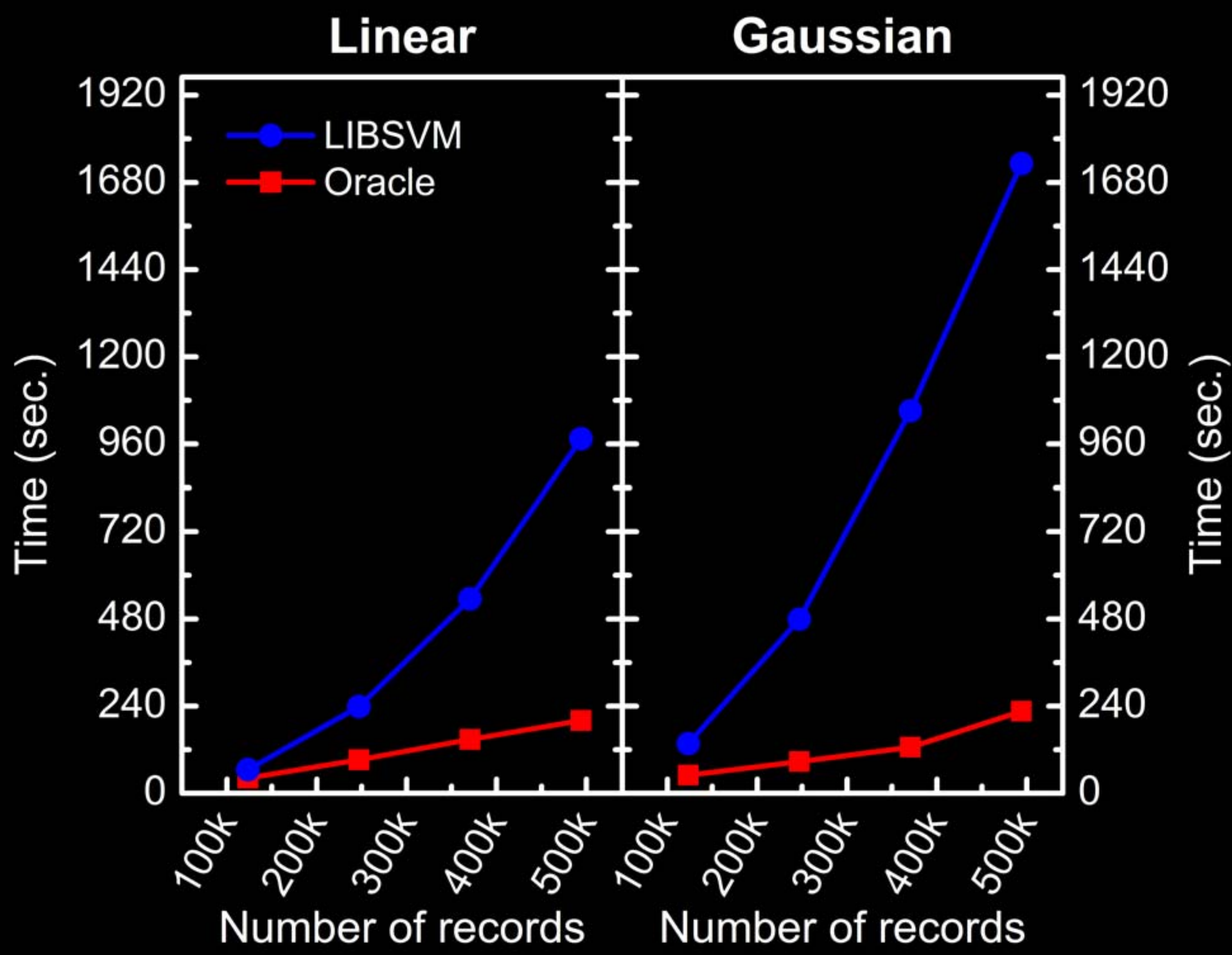
Oracle's Additional Scalability Improvements

- Stratified sampling
  - Classification and regression
  - Single pass through the data
- Working set selection
  - Smooth transitions between working sets
  - Faster convergence
  - Computationally efficient

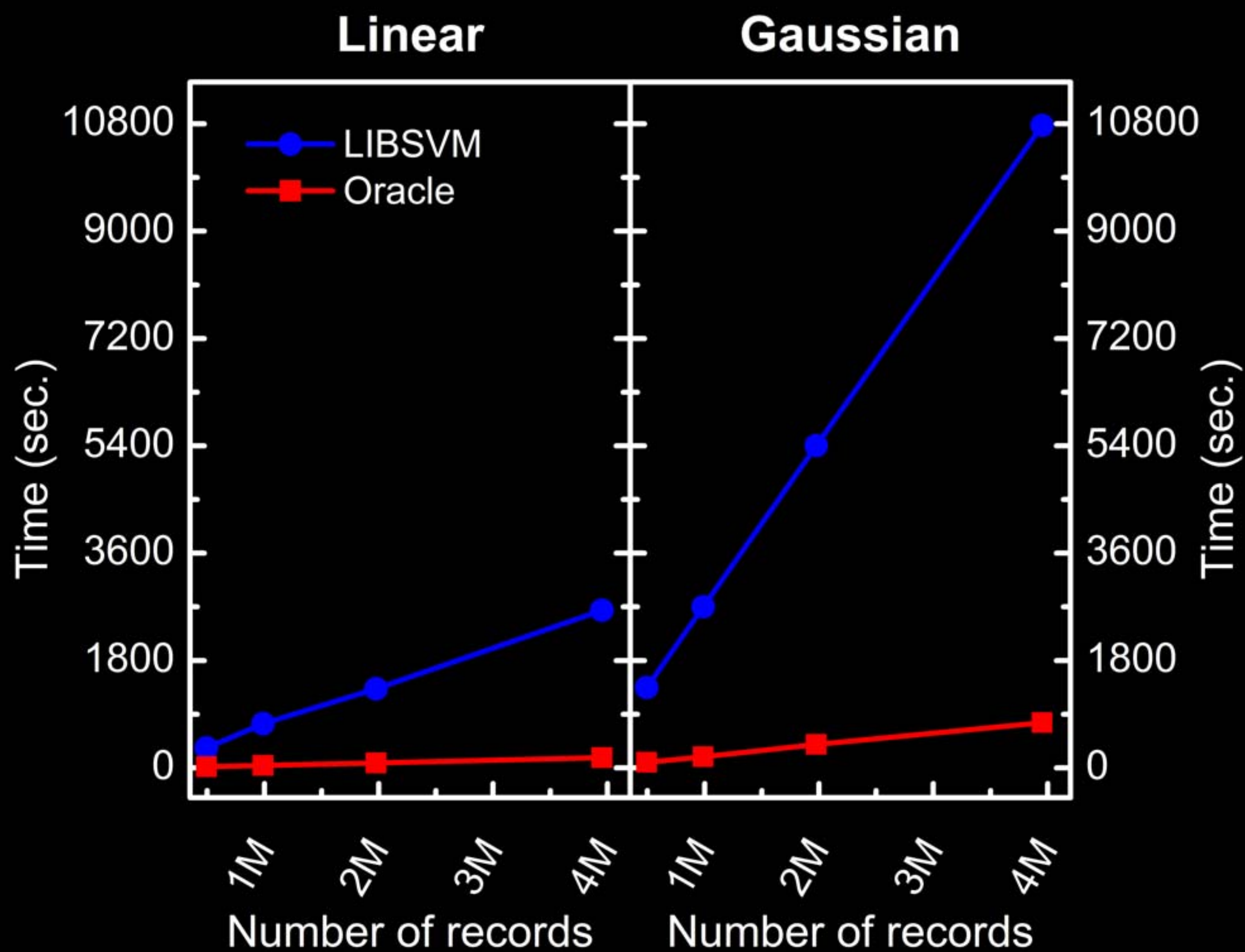
Oracle's Additional Scalability Improvements (cont.)

- Reduced model size
  - Specialized linear representation
  - Active learning for non-linear kernels
    1. Construct a small initial model
    2. Select additional influential training records
    3. Retrain on the augmented training sample
    4. Exit when the maximum allowed model size is reached

Build Scalability Results



Scoring Scalability Results



Oracle Scoring Time Breakdown

Linear classification model

	50K	1M	2M	4M
SVM scoring (sec)	18	37	71	150
Persistence (sec)	2	4	11	22

SVM Scoring as a SQL Operator

- Easy integration
  - DML statements, subqueries, functional indexes
- Parallelism
- Small memory footprint
  - Model cached in shared memory
- Pipelined operation

```
SELECT id, PREDICTION(svm_model_1 USING *)
FROM user_data
WHERE PREDICTION_PROBABILITY(svm_model_2,
     'target_val' USING *) > 0.5
```

Conclusions

- Implementing an SVM tool with an adequate level of usability and performance is a non-trivial task
- Oracle's SVM implementation allows database users with little data mining expertise to achieve reasonable out-of-the-box results
  - Corroborated by independent evaluations by the University of Rhode Island and the University of Genoa

Final Note

- SVM is available in Oracle 10g database
  - Implementation details described here refer to Oracle 10g Release 2
  - JAVA (J2EE) and PL/SQL APIs
  - Oracle Data Miner GUI
- Oracle's SVM has been integrated by ISVs
  - SPSS (Clementine)
  - InforSense KDE Oracle Edition

ORACLE®