http://doi.org/10.1016/j.neucom.2009.11.045

Feature Modeling Using Polynomial Classifiers and Stepwise Regression

T. Shanableh[1] and K. Assaleh[2]
Department of Computer Science and Engineering[1]
Department of Electrical Engineering [2]
American University of Sharjah[1,2]
tshanableh@aus.edu

Abstract

In polynomial networks, feature vectors are mapped to a higher dimensional space through a polynomial function. The expanded vectors are then passed to a single layer network to compute the model parameters. However, as the dimensionality of the feature vectors grows with polynomial expansion, polynomial training and classification become impractical due to the prohibitive number of expanded variables. This problem is more prominent in vision-based systems where high dimensionality feature vectors are extracted from digital images and/or video. In this paper we propose to reduce the dimensionality of the expanded vector through the use of stepwise regression. We compare our work to the reduced-model multinomial networks where the dimensionality of the expanded feature vectors grows linearly whilst preserving the classification ability. We also compare the proposed work to standard polynomial classifiers and to established techniques of polynomial classifiers with dimensionality reduction. Two application scenarios are used to test the proposed solution, namely; image-based hand recognition and video-based recognition of isolated sign language gestures. Various datasets from the UCI machine learning repository are also used for testing. Experimental results illustrate the effectiveness of the proposed dimensionality reduction technique in comparison to published methods.

Keywords:

Polynomial classifier, pattern classification, vision-based intelligent systems, image/video processing,.

1. Introduction

Linear discriminant functions are considered amongst the simplest supervised classification methods. In such methods, a sequence of feature vectors is linearly mapped into a sequence of class labels. Multi-class classification problems can be reduced to multiple two-class classification problems. Linear discriminant functions work very well with linearly separable data. However, they are less accurate when the data is not linearly separable. As a solution to this problem, many nonlinear classification methods were introduced in the past few decades including neural and statistical classifiers. Amongst the neural classifiers falls the polynomial classifier [1],[2] and [3] which can be thought of as a network which accepts feature vectors, maps them to a higher dimensional space through a polynomial function and passes the expanded vectors through a single layer network. The weights of this network are obtained through the minimization of the L^2-*norm* of the error between the output of the network and the desired outputs for the training data.

However, as the dimensionality of the feature vectors grows with polynomial expansion, polynomial training and classification become impractical due to the prohibitive number of expanded variables. One approach to solve this problem is through dimensionality reduction of the expanded feature set. For instance [4] proposed a speaker verification system based on polynomial networks with dimensionality reduction. A random dimensionality reduction technique was proposed based on linear transforms such as Principal Component Analysis (PCA) [5, 6] and Fast Fourier Transformation (FFT).

Another approach is based on piecewise regression. In [7] it was proposed to use polynomial models to fit each subset or piece of the predictors. Then contiguous pieces of the predictor space are generated using a recursive partitioning algorithm. Lastly, piecewise polynomial model estimates are combined using weighted averaging.

More recently [8] proposed a simple yet promising reduced polynomial model whose number of parameters increases linearly whilst preserving decent classification capability. Multinomials that are a special case of multivariate polynomials are used for expansion and model estimation.

In this paper, we propose to reduce the dimensionality of expanded feature sets through the use of the stepwise regression procedure. In such a procedure the predictor variables or the elements of the expanded feature vectors are screened to obtain the best subset of variables. We apply the proposed solution to two application scansions; image-based hand recognition and video-based recognition of isolated sign language gestures. We compare the proposed solution against the reviewed reduced polynomial model and standard polynomials networks.

This paper is organized as follows. Section 2 reviews standard polynomial networks including expansion, training and model estimation. Section 3 briefly reviews the reduced polynomial model. Section 4 introduces the proposed solution that integrates polynomial expansion with stepwise regression. Section 5 introduces the application scenarios used to verify the proposed solution. The experimental results are presented in Section 6 followed by the concluding remarks.

2. Polynomial networks

A Polynomial network is a supervised classifier that is capable of learning complex patterns that could be linearly inseparable. Polynomial networks have been successfully used in various applications of pattern recognition including speech and speaker recognition [1],[2] and[3] and biomedical signal separation [9].

A polynomial network is a parameterized nonlinear map which nonlinearly expands a sequence of input vectors to a higher dimension and maps them to a desired output sequence.

Training a P^{th} order polynomial network consists of two main parts. Part one is expanding the training feature vectors via polynomial expansion. The purpose of this expansion is to improve the separation of the different classes in the expanded vector space. Ideally, we aim to have this expansion make all the classes linearly separable. Part two is linearly mapping the polynomial-expanded vectors to an ideal output sequence by minimizing an objective criterion. The mapping parameters represent the weights of the polynomial network. These weights are often referred to as the class models.

2.1 Polynomial Expansion

Polynomial expansion of an M-dimensional feature vector $\mathbf{x} = [x_1 \quad x_2 \ \ldots \ x_M]$ is achieved by combining the vector elements with multipliers to form a set of basis functions, **p(x).** The elements of **p(x)** are the monomials of the form $\prod_{j=1}^{M} x_j^{k_j}$, where k_j is a positive integer, and $0 \le \sum_{j=1}^{M} k_j \le P$.

Therefore, the P^{th} order polynomial expansion of an M-dimensional vector **x** generates an $O_{M,P}$-dimensional vector **p**(**x**). $O_{M,P}$ is a function of both M and P and can be expressed as

$$O_{M,P} = 1 + PM + \sum_{l=2}^{P} C(M,l) \qquad (1)$$

where $C(M,l)=\binom{M}{l}$ is the number of distinct subsets of l elements that can be made out of a set of M elements. Therefore, for class i the sequence of feature vectors $\mathbf{X}_i=[\mathbf{x}_{i,1}\ \mathbf{x}_{i,2}\ \cdots,\mathbf{x}_{i,N_i}]^T$ is expanded into:

$$\mathbf{V}_i=[\mathbf{p}(\mathbf{x}_{i,1})\quad \mathbf{p}(\mathbf{x}_{i,2})\quad \cdots\quad \mathbf{p}(\mathbf{x}_{i,N_i})]^T \tag{2}$$

Notice that while $\mathbf{X}_i$ is a $N_i\times M$ matrix, $\mathbf{V}_i$ is a $N_i\times O_{M,\mathrm{p}}$ matrix.

Expanding all the training feature vectors results in a global matrix for all K classes obtained by concatenating all the individual $\mathbf{V}_i$ matrices such that $\mathbf{V}=[\mathbf{V}_1\quad \mathbf{V}_2\quad \cdots\quad \mathbf{V}_K]^T$.

2.2 Solving for the network weights

For each class i, the training problem reduces to finding an optimum weight vector. This weight vector is obtained by minimizing the distance between the ideal output vector $\mathbf{y}_i$ and a linear combination of the polynomial expansion of the training feature vectors $\mathbf{V}\,\mathbf{w}_i$ such that

$$\mathbf{w}_i^{opt}=\arg\min_{\mathbf{w}_i}\|\mathbf{V}\,\mathbf{w}_i-\mathbf{y}_i\|_p \tag{3}$$

The ideal output for the i^{th} class, $\mathbf{y}_i$, is a column vector comprised of ones and zeros such as $\mathbf{y}_i=[\,\mathbf{0}_{N_1},\mathbf{0}_{N_2},\ldots,\mathbf{0}_{N_{i-1}},\mathbf{1}_{N_i},\mathbf{0}_{N_{i+1}},\ldots,\mathbf{0}_{N_k}\,]^{\mathrm{T}}$. Equation 3 indicates that weight vector is obtained by minimizing the *L^p-norm* of the error vector $\mathbf{e}_i=\mathbf{V}\mathbf{w}_i-\mathbf{y}_i$. For the special case of p=2, we arrive at the well-known *L^2-regression* problem. That is, finding $\mathbf{w}_i^{opt}$ that attains the minimum of the *L^2-norm* of the error sequence $\mathbf{e}_i$. Or equivalently, minimizing the square of the *L^2-norm* such as

$$\mathbf{w}_i^{opt}=\arg\min_{\mathbf{w}_i}\|\mathbf{V}\mathbf{w}_i-\mathbf{y}_i\|_2^2 \tag{4}$$

Fortunately, for this particular formulation with the *L^2-norm* there is an explicit formula for the solution $\mathbf{w}_i^{opt}$. This solution can be obtained by applying the normal equations method [10] such as

$$\mathbf{V}^T\mathbf{V}\mathbf{w}_i^{opt}=\mathbf{V}^T\mathbf{y}_i \tag{5}$$

By incorporating Equation 3, Equation 5 can be rearranged as

$$\sum_{j=1}^{K}\mathbf{V}_j^T\mathbf{V}_j\mathbf{w}_i^{opt}=\mathbf{V}_i^T\mathbf{1}_i \tag{6}$$

If we define $\mathbf{R}_j = \mathbf{V}_j^T \mathbf{V}_j$, $\mathbf{R} = \sum_{j=1}^{K} \mathbf{R}_j$, and $\mathbf{v}_i = \mathbf{V}_i^T \mathbf{1}_i$ then equation (6) yields an explicit solution for $\mathbf{w}_i^{opt}$ expressed as

$$\mathbf{w}_i^{opt} = \mathbf{R}^{-1}\mathbf{v}_i \qquad (7)$$

The set $\{\mathbf{w}_i^{opt}\}$ represents the weights of the K polynomial networks which we refer to as the class models.

In [1], Campbell and Assaleh discuss the computational aspects of solving for $\mathbf{w}_i^{opt}$ and they present a fast method for training polynomial networks by exploiting the redundancy of the $\mathbf{R}_j$ matrices. They also discuss in details the computational and storage advantages of their training method.

2.3 Identification

In the identification stage we are given a sequence of N_c feature vectors $\mathbf{X}_c$ and we are required to determine its class c as one of the enrolled classes in the set $\{1,2,\cdots,K\}$. This is done by two steps: first, expand $\mathbf{X}_c$ into its polynomial basis terms $\mathbf{V}_c = [\mathbf{p}(\mathbf{x}_{c,1}) \quad \mathbf{p}(\mathbf{x}_{c,2}) \quad \cdots \quad \mathbf{p}(\mathbf{x}_{c,N_c})]^T$, and second, evaluate the output sequences against all K models $\{\mathbf{w}_i^{opt}\}$ to obtain a set of score sequences $\{\mathbf{s}_i\}$ such as

$$\mathbf{s}_i = \mathbf{V}_c \mathbf{w}_i^{opt}. \qquad (8)$$

The elements of the score sequence $\mathbf{s}_i$ represent the individual scores of each feature vector in the vector sequence $\mathbf{X}_c$. The class of the sequence $\mathbf{X}_c$ is determined by maximizing $\{g(\mathbf{s}_i)\}$ such as

$$c = \arg\max_i (g(\mathbf{s}_i)) \qquad (9)$$

where g is a function that outputs a statistic of the sequence $\mathbf{s}_i$ such as the mean or the median. In our case we chose g to compute the mean of $\mathbf{s}_i$ such as

$$g(\mathbf{s}_i) = \frac{1}{N_c} \sum_{j=1}^{N_c} s_{i,j} \qquad (10)$$

3. Review of reduced polynomial model

In [8] the use of multinomial for expansion and model estimation was proposed. The weight parameters are estimated from the following multinomial model:

$$f_{RM}(\boldsymbol{\alpha},\boldsymbol{x})=\alpha_0+\sum_{k=1}^{r}\sum_{j=1}^{l}\alpha_{kj}x_j^k+\sum_{j=1}^{r}\alpha_{rl+j}(x_1+x_2+\cdots+x_l)^j+\sum_{j=2}^{r}(\boldsymbol{\alpha}_j^T.\boldsymbol{x})(x_1+x_2+\cdots+x_l)^{j-1},\ l,r\geq 2 \quad (11)$$

Where r is the order of the polynomial, $\boldsymbol{\alpha}$ is the polynomial weights to be estimated, $\boldsymbol{x}$ is the feature vector containing l inputs. The total number of terms in $f_{RM}(\boldsymbol{\alpha},\boldsymbol{x})$ is equal to 1+r+l(2r-1). Just like the case of standard polynomial networks, the polynomial weights can be estimated using least-squares error minimization.

4. Proposed polynomial networks with stepwise regression:

Stepwise regression is a wildly used regressor variable selection procedure. To illustrate the procedure (as described in [11]), assume that we have K candidate variables $x_1,x_2,\ldots,x_k$ and a single response variable y. In classification the candidate variables correspond to the polynomial-expanded elements of the feature vectors and the response variable corresponds to the class label. Note that with the intercept term β_0 we end up with $K+1$ variables.

In the procedure the polynomial weights (or the regression model) are iteratively found by adding or removing variables at each step. The procedure starts by building a one variable regression model using the variable that has the highest correlation with the response variable y. This variable will also generate the largest partial F-statistic. In the second step, the remaining $K-1$ variables are examined. The variable that generates the maximum partial F-statistic is added to the model provided that the partial F-statistic is larger than the value of the F-random variable for adding a variable to the model, such an F-random variable is referred to as f_{in}. Formally the partial F-statistic for the second variable is computed by: $f_2=\frac{SS_R(\beta_2|\beta_1,\beta_0)}{MS_E(x_2,x_1)}$. Where $MSE(x_2,x_1)$ denotes the mean square error for the model containing both x_1 and x_2. $SS_R(\beta_2|\beta_1,\beta_0)$ is the regression sum of squares due to β_2 given that β_1,β_0 are already in the model.

In general the partial F-statistic for variable j is computed by:

$$f_j=\frac{SS_R(\beta_j|\beta_0,\beta_1,\ldots,\beta_{j-1},\beta_{j+1},\ldots,\beta_k)}{MS_E} \quad (12)$$

If variable x_2 is added to the model then the procedure determines whether the variable x_1 should be removed. This is determined by computing the F-statistic $f_1 = \frac{SS_R(\beta_1|\beta_2\beta_0)}{MS_E(x_2,x_2)}$. If f_1 is less than the value of the F-random variable for removing variables from the model, such an F-random variable is referred to as f_{out}.

The procedure examines the remaining variables and stops when no other variable can be added or removed from the model. More information on stepwise regression can be found in classical statistics and probability texts such as [11].

It is also worth mentioning that one cannot arrive to the conclusion that all of the regressors that are important for predicting the response variable have been retained in the stepwise procedure. This is so because such a procedure retains regressors based on the use of sample estimates of the true model weights. It is understood that there is a probability of making errors in retaining regressors.

The integration of the stepwise regression into the polynomial classifier is illustrated in Figure 1. Note that the elements of the expanded feature vectors are examined using the aforementioned procedure during the training stage of the classifier. The indices of the retained elements of the expanded feature vectors are stored and passed on to the testing or validation stage. Polynomial expansion is applied to a test feature vector. Only, the feature vector elements corresponding to the indices found from the training stage are retained. Thus the stepwise regression procedure is applied during the training stage only. The model parameters are based on the reduced training feature vectors. The same parameters are used for classification during the testing stage.

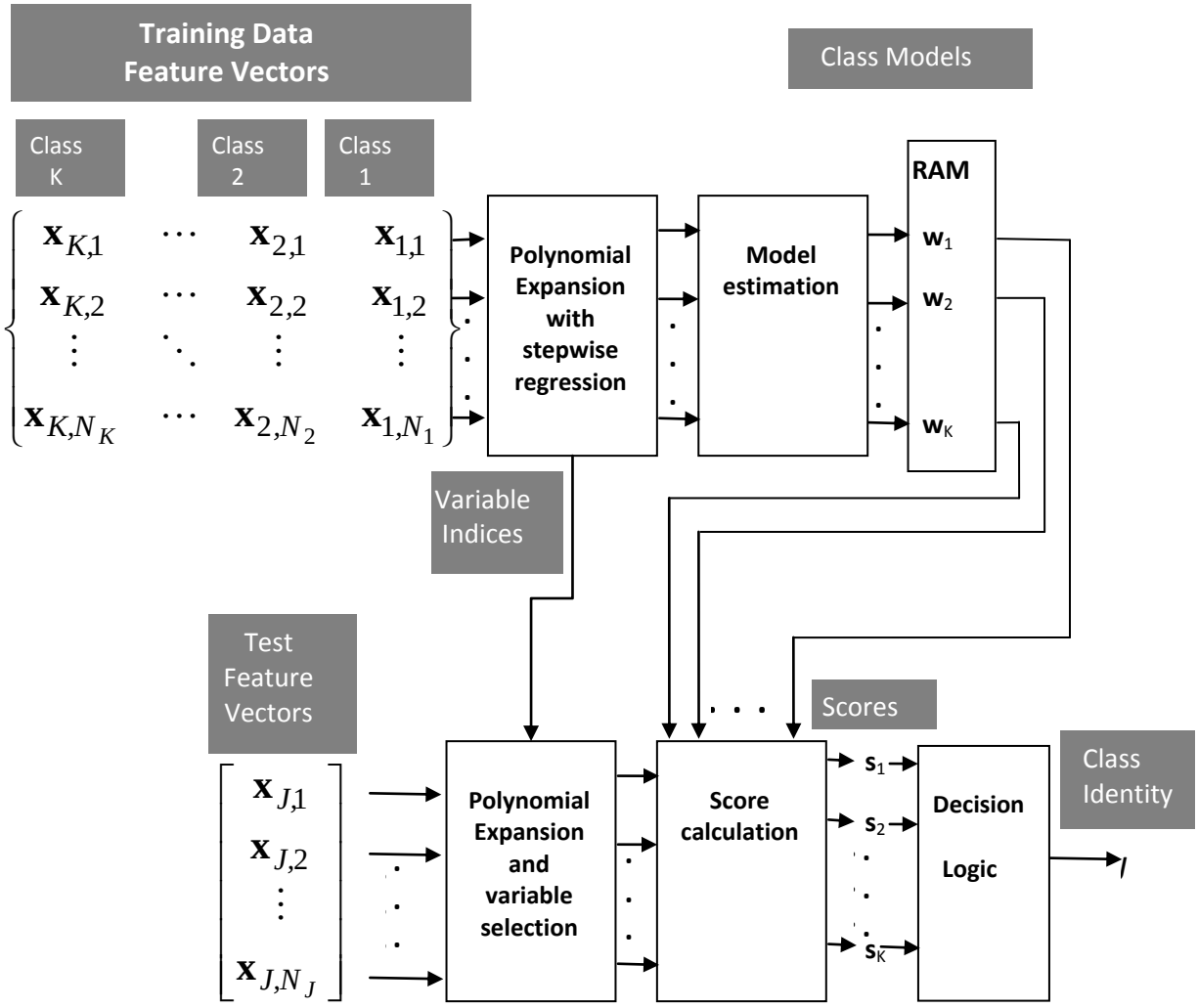


Figure 1. Illustration of polynomial classification with stepwise regression.

Note that the time complexity of the stepwise regression using the Big-O notation is $O(N^2)$. But again, such a procedure is only needed during the training stage.

5. Application scenarios

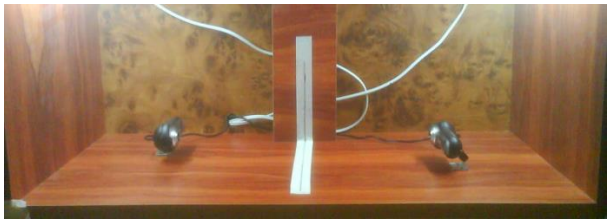
We use two computer vision application scenarios to validate the proposed solution, namely; image-based hand recognition and recognition of video-based isolated sign language gesture. Additionally, to further test the proposed solution, we use 10 classification datasets taken from the UCI Machine Learning Repository [12].

5.1 Hand recognition

This application scenario is based on an image-based hand recognition system. The application scenario proposes the use of both palm and back of hand images for hand recognition.

a. Dataset description:

A wooden box that contains 2 digital cameras is used to collect images. The box is shown in part 'a' of Figure 2. A total of 53 users participated in the data collection phase. Images of both sides of the hand of each subject were captured. Example images are shown in parts 'b' and 'c' of Figure 2. Subjects are asked to reenter their hands into the box after each capture of image pairs. Data was collected over two sessions. In each session 10 image pairs are collected per subject. The total number of images per session is therefore *53x10x2*. The data collected in the first session is used for training and the data collected in the second session is used for validation.



(a) Data collection box fixed with two cameras to capture hand images

(b) Example image of the hand's palm

(c) Example image of the back of the hand.

Figure 2. Data collection for the hand recognition

b. Feature extraction:

The feature extraction scheme we use is rather simple. Each image is converted to gray scale and transformed into the frequency domain using the 2-D Discrete Cosine Transformation (DCT) given by:

$$F(u,v) = \frac{2}{\sqrt{MN}} C(u)C(v) \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} f(i,j) \cos\left(\frac{\pi u}{2M} \cdot (2i+1)\right) \cos\left(\frac{\pi v}{2N} \cdot (2j+1)\right) \quad (13)$$

Where *NxM* are the dimensions of the input image '*f*' and *F(u,v)* is the DCT coefficient at row *u* and column *v* of the DCT matrix. *C(u)* is a normalization factor equal to $1/\sqrt{2}$ for u=0 and 1 otherwise. More information about DCT transforms are found in [13].

An attractive property of this transformation is its energy compaction. Low frequencies are concentrated in the top left corner of the transformed image. Thus the input image can be coarsely represented by discarding high frequencies. In this work the DCT coefficients are zigzag scanned from the top left corner into an *n* dimensional vector [14]. The dimensionality is empirically determined as illustrated in the experimental results section. The block diagram of the proposed spatial feature extraction is shown in Figure 3:

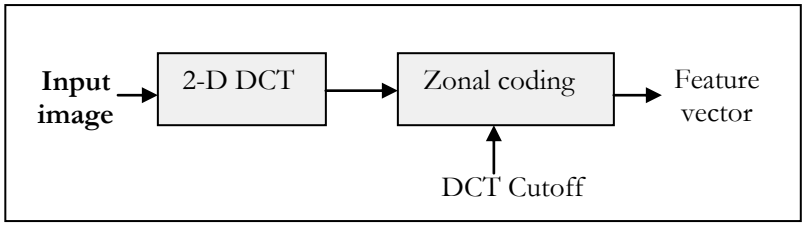


Figure 3. block diagram of the feature extraction technique from hand images.

Note that other transform-based feature extraction techniques for hand recognition are reported in the literature. For instance [15, 16] proposed to extract features from wavelet coefficients of the palm. Likewise [17] proposed the use of Fourier transformation in the feature extraction stage.

5.2 Isolated Sign language recognition

This section describes the sign language dataset used in this application scenario followed by feature extraction.

a. Dataset description

Arabic Sign Language does not yet have a standard database that can be purchased or publicly accessed. Therefore, we decided to use our own ArSL database which was collected in [18, 19]. The dataset contains 23 Arabic gestured words/phrases from 3 different signers. The list of words is shown in Table 1.

#	Arabic word	Meaning in English	#	Arabic word	Meaning in English
1	صديق	Friend	13	يأكل	To Eat
2	جار	Neighbor	14	ينام	To sleep
3	ضيف	Guest	15	يشرب	To Drink
4	هدية	Gift	16	يستيقظ	To wake up
5	عدو	Enemy	17	يسمع	To listen
6	السلام عليكم	Peace upon you	18	يسكت	To stop talking
7	اهلا وسهلا	Welcome	19	يشم	To smell
8	شكرا	Thank you	20	يساعد	To help
9	تفضل	Come in	21	امس	Yesterday
10	عيب	Shame	22	يدهب	To go
11	بيت	House	23	يأتي	To come
12	انا	I/me			

Table 1: Arabic sign language gestures and their English meanings.

Each of the three signers was asked to repeat each gesture 50 times over three different sessions resulting in a total of 150 repetitions of the 23 gestures which corresponds to 3450 video segments. The signer was videotaped without imposing any restriction on clothing or image background.

b. Sign language feature extraction

We adopt one of the feature extraction techniques proposed by the authors in [18]. For completeness this section provides a summary of the adopted technique.

It was shown that the motion information in a video-based gesture is extracted from the temporal domain of the input image sequence through successive image differencing. Let $I_{g,i}^{(j)}$ denote image index j of the i^{th} repetition of a gesture at index g. The image formed from the Accumulated Differences (ADs) can be computed by:

$$AD_{g,j} = \sum_{j=1}^{n-1} \partial_j \left(\left| I_{g,j}^{(j)} - I_{g,i}^{(j-1)} \right| \right) \qquad (14)$$

Where n is the total number of images in the i^{th} repetition of a gesture at index g. ∂_j is a binary threshold function of the j^{th} frame.

Radon transformation is applied to the resultant ADs image. As such, the pixel intensities of the ADs image are projected at a given angle θ using the following equation:

$$R_\theta(x) = \int_{-\infty}^{+\infty} f(x'\cos\theta - y'\sin\theta, x'\sin\theta + y'\cos\theta)dy' \qquad (15)$$

Where f is the input image and the line integral is parallel to the y' axis where x' and y' are given by:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad (16)$$

The projected ADs image is then coarsely represented by transforming it into the frequency domain using a 1-D DCT followed by an ideal low pass filter. It was shown in [18] that the number of DCT coefficients to retain in the ideal low pass filter can be determined empirically. Given that the ADs image is projected onto the x-axis, a DCT cutoff of 100 was shown to be adequate. The feature vectors entail the retained 100 DCT coefficients.

6. Experimental results:

In this section, the proposed use of stepwise regression with polynomial networks is compared against the reviewed reduced polynomial model. The proposed work is also compared against standard polynomial classifiers with polynomial expansion but without dimensionality reduction.

In Figure 4, we experiment with the application scenarios of hand recognition and sign language recognition. The figure shows the classification rate as a function of the dimensionality of the feature vectors prior to the expansion. In both application scenarios this corresponds to the DCT cutoff in the feature extraction stage as described in Sections 5.1 and 5.2 above.

DCT cutoff	Proposed	Reduced model	Polynomial
40	0.918	0.86	0.834
50	0.92	0.84	0.868
Average	0.92	0.85	0.85

(a) Palm recognition

DCT cutoff	Proposed	Reduced model	Polynomial
40	0.78	0.73	0.55
50	0.78	0.72	0.63
Average	0.78	0.72	0.59

(b) Back of hand recognition

DCT cutoff	Proposed	Reduced model	Polynomial
40	0.962	0.955	0.911
50	0.973	0.948	0.823
Average	0.968	0.956	0.867

(c) Sign language classification

Figure 4. DCT cutoff versus classification rate for different application scenarios.

In Part 'a' of the figure, it is shown that the classification rate of the proposed solution is higher than the both the reduced polynomial model and the standard polynomial classifier.

In part 'b' of the figure, the classification rates of the reduced polynomial model are on average higher than those of the standard polynomial classifier. However both are inferior to the classification rates of the proposed solution. On average, the classification rate of the proposed solution is around 6% higher than the reduced polynomial model and 19% higher than the standard approach. Also note that due to the 'curse of dimensionality' the standard polynomial classifier results in expanded feature vectors with high dimensionality affecting the numerical stability of the inversion of a potentially ill-conditioned matrix R in Equation 7 above, thus the less accurate classification results. This problem is frequent and is one of the major drawbacks of standard polynomial networks.

Part 'c' of the figure presents the classification results for the sign language recognition application scenario. Again on average, the proposed solution generates the highest classification results. However, the gain in this application scenario is not as pronounced as in the previous two figures.

In general, the gain in classification rate using the proposed solution is expected. This is because the stepwise regression procedure adds regressor variables that will affect the response variable most rather than blindly using all the regressors of the expanded feature vector.

It is also shown in the figure that the classification rate using the polynomial classifier decreases at a cutoff of 50 DCT coefficients. Again this is due to the aforementioned numerical instability caused by the fact that the matrix R in Equation 7 is ill conditioned which leads to an unreliable set of model weights hence a lower classification rate as reported in Figure 4.c.

In Figure 5 we present the dimensionality of the expanded feature vectors through the use of the proposed and the reviewed solutions. The three parts of the figure correspond to the experiments in Figure 4 with a second order expansion. It is shown that the dimensionality of the expanded feature vectors using the reduced polynomial model and the standard polynomial expansion are of fixed

length regardless of the content of the training datasets. On the other hand, in the proposed solution only regressors that affect the response variable most are selected, hence the length of the expanded feature vector depends on the content of the training dataset.

DCT cutoff	Standard Polynomials	Multinomials	Proposed
40	861	123	123
50	1326	153	129

(a) Palm expansion

DCT cutoff	Standard Polynomials	Multinomials	Proposed
40	861	123	121
50	1326	153	111

(b) Back of hand expansion

DCT cutoff	Standard Polynomials	Multinomials	Proposed
40	861	123	199
50	1326	153	284

(c) Sign language expansion

Figure 5. Dimensionality of expanded feature vectors using the proposed solution versus existing work.

Clearly, the standard polynomial expansions results in an exponential growth of the feature vector as a function of the DCT cutoff. At a moderate DCT cutoff of 50 the length of the expanded vector grows to 1326. On the other hand, the proposed solution reasonably expands the dimensionality of the feature vectors. In the first application scenario, this dimensionality is lower than that proposed by the reduced polynomial model. However the reverse situation is evident in the sign language recognition scenario. Nonetheless, the dimensionality is still much smaller than that proposed by the standard polynomial expansion.

Regarding the computational complexity of the proposed method, we mentioned in Section 4 that the training of polynomial networks with stepwise regression is computationally expensive. In Table 2, the elapsed times of the standard polynomial network and the proposed method on the hand dataset are reported. The experiments were repeated a number of times and the average time in seconds is reported in the table. The elapsed time is captured using Matlab Profiler tool running on an IBM Thinkpad Laptop with Due 1.8GHz CPUs and 2 GB of RAM. Recall that 50% of the dataset is used for training. Both the training and testing times are reported in the table. Note that the training time in the standard polynomial classifier at higher feature dimensionality is also high

because of the exponential growth of the expanded terms. On the other hand, the elapsed time for the testing phase of the proposed solution is more efficient than that of the standard polynomial classifier due to the reduced number of model weights.

Dimensionality prior to expansion (DCT cutoff)	Standard Polynomial classifier		Polynomial classifier with stepwise regression	
	Train (sec)	Test (sec)	Train (sec)	Test (sec)
40	5.8	0.23	13.12	0.17
50	19.58	0.39	19.45	0.195

Table 2. Comparison of training and testing elapsed times.

Lastly, we test our proposed solution on a number of classification datasets taken from the UCI Machine Learning Repository [12]. The datasets are described in Table in 3.

Index	Name	Attribute Types	# Instances	# Attributes	# classes
1-1	Breast Cancer Wisconsin (Diagnostic)	Real	569	32	2
2-4	Ionosphere	Integer, Real	351	34	2
3-5	Iris	Real	150	4	3
4-6	Letter Recognition	Real	20000	16	26
5-7	Lung Cancer	Integer	32	56	3
6-8	MAGIC Gamma Telescope	Real	19020	11	2
7-9	Pima Indians Diabetes	Integer, Real	768	8	2
8-11	Image Segmentation	Real	2310	19	7
9-12	Connectionist Bench (Sonar, Mines vs. Rocks)	Real	208	60	2
10-14	Statlog (Shuttle)	Integer	58000	9	7

Table 3. Description of classification datasets taken from the UCI Machine Learning Repository.

It is worth noting that subspace learning algorithms can also be used to reduce the dimensionality of expanded feature vectors. One efficient subspace learning algorithm is known as Spectral Regression [20] which combines spectral graph analysis and ordinary regression. The algorithm can be applied to the training data to generate a projection matrix. Consequently, this matrix is used to project both the training and the testing datasets into lower dimensionality. In the following experiment we compare the classification results of the proposed solution to both the aforementioned procedure and the reduced model. In the same figure we also report the classification results of the subspace method proposed in [4] since it was also applied to polynomial classifiers. The idea is to reduce the dimensionality of the expanded feature vectors by linearly transforming them into a lower dimensional space. This is achieved by the use of a transformation matrix whose entries are IID

Gaussian. In [4] and references within it is shown that using such a matrix for dimensionality reduction preserves similarity.

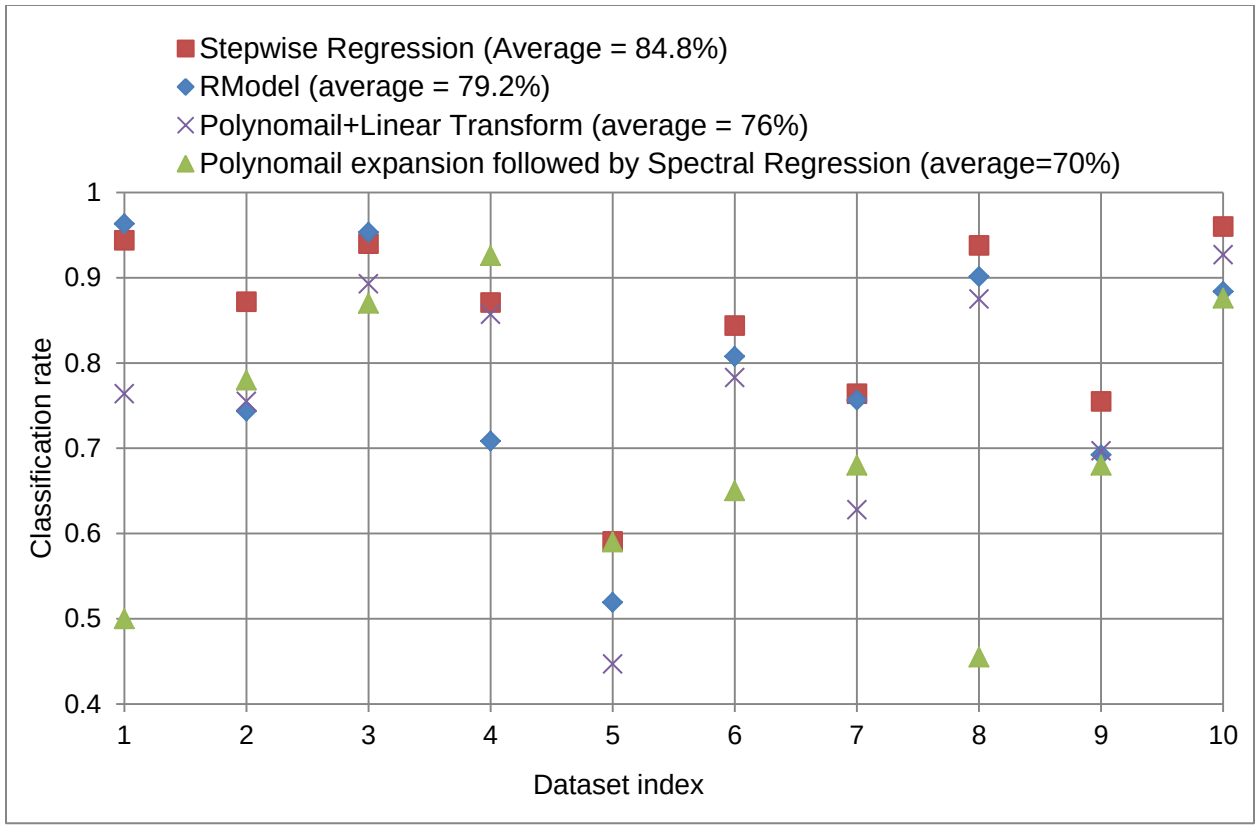


Figure 6. Classification results using 10 datasets from the UCI machine learning repository.

Figure 6 shows a comparison between the classification rates of the proposed solution and the reviewed ones. All training datasets contain 50% of the features vectors in a cross validation fashion. And a third order polynomial is used in expanding the feature vectors. It is shown in the figure that the average classification rate of the proposed solution is 84.8%, while that of the reduced model is 79.2%. The

Polynomial expansion followed by spectral regression achieved an average classification rate of 70%

For a fair comparison with the linear transformation applied to the expanded feature vectors [4] , we stored the dimensions that resulted from the proposed solution and reused them in creating the transformation matrix. As such, the lengths of the final feature vectors are the same in both approaches. Nonetheless, it is shown in the figure that the classification rates of the proposed method are constantly higher than those of the reviewed solutions.

7. Conclusion

In this paper we examined a number of application scenarios for the purpose of verifying the use of polynomial expansion followed by stepwise regression. It was shown that stepwise regression can be used to reduce the dimensionality of expanded feature vectors whilst preserving the discrimination ability. The indices of the selected feature variables or regressosr are stored and used during the testing or validation stage. The proposed work was compared against both standard polynomial networks and recently proposed reduced model multinomial networks. We also compared the proposed algorithm to two other dimensionality reduction techniques; namely, spectral regression and random dimension reduction. The comparisons showed that the proposed technique offered favorable classification accuracy for the majority of the datasets used in this work. It is worthwhile to mention that the computational complexity of the proposed method in the training mode is relatively high. However, training is usually done in an offline mode where computational complexity is not critical as such. Nonetheless, the computational complexity of the proposed method in the testing mode is lower than that of the standard polynomial classifier.

References:

[1] W. M. Campbell,K. T. Assaleh, and C. C. Broun, "Speaker recognition with polynomial classifiers," IEEE Transactions on Speech and Audio Processing , 10(4), pp 205 -212, 2002.

[2] K. T. Assaleh and W. M. Campbell, "Speaker Identification Using a Polynomial-based Classifier," Proc. of the fourth International Symposium on Signal Processing and its Applications ISSPA'99, Brisbane, Australia, August 1999.

[3] W. M. Campbell and K. T. Assaleh, "Low-Complexity Small-Vocabulary Speech Recognition for Portable Devices," Proc. of the fourth International Symposium on Signal Processing and its Applications ISSPA'99, Brisbane, Australia, August 1999.

[4] W. Campbell, K. Torkkola, and S. Balakrishnan, "Dimension Reduction Techniques for Training Polynomial Networks," Proc. Int'l Conf. Machine Learning, June 2000.

[5] Y. Pang, D. Tao, Y. Yuan, and X. Li, "Binary Two-Dimensional PCA," IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 38(4), pp. 1176-1180, 2008.

[6] Y. Pang, Y. Yuan, and X. Li, "Iterative Subspace Analysis Based on Feature Line Distance," IEEE Transactions on Image Processing, 18(4), pp. 903-907, 2009.

[7] P. Chaudhuri, M.-C. Huang, W.-Y. Loh, and R. Yao, "Piecewise-Polynomial Regression Trees," Statistica Sinica, vol. 4, pp. 143-167, 1994.

[8] K.-A Toh, Q.-L. Tran and D. Srinivasan, "Benchmarking a Reduced Multivariate Polynomial Pattern Classifier," IEEE Trans. on pattern analysis and machine intelligence, 26(6), JUNE 2004

[9] K. Assaleh, and H. Al-Nashash, "A Novel Technique for the Extraction of Fetal ECG Using Polynomial Networks," IEEE Trans. on Biomedical Engineering, 52(6), pp. 1148 – 1152, June 2005.

[10] G. Golub and C. Van Loan, "Matrix Computations," John Hopkins, 1989.

[11] D. Montgomery, G. Runger, "Applied statistics and probability for engineers," Wiley, 1994.

[12] C.L. Blake and C.J. Merz, "UCI Repository of Machine Learning Databases," Dept. of Information and Computer Sciences, Univ. of Calif., Irvine, http://www.ics.uci.edu/mlearn/MLRepository.html , 1998.

[13] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," IEEE Trans. on Computers, vol. C-32, pp. 90-93, Jan. 1974.

[14] W.-H. Chen and W.K. Pratt, "Sense Adaptive Coder," IEEE Trans. on Communications, COM-32, pp. 225-232, March 1984.

[15] K. Wong, G. Sainarayanan and A. Chekima," Palmprint Identification Using Wavelet Energy," International Conference on Intelligence and Advance Systems, Kuala Lumpur, Malaysia, November 2007

[16] Xiang-Qian Wu, Kuan-Quan Wang and David Zhang, "Wavelet Based Palmprint Recognition," Proceedings of the First International Conference on Machine Learning and Cybernetics, Beijing, November 2002.

[17] Wenxin Li, David Zhang and Zhuoqun Xu, "Palmprint Identification By Fourier Transform", International Journal of Pattern Recognition and Artificial Intelligence, 16(4), 2002.

[18] T. Shanableh, K. Assaleh, and M. Al-Rousan, "Spatio-temporal Feature Extraction Techniques for Isolated Arabic Sign Language Recognition," IEEE Trans. on Systems, Man and Cybernetics-Part B: Cybernetics, 37(3), June 2007.

[19] T. Shanableh and K. Assaleh, "Telescopic Vector Composition and polar accumulated motion residuals for feature extraction in Arabic Sign Language recognition," EURASIP Journal on Image and Video Processing, vol. 2007, Article ID 87929, 10 pages, 2007. doi:10.1155/2007/87929.

[20] D. Cai, X. He, and J. Han, "SRDA: An Efficient Algorithm for Large Scale Discriminant Analysis", IEEE Transactions on Knowledge and Data Engineering, 20(1), pp. 1-12, 2008.