SPARSE KERNEL-BASED MODELS FOR SPEECH RECOGNITION

P. Karsmakers

Promotors:
Prof. Dr. Ir. J. Suykens
Prof. Dr. Ir. H. Van hamme

Public Ph.D. Defense
May 2010

Outline

- Introduction
  - Automatic speech recognition
  - Kernel methods
  - Sparse models
  - Motivation for kernel methods in automatic speech recognition
  - Model inference
  - Challenges and objectives
  - Main contributions
- Fixed-size Multi-class Kernel Logistic Regression
- Sparse Conjugate Directions Pursuit with Kernels
- Segment-based Phone Recognition using Kernel Methods
- Conclusions
- Future Work

1. Introduction: Automatic speech recognition

- Automatic Speech Recognition (ASR) = technology for converting an acoustic speech signal into a sequence of words by means of computer program

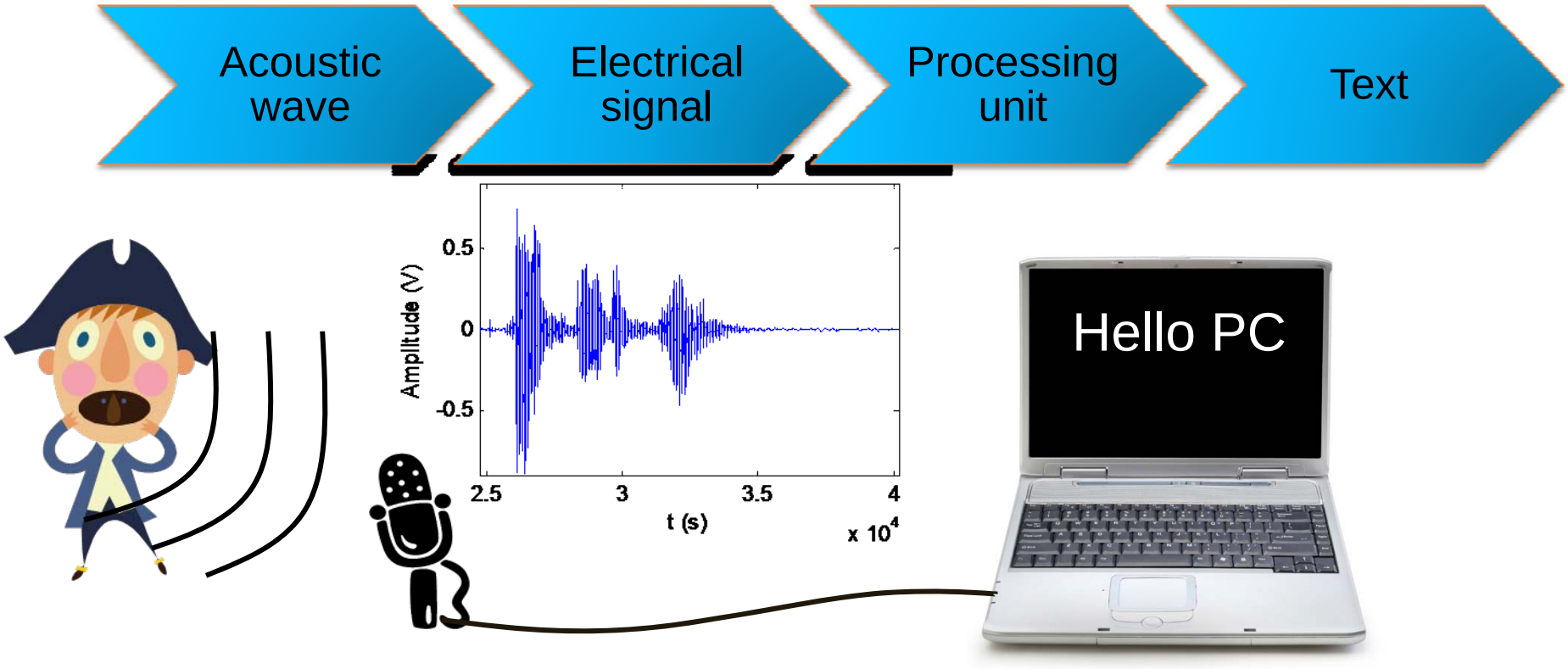
- Many applications examples, e.g.:
  - Health:
    - assistive technology: e.g. enable deaf people to understand spoken words, voice controlled home automation for people with mobility disabilities
  - Consumer electronics:
    - data entry and dictation
    - small mobile devices (e.g. smartphones) with voice dialing, voice controlled user interface

Different levels of complexity

- **Speech vocabulary:** the list of words which might be pronounced
- **Isolated or Continuous mode:** user clearly indicates word boundaries or not
- **Speaker dependent or independent:** system is developed for a single speaker or can be used by any speaker for a given language
- **Adverse environments:** mismatch between development and operational environment
- **Adaptivity:** ability of system to adjust to varying operating conditions

Experiments in this work are on continuous non-adaptive speaker-independent task in clean conditions

From acoustic wave to text



How to relate time-domain speech signals to text?

Acoustic model

- Relation too complex to be described by a set of rules, e.g.:

```
IF duration(x) < 1ms THEN y='hi'
IF max_amplitude(x) = 1 THEN y='peter'
…
```

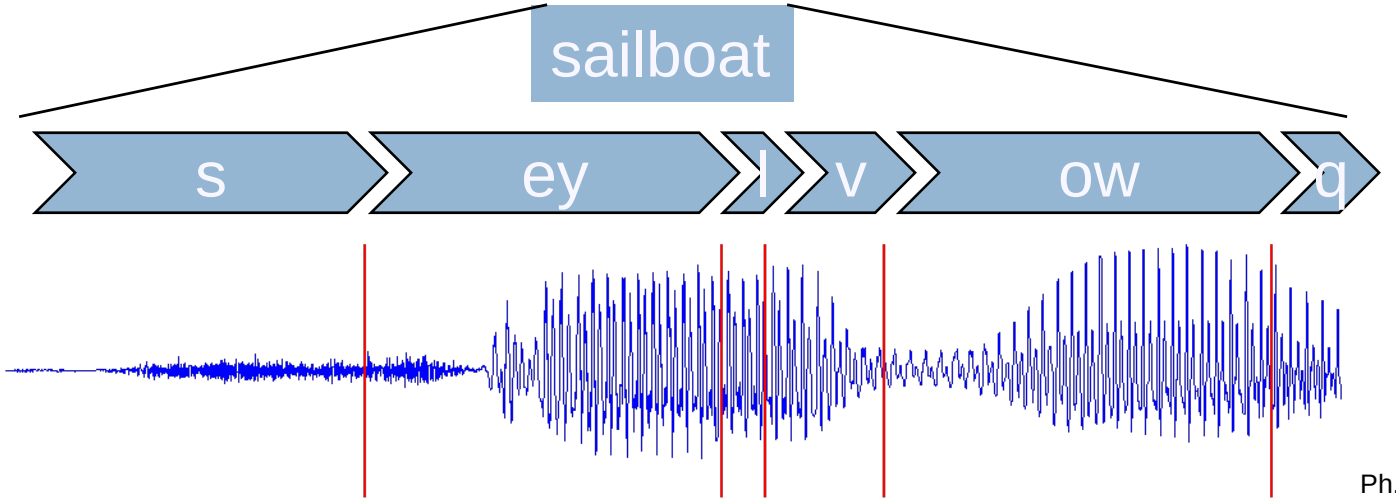
- Instead, let computer search for a (statistical) relation $y = f(x)$, acoustic model, based on annotated examples, e.g.:

x=Electrical signal	y=text
x_1= 	y_1='hello'
x_2= 	y_2='world'
x_3= 	y_3='steve'
…	…

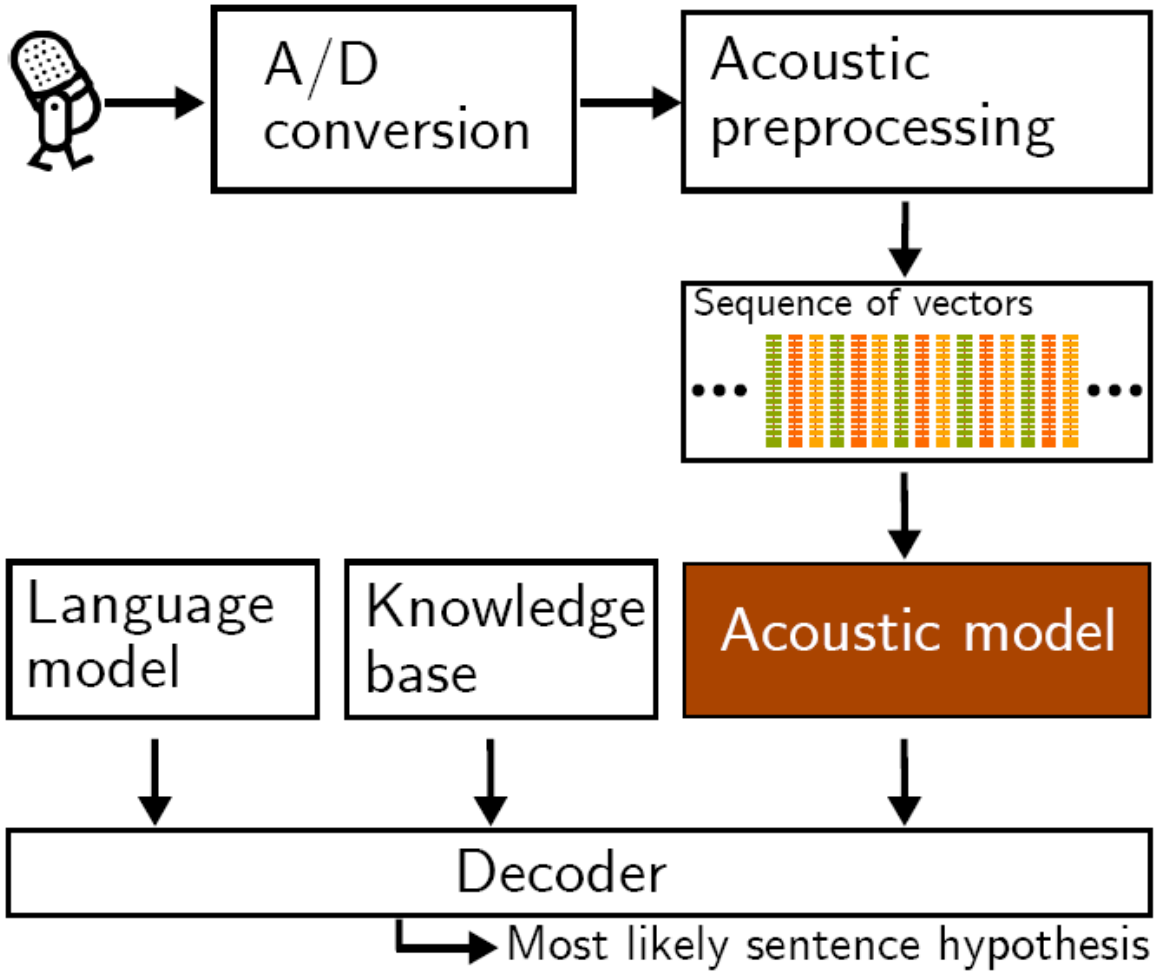
MACHINE LEARNING problem

Recognition of subwords

- Problem:
  - To obtain a good relation (model): each word needs a sufficient amount of examples. ±240.000 dutch words in Van Dale hence a lot of examples.
  - New words might appear
- Instead of modeling words directly use subwords (e.g. phone).
- Phone = smallest subword which has meaningful contrasts between utterances (about 50 phones)
- All words can be expressed by a combination of phones.

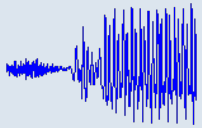
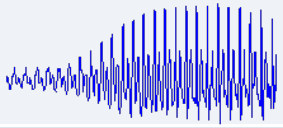
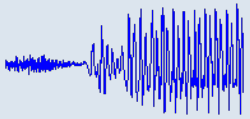


Classical speech recognizer architecture

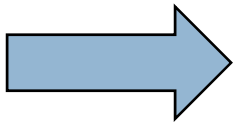


1. Introduction: Kernel-based models

- Learning an acoustic model = machine problem
- Kernel-based methods are specific family
- Speech classification example

x=Electrical signal	y=text
x_1= 	y_1='ey'
x_2= 	y_2='ow'
x_3= 	y_3='ey'
…	…

Different representation

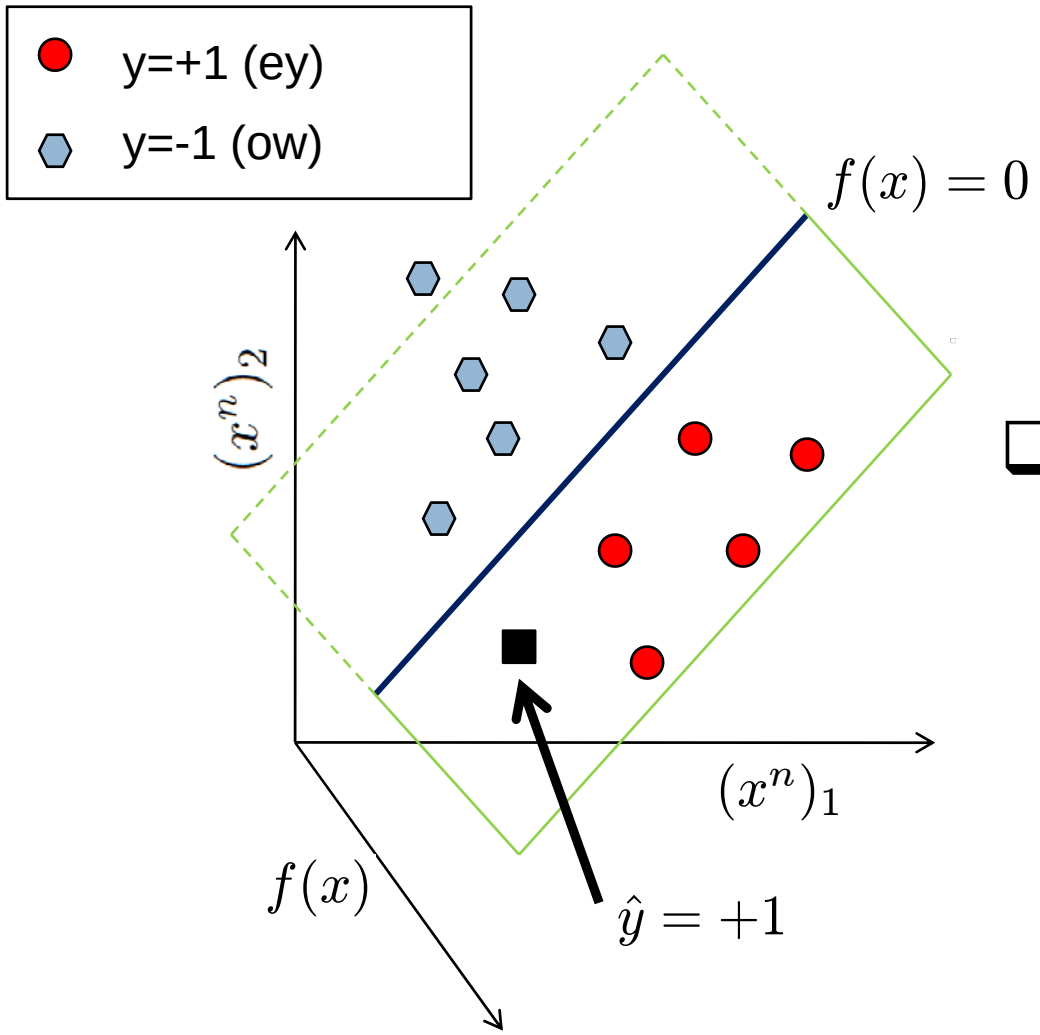


x^n	y^n
$x^n_1=F(x_1)$	$y^n_1=+1$
$x^n_2=F(x_2)$	$y^n_2=-1$
$x^n_3=F(x_3)$	$y^n_3=+1$
…	…

- E.g.

$$F(x) : \mathbb{R}^M \rightarrow \mathbb{R}^2, F(x) = (mse(x), duration(x))^T$$

Linear classification



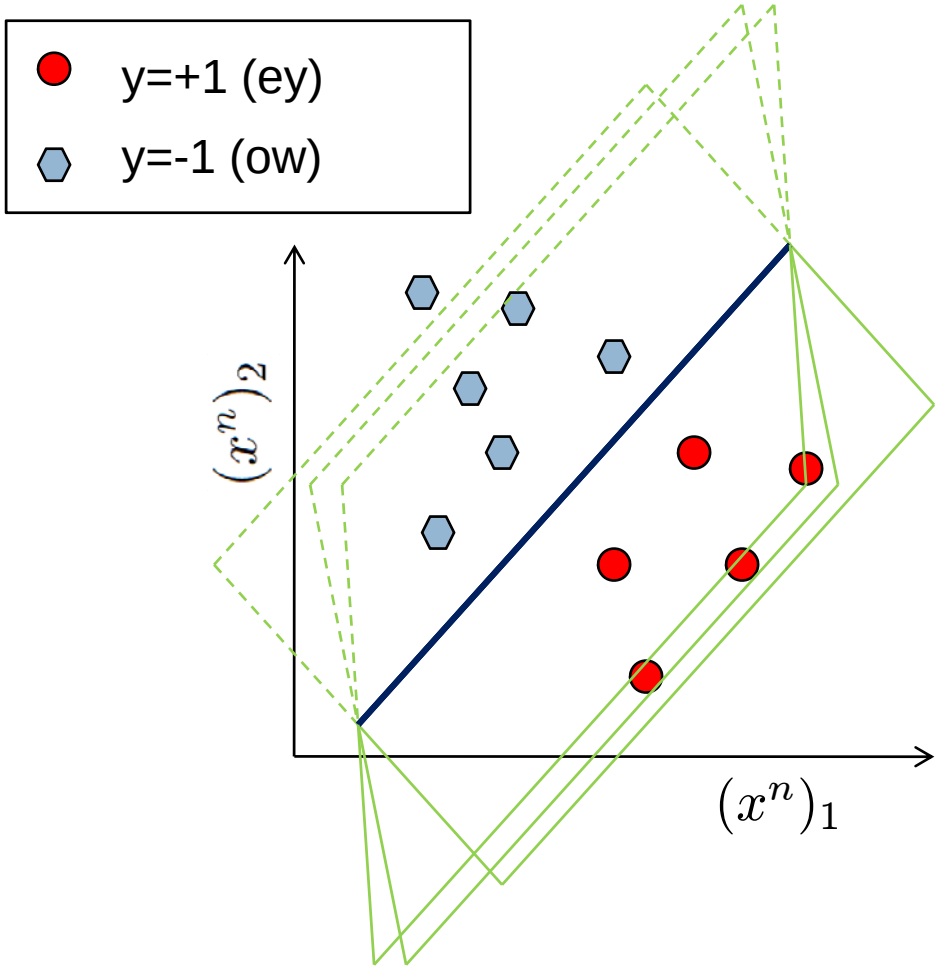
- Since classes are lineary separable, use linear model

$$f(x) = (w)_1(x)_1 + (w)_2(x)_2 + b$$

- Classify new point x* (■)

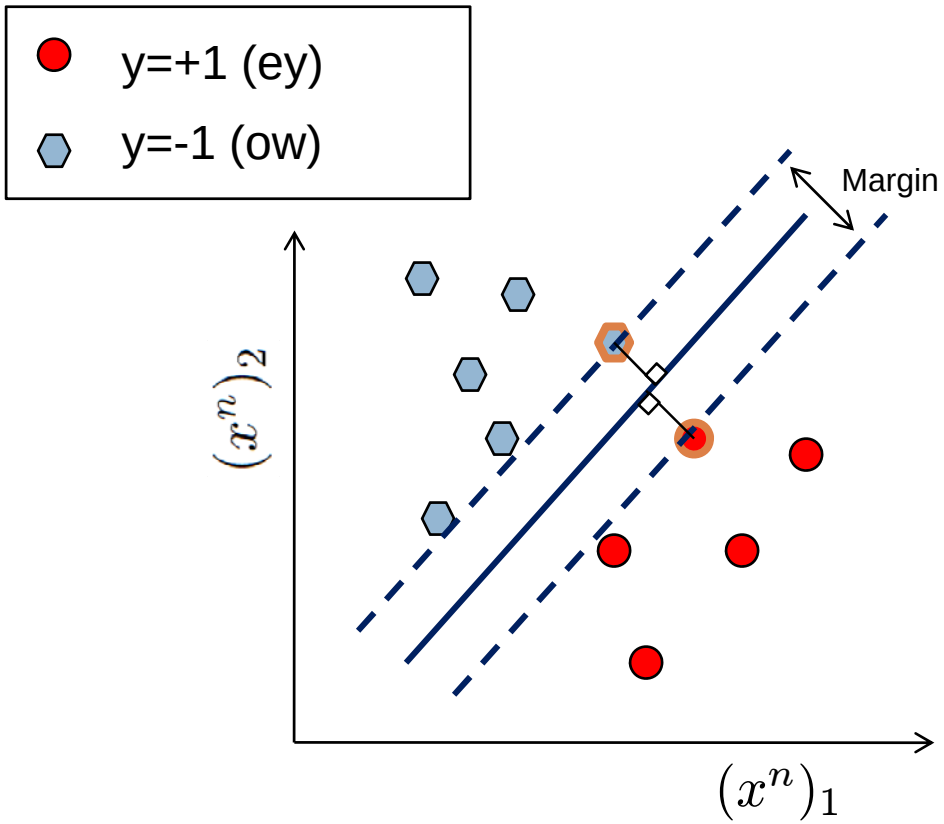
$$\hat{y} = \text{sign}(f(x))$$

Linear classification



- Which hyperplane to select?

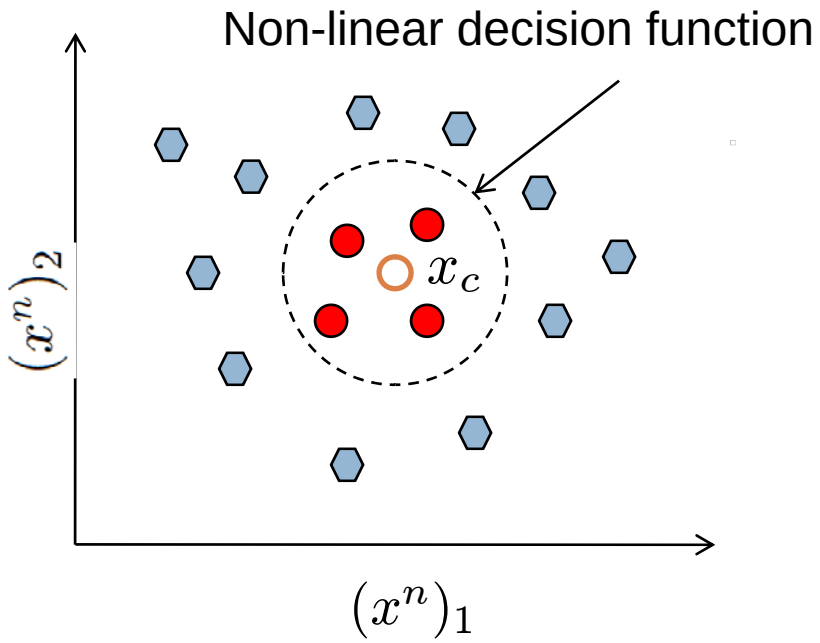
Linear classification



- Which hyperplane to select?
- Popular criterion → maximize the margin

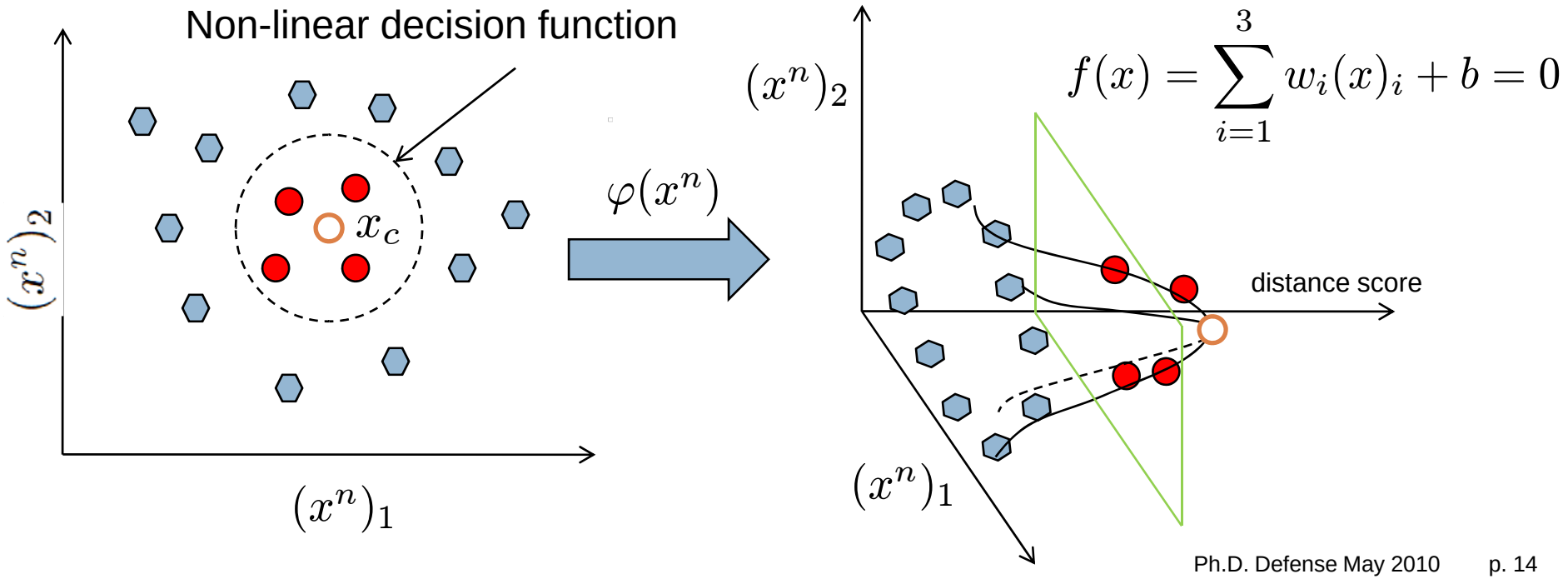
Non-linear classification

- What if data not linearly separable?



Non-linear classification

- What if data not linearly separable?
- Idea: first transform input data to higher dimensional space, then do linear classification
- E.g. use fixed mapping function $\varphi(x^n) = \left((x^n)_1, (x^n)_2, e^{\frac{-\sum_{i=1}^{2} ((x^n)_i - (x_c)_i)^2}{2\sigma^2}} \right)^T$



Kernel methods

Feature map

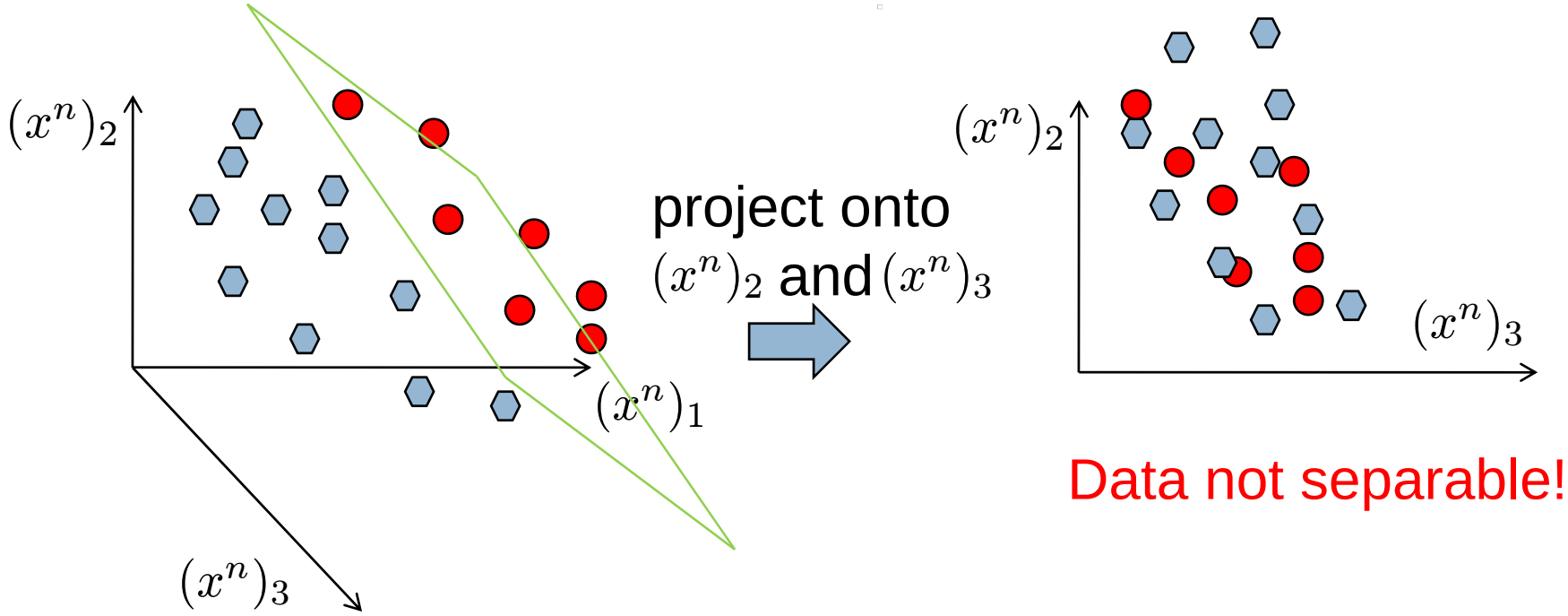
- Kernel methods implicitly define a feature map using kernel function $K(x,x')=\varphi(x)^T\varphi(x')$
- Reformulate learning problem such that input data only appears in a dot-product
- Change a single kernel parameter → different feature
- Popular kernels: RBF kernel, polynomial kernel, linear kernelmap

Linear classification in high dimensional space

- In higher dimensional space use well-understood algorithms to discover linear relations
- Popular implementations: Support Vector Machines (SVMs) and Least-Squares Support Vector Machines (LS-SVMs)

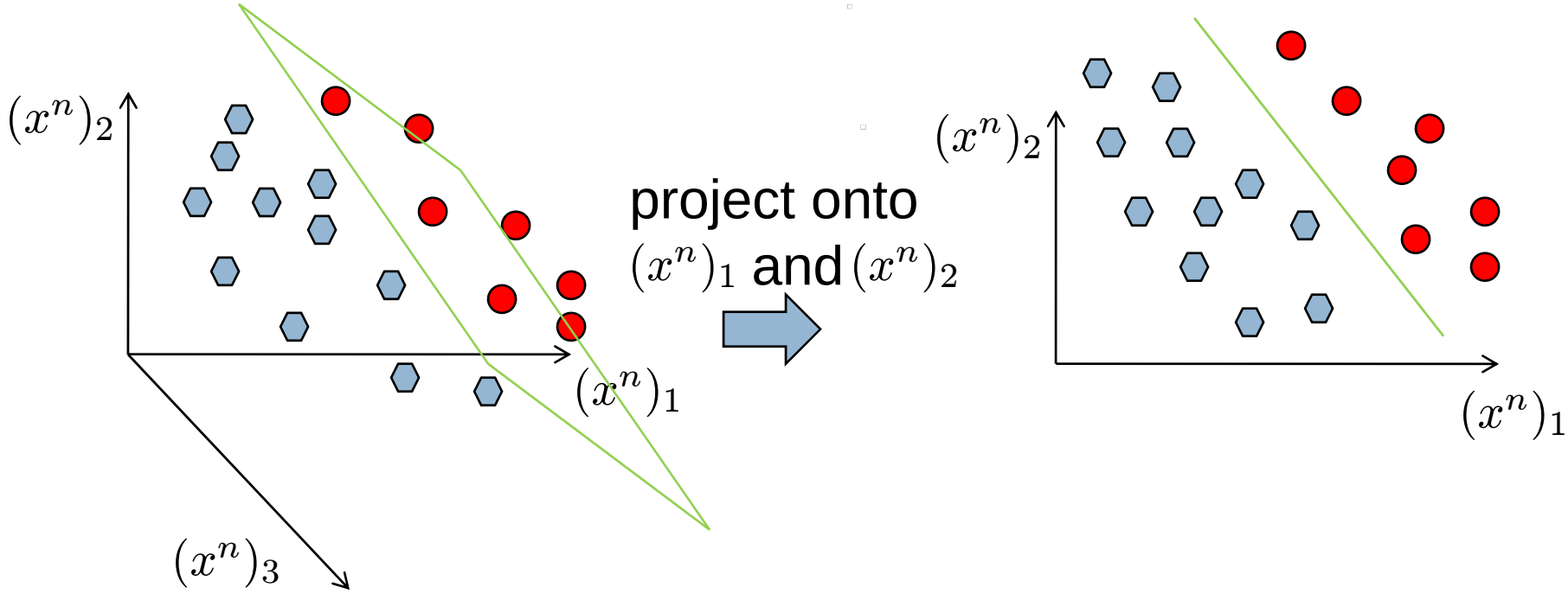
1. Introduction: Sparse models

- Keep the complexity of the model as low as possible.
- This might be obtained by setting a subset of the model parameters to zero → ignore corresponding data dimensions
- The more parameters are zero the sparser the model



1. Introduction: Sparse models

- Keep the complexity of the model as low as possible.
- This might be obtained by setting a subset of the model parameters to zero → ignore corresponding data dimensions
- The more parameters are zero the sparser the model



1. Introduction: Sparse models

- Without $(x^n)_3$ data is also separable
- No discriminative information in dimension $(x^n)_3$
- Set corresponding model parameter to zero
- $f(x) = (w)_1(x)_1 + (w)_2(x)_2 + 0.(x)_3 + b$, third dimension is ignored
- We have a sparse model!

1. Introduction: Model inference

- Consider our previous data set $\mathcal{D} = \{x_i^n, y_i^n\}_{i=1}^N$ (*N*=#examples)
- How to find the optimal model parameters?
  - Choose learning objective e.g.

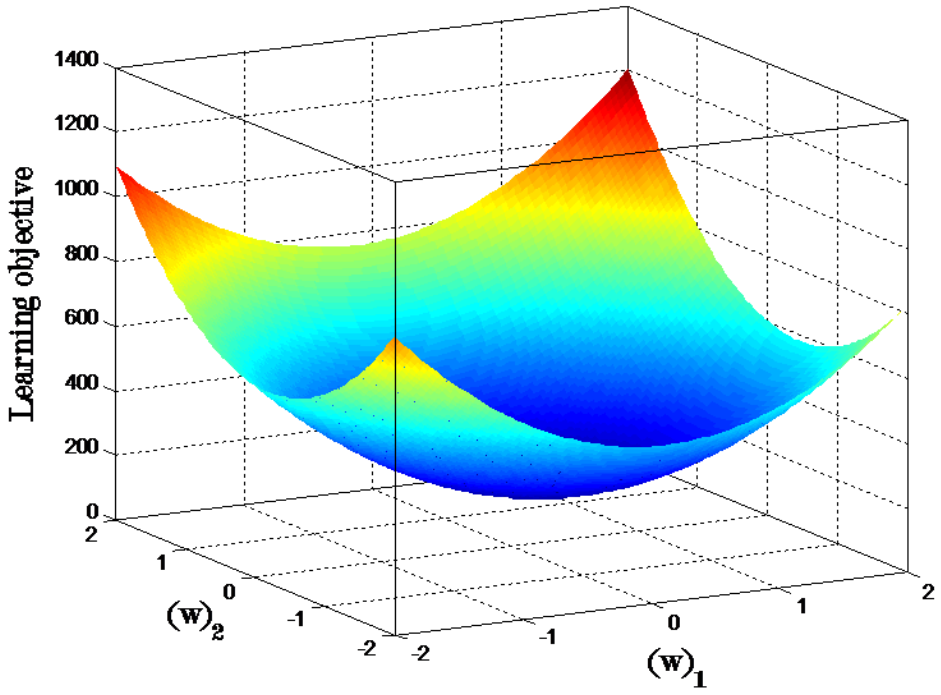
$$\min_{w} \sum_{i=1}^{N} [(w)_1(x_i)_1 + (w)_2(x_i)_2 + b - y_i]^2$$

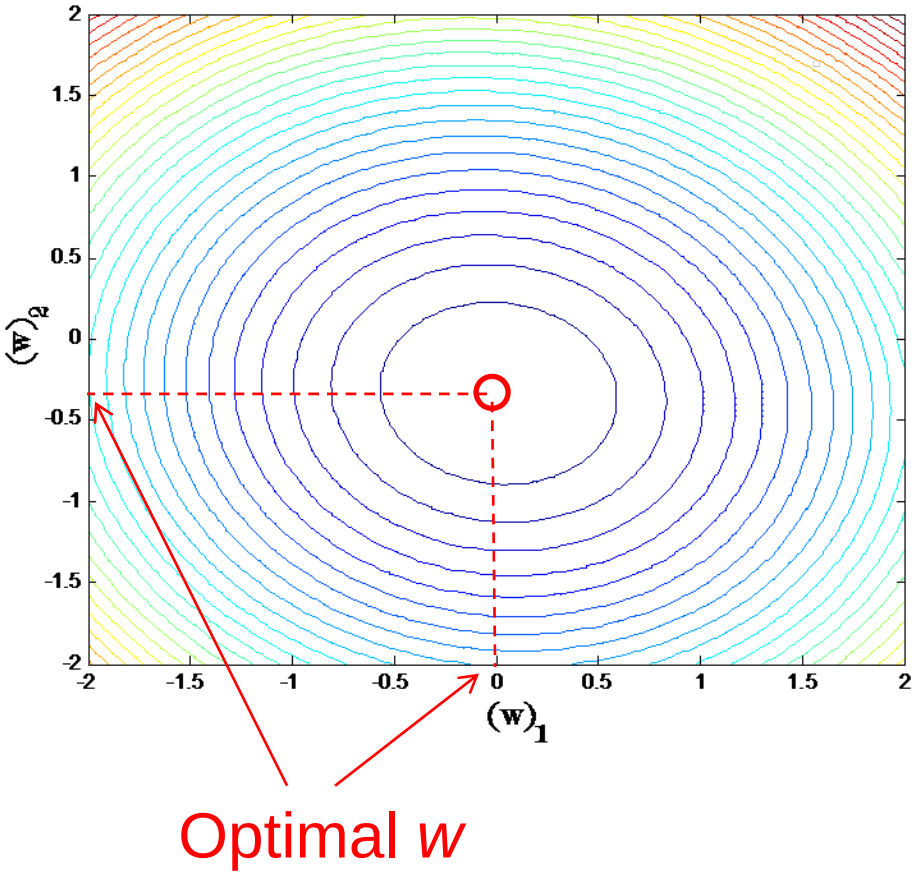
$f(x_i)$

  - Per parameter combination $w = ((w)_1, (w)_2)^T$, a “badness” score
  - Try different parameter combinations in an intelligent manner
  - Select that combination with the lowest “badness” score
  - → optimization

1. Introduction: Model inference

- Visualize learning objective surface $\min_{w} \sum_{i=1}^{N} [(w)_1 (x_i)_1 + (w)_2 (x_i)_2 + b - y_i]^2$





Kernel Logistic Regression

- Equivalent dual formulation [*Karsmakers et al, 2007*]:
  - using Newton's method,
  - and standard LS-SVM approach

Dual formulation: solve in each iteration *k*

$$\left(\Omega + \frac{1}{\nu} G^{(k)^{-1}}\right) \alpha^{(k)} = \frac{1}{\nu} q^{(k)} + \Omega \alpha^{(k-1)},$$

where $\Omega_{ij} = \varphi(x_i)^T \varphi(x_j) = K(x_i, x_j)$, $G^{(k)}$ a weight matrix, and $q^{(k)}$ a regressor vector.

- Mapped input vectors only appear in dot-products
- Kernel functions can be used → non-linear classification

1. Introduction: Motivation for Kernel Methods in Automatic Speech Recognition

- **ASR problem is far from solved**
- **Kernel methods might offer the following advantages:**
  - **Convexity:**
    - Model parameters in ASR usually found using non-convex optimization → might give suboptimal results
    - Kernel methods such as SVM and LS-SVM are convex
  - **High dimensional spaces:** Kernel-based methods generalize well even in high dimensional spaces
  - **Customized kernels:** Different learning problems might be tackled using the same methodology and a different type of kernel.
  - **Success in other applications:** e.g. in bioinformatics, financial engineering, or time series prediction

1. Introduction: Challenges and Objectives

Requirements in ASR:

- Scalability: Training scales $O(N^2)$
- Sparseness: Sparser models give faster evaluation times
- Multi-class classification: Binary case typically extended with, mostly ad-hoc, approaches (e.g. one-versus-one, one-versus-all)
- Probabilistic outcomes: As a common interface in ASR a probabilistic language is normally used.
- Variable-length sequences: Mapping from variable length sequence to a fixed representation.

1. Introduction: Main contributions

- A. Large-scale Fixed-size Multi-class Kernel Logistic Regression (FS-MKLR)
  - P. Karsmakers, K. Pelckmans, H. Van hamme, J.A.K. Suykens, "***Large-Scale Kernel Logistic Regression for Segment-Based Phoneme Recognition,***" Internal Report 09-174, ESAT-SISTA, K.U.Leuven, Leuven, 2009, submitted for publication.
  - P. Karsmakers, K. Pelckmans, J. A. K. Suykens, "***Multi-class kernel logistic regression: a fixed-size implementation***," In Proc. of the international joint conference in neural networks (IJCNN),Orlando, Florida, U.S.A, pp. 1756-1761, 2007.
- B. Sparse Conjugate Directions Pursuit (SCDP)
  - P. Karsmakers, K. Pelckmans, K. De Brabanter, H. Van hamme, J.A.K. Suykens, "***Sparse Conjugate Directions Pursuit with Application to Fixed-size Kernel Models,***" Internal Report, ESAT-SISTA, K.U.Leuven, Leuven, 2010, submitted for publication.

1. Introduction: Main contributions

- C. Kernel-based phone classification and recognition using segmental features
  - P. Karsmakers, K. Pelckmans, H. Van hamme, J.A.K. Suykens, "***Large-Scale Kernel Logistic Regression for Segment-Based Phoneme Recognition,***" Internal Report 09-174, ESAT-SISTA, K.U.Leuven, Leuven, 2009, submitted for publication.
  - P. Karsmakers, K. Pelckmans, J. A. K. Suykens, H. Van hamme, " ***Fixed-Size Kernel Logistic Regression for Phoneme Classication***," In Proc. of INTERSPEECH, Antwerpen, Belgium, pp. 78-81, 2007.

2. Fixed-size Kernel Logistic Regression

- Focus on "kernelized" variant of Multi-class Logistic Regression (MKLR)
- Potential advantages:
  - Well-founded non-linear discriminative classifier
  - Yields a-posteriori probabilities of class membership based on maximum likelihood argument
  - Well described extension to multi-class case
- Potential disadvantages:
  - Scalability
  - Non-sparse models

Kernel Logistic Regression

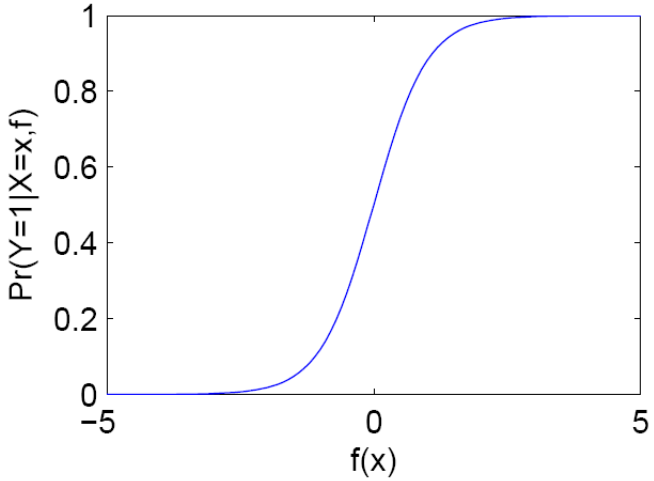
- Estimate a-posteriori probabilities using logistic function

Logistic model

$$\begin{cases} \Pr(\mathbf{Y} = -1 \mid \mathbf{X} = x; f) = \frac{e^{f(x)}}{1+e^{f(x)}} \\ \Pr(\mathbf{Y} = 1 \mid \mathbf{X} = x; f) = \frac{1}{1+e^{f(x)}}, \end{cases}$$

where $f(x) = w^T \varphi(x) + b$. Classification rule is

$$\arg\max_{c \in \{-1,1\}} \Pr(\mathbf{Y} = c | \mathbf{X} = x_*; f).$$



- Learning performed using a convex conditional maximum likelihood objective

Fixed-size Kernel Logistic Regression

- Use all-at-once multi-class logistic model
- Explicit approximation of the nonlinear mapping $\hat{\varphi} : \mathbb{R} \rightarrow \mathbb{R}^M$ using Nyström
- Based on subsample (set of Prototype Vectors (PVs)) of example set, selected using k-center clustering
- Solve primal problem using customized Newton-trust region optimization for multi-class classification

Advantages compared to classical MKLR

- Scalable to large-scale data sets (N > 50,000)
- Sparse models with easily manageable model size M

- [*Karsmakers et al. 2007; Karsmakers et al. 2009*]

Selected experiments: Active PV selection methods

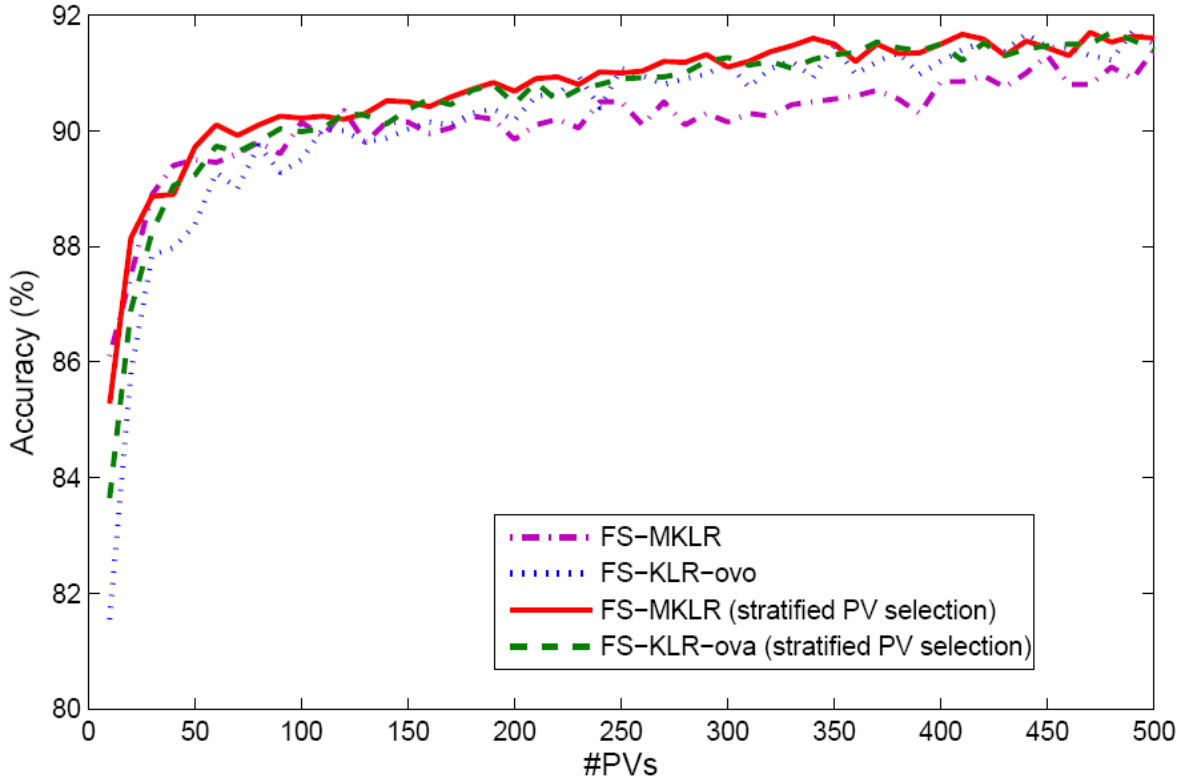
- PV selection important to approximate feature map
- Compare 3 different active PV selection methods on 11 benchmark data sets

	***k*-center**		**renyi**		**random**	
	err	**log likelihood**	**err**	**log likelihood**	**err**	**log likelihood**
banana	**10.63(±0.5)**	−0.2463(±0.0101)	10.72(±0.5)	−0.2501(±0.0128)	10.68(±0.52)	−0.2501(±0.0114)
breast-cancer	27.52(±4.71)	−0.5426(±0.0485)	**27.42(±4.52)**	−0.5436(±0.047)	27.94(±4.53)	−0.5473(±0.0492)
diabetis	**23.23(±1.83)**	−0.4841(±0.0186)	23.3(±1.84)	−0.4856(±0.0184)	23.6(±1.68)	−0.4823(±0.0177)
flare-solar	33.67(±1.65)	−0.5725(±0.0139)	**33.62(±1.61)**	−0.573(±0.0136)	34.02(±1.88)	−0.5731(±0.0166)
heart	**16.7(±3.55)**	−0.4071(±0.0582)	18.33(±4.16)	−0.4295(±0.0648)	17.75(±3.49)	−0.4167(±0.0548)
ringnorm	**1.85(±0.28)**	−0.0575(±0.0156)	2.29(±0.34)	−0.0654(±0.009)	1.91(±0.3)	−0.0569(±0.0089)
splice	**16.61(±1.03)**	−0.3743(±0.0137)	17.78(±1.31)	−0.3926(±0.0183)	17.3(±0.92)	−0.3821(±0.0154)
thyroid	**4.95(±2.23)**	−0.1388(±0.1029)	5.67(±2.72)	−0.17(±0.0954)	5.36(±2.63)	−0.1482(±0.0973)
titanic	22.71(±0.98)	−0.5336(±0.039)	**22.62(±0.86)**	−0.6377(±0.8776)	23.13(±1.43)	−0.5727(±0.1812)
twonorm	**2.79(±0.32)**	−0.0769(±0.0072)	2.94(±0.33)	−0.0808(±0.0126)	2.99(±0.31)	−0.0805(±0.0082)
waveform	**10.13(±0.56)**	−0.224(±0.0105)	10.93(±0.47)	−0.2402(±0.0084)	10.92(±0.53)	−0.2402(±0.0105)

On average k-center clustering gave the best results

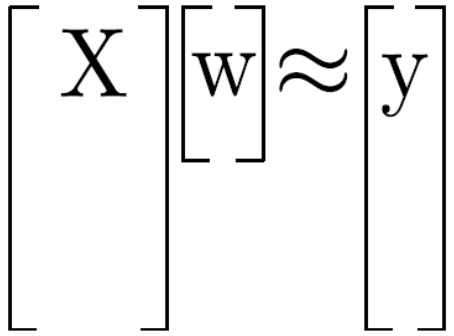
Selected experiments: Sparsity of multi-class schemes

- Compared to combined binary classifiers, all-at-once multi-class approach (with stratified PV selection) is preferred
- `satimage` data set (#classes = 6; #examples = 4,435; #dimensions = 36)



All-at-once FS-MKLR sparsest, and fastest models and similar or better accuracies

3. Sparse Conjugate Directions Pursuit



- Suppose a set of *N* equations, and *D* unknowns
- If *N*>*D* then over-determined linear system
- In general such system has no solution, therefore choose solution according to some optimality criterion, e.g.

Objective

$$w^{0,*} = \arg\min_{w} \|w\|_0 \quad \text{such that} \quad \|Xw - y\|_2^2 \leq \delta,$$

where $\delta > 0$ is an appropriate error tolerance.

Motivations for L_0-norm

- Estimation problems: sparse coefficients

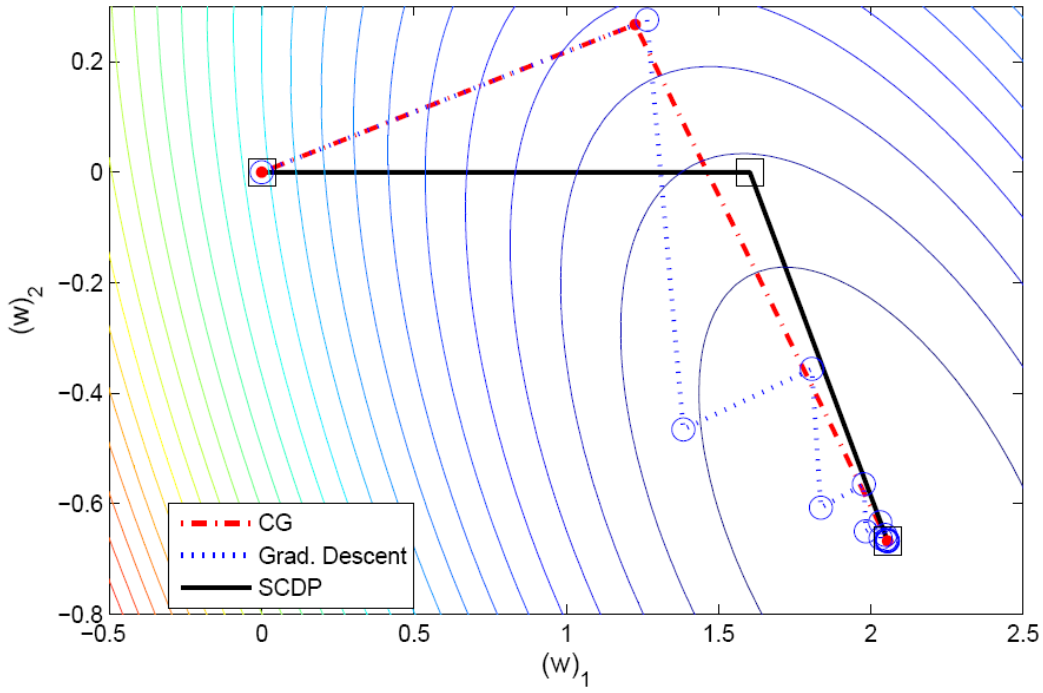
  → feature selection
- Machine learning: sparse predictor rules

  → improved generalization
- Sparse solution leads to computation and memory-efficient model evaluations
- Sparse solution might be exploited when designing scalable algorithms

Greedy heuristic: sparse conjugate directions pursuit

- Heuristically solve previous objective by adapting Conjugate Gradient (CG):

- Iteratively construct a sparse conjugate basis $||w^{(k)}||_0 = k$, starting with $w^{(1)} = 0_D$.
- Globally optimal, local optimization associated to each conjugate basis vector.
- Sequence of conjugate basis vectors is such that a small number of iterations suffice, leading to a sparse solution.



- [*Karsmakers et al., 2010*]

Application: using SCDP for sparse reduced LS-SVMs

Based on SCDP a new kernel-based learning method is derived within the LS-SVM setting:

- Fixed-size LS-SVM approximately solves the primal LS-SVM problem, based on *M* PVs (selected beforehand) Nyström approximation
- Leads to an over-determined linear system of size $N >> M$
- SCDP is then applied to obtain sparser models $\|w\|_0 < M$ (SCDP-FSLSSVM)
- [*Karsmakers et al., 2010*]

Selected experiments

- Decision boundary and final PV positions when using SCDP-FSLSSVM on `ripley` benchmark (RBF kernel is used)



Selected experiments

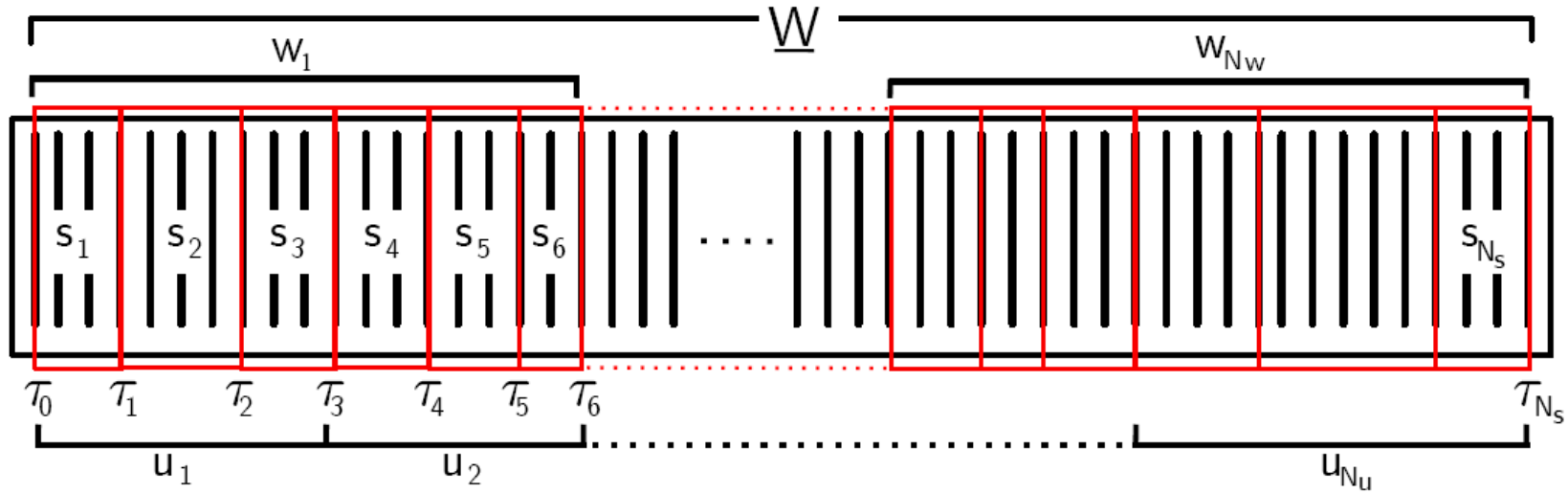
- Binary classification on benchmarks `adult` and `gamma telescope`

Data set	SCDP-FSLSSVM		SVM	
	err (%)	# PV	err (%)	# SV
adult	$14.55(\pm 0.34)$	$166(\pm 41.6)$	$14.53(\pm 0.14)$	$10,494(\pm 212.4.0)$
gamma telescope	$13.90(\pm 0.27)$	$778(\pm 58.9)$	$13.3(\pm 0.12)$	$4,211(\pm 112.6)$

SCDP models much sparser than those of SVM

4. Segment-based Phone Recognition

- Recall our feature set $F(x) : \mathbb{R}^M \rightarrow \mathbb{R}^2, F(x) = (mse(x), duration(x))^T$
- In practice more and better features needed
- State-of-the-art uses features computed for very small speech parts (frames)
- Our setup operates on larger parts, i.e. phone segments
- Focus on recognizing phone sequences, forms a basis for a good word recognition



Motivation: Kernel Models in Segment-based Approach

- State-of-the-art in ASR based on Hidden Markov Models (HMM)
- Impressive HMMs but modeling limitations, e.g. duration- and trajectory modeling

Potential advantages using segment-based approach

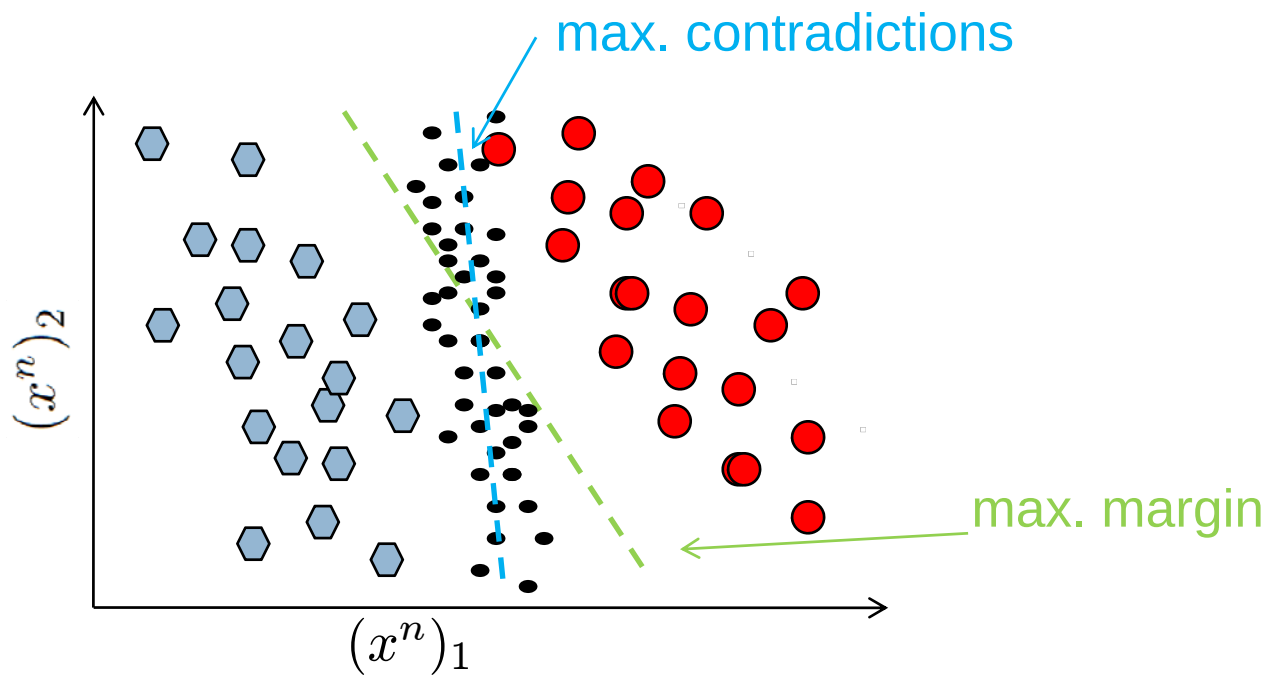
- Segment-based setup might overcome HMM limitation
- Segments span larger context windows, possibly more features (more dimensions) are needed
- Kernel-based methods generalize well even in high dimensional spaces

Inference with a universum

- During recognition segments with no lexical meaning are presented to the models
- These garbage segments do not belong to a certain class → universum data
- Incorporate universum information in learning process by altering learning objective → maximize model output contradictions for universum data
- Final model size remains unchanged
- Applied to FS-MKLR in [*Karsmakers et al., 2009*]

Inference with a universum

- Learning objective: Maximize margin versus maximize model output contradictions for universum data



Selected experiments

- On `timit` benchmark speech corpus.
- Segment-based classification and recognition. Segment (unit)= phone

Data set characteristics

- 142,910 examples to learn model
- 51,681 examples to validate model
- 181 features (dimensions) per example

- [*Karsmakers et al., 2007;Karsmakers et al., 2009*]

Phone Classification

- Segment boundaries are known
- Compare to 78.6% for state-of-the-art ASR system (HMM)

alg	acc(%)	t_{tr}	t_{es}
$\text{SVMprob}_{\text{pc}}$ (89,563 SVs)	83.2	10h	2h
SVM_{vs} (89,563 SVs)	82.9	3h	1h
$\text{SCDP-FSLSSVM}_{\text{pc}}$ (66,001 PVs)	82.7	9.6h	44min
$\text{FS-KLR}_{\text{pc}}$ (80,000 PVs)	82.5	23h	59min
$\text{FS-KLR}_{\text{pkpd}}$ (80,000 PVs)	82.5	23h	50min
FS-MKLR (8,000 PVs)	81.9	86h	0.5min
$\text{FS-KLR}_{\text{pc}}$ (8,000 PVs)	80.5	50min	7min
$\text{FS-KLR}_{\text{pkpd}}$ (8,000 PVs)	80.5	50min	6min
LDA	66.5	4.7min	0.03min
KNN (k=1)	67.8	-	11.5h

Kernel based alternatives outperform a state-of-the-art HMM classifier

Phone Recognition

- Segment boundaries unknown

LM	FS-MKLR		SVM		FS-MKLR$_{\mathcal{U}}$	
	(I,D,S) (%)	PER (%)	(I,D,S) (%)	PER (%)	(I,D,S) (%)	PER (%)
no LM	(4.69, 10.40, 16.02)	31.10	(4.13, 11.92, 15.42)	31.47	(4.68, 8.03, 16.10)	28.81
2-gram	(4.01, 9.14, 16.17)	29.31	(4.16, 9.52, 15.52)	29.20	(4.84, 7.04, 16.15)	28.03
3-gram	(4.09, 8.50, 15.74)	28.32	(4.82, 8.18, 15.14)	28.13	(4.36, 7.27, 15.70)	27.32
4-gram	(4.59, 7.80, 15.73)	28.13	(4.44, 8.60, 14.98)	28.02	(4.52, 7.05, 15.61)	27.17

LM	HMM	
	(I,D,S) (%)	PER (%)
no LM	(3.04, 9.23, 17.99)	30.26
1-gram	(2.97, 8.98, 17.01)	28.97
2-gram	(2.70, 7.65, 16.36)	26.70
3-gram	(3.17, 6.50, 15.96)	25.62
4-gram	(3.41, 6.16, 15.71)	25.28

Although much smaller model size FS-MKLR produced similar results compared to SVM

Phone Recognition

- Segment boundaries unknown

LM	FS-MKLR		SVM		FS-MKLR$_{\mathcal{U}}$	
	(I,D,S) (%)	PER (%)	(I,D,S) (%)	PER (%)	(I,D,S) (%)	PER (%)
no LM	(4.69, 10.40, 16.02)	31.10	(4.13, 11.92, 15.42)	31.47	(4.68, 8.03, 16.10)	28.81
2-gram	(4.01, 9.14, 16.17)	29.31	(4.16, 9.52, 15.52)	29.20	(4.84, 7.04, 16.15)	28.03
3-gram	(4.09, 8.50, 15.74)	28.32	(4.82, 8.18, 15.14)	28.13	(4.36, 7.27, 15.70)	27.32
4-gram	(4.59, 7.80, 15.73)	28.13	(4.44, 8.60, 14.98)	28.02	(4.52, 7.05, 15.61)	27.17

LM	HMM	
	(I,D,S) (%)	PER (%)
no LM	(3.04, 9.23, 17.99)	30.26
1-gram	(2.97, 8.98, 17.01)	28.97
2-gram	(2.70, 7.65, 16.36)	26.70
3-gram	(3.17, 6.50, 15.96)	25.62
4-gram	(3.41, 6.16, 15.71)	25.28

Universum objective improved the accuracy without increasing the number of parameters of the phone model

Phone Recognition

- Segment boundaries unknown

LM	FS-MKLR		SVM		FS-MKLR$_{\mathcal{U}}$	
	(I,D,S) (%)	PER (%)	(I,D,S) (%)	PER (%)	(I,D,S) (%)	PER (%)
no LM	(4.69, 10.40, 16.02)	31.10	(4.13, 11.92, 15.42)	31.47	(4.68, 8.03, 16.10)	28.81
2-gram	(4.01, 9.14, 16.17)	29.31	(4.16, 9.52, 15.52)	29.20	(4.84, 7.04, 16.15)	28.03
3-gram	(4.09, 8.50, 15.74)	28.32	(4.82, 8.18, 15.14)	28.13	(4.36, 7.27, 15.70)	27.32
4-gram	(4.59, 7.80, 15.73)	28.13	(4.44, 8.60, 14.98)	28.02	(4.52, 7.05, 15.61)	27.17

LM	HMM	
	(I,D,S) (%)	PER (%)
no LM	(3.04, 9.23, 17.99)	30.26
1-gram	(2.97, 8.98, 17.01)	28.97
2-gram	(2.70, 7.65, 16.36)	26.70
3-gram	(3.17, 6.50, 15.96)	25.62
4-gram	(3.41, 6.16, 15.71)	25.28

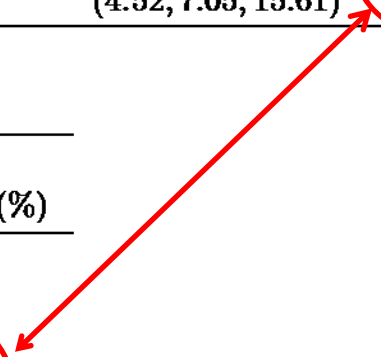
Although improvement possible, without LM segment-based approach has state-of-the-art accuracy

Phone Recognition

- Segment boundaries unknown

LM	FS-MKLR		SVM		FS-MKLR$_{\mathcal{U}}$	
	(I,D,S) (%)	PER (%)	(I,D,S) (%)	PER (%)	(I,D,S) (%)	PER (%)
no LM	(4.69, 10.40, 16.02)	31.10	(4.13, 11.92, 15.42)	31.47	(4.68, 8.03, 16.10)	28.81
2-gram	(4.01, 9.14, 16.17)	29.31	(4.16, 9.52, 15.52)	29.20	(4.84, 7.04, 16.15)	28.03
3-gram	(4.09, 8.50, 15.74)	28.32	(4.82, 8.18, 15.14)	28.13	(4.36, 7.27, 15.70)	27.32
4-gram	(4.59, 7.80, 15.73)	28.13	(4.44, 8.60, 14.98)	28.02	(4.52, 7.05, 15.61)	27.17

LM	HMM	
	(I,D,S) (%)	PER (%)
no LM	(3.04, 9.23, 17.99)	30.26
1-gram	(2.97, 8.98, 17.01)	28.97
2-gram	(2.70, 7.65, 16.36)	26.70
3-gram	(3.17, 6.50, 15.96)	25.62
4-gram	(3.41, 6.16, 15.71)	25.28



However, less gain when using LMs as is the case for the HMM recognizer

5. General Conclusions

- Previously specified requirements were tackled as follows:
  - Scalability: Two practical, and scalable kernel-based algorithms: FS-MKLR and SCDP-FSLSSVM. Trade-off between model accuracy and training (and model) complexity, directly controlled by the user
  - Multi-class classification: All-at-once FS-MKLR is preferred compared to binary coupled variants in terms of classification accuracy and model sparsity
  - Sparseness:
    - Tuned SVM model not really sparse
    - One-versus-one coding scheme additionally increases model size
    - Proposed methods produce significantly sparser models while having comparable accuracies to state-of-the-art

5. General Conclusions

- Probabilistic interpretation: All considered methods yield, either direct or indirect, probabilistic outcomes, both empirically gave adequate results
- Variable-length sequence: A simple and fast mapping to fixed-length vectors was used

- We successfully integrated our new kernel models in a segment-based speech recognition system and compared to a state-of-the-art ASR system

6. Future Work

- Segmentation model: use a more sophisticated boundary detection model, possibly using other types of features

- Design of a customized kernel: use other positive semi-definite kernels, e.g. sequence kernels → score pairs of variable-length segments directly

- SCDP-FSMKLR: apply SCDP on FS-MKLR method

Acknowledgements

Thanks for listening!