|  |
| --- |
|  |
| **Evaluation of Dimension Reduction on Features for Leaf Discrimination** |
|  |

**Chanmann Lim Haidong Shen Fernando Torre Mora**

Computer Science Computer Engineering Computer Science

University of Missouri University of Missouri University of Missouri

*cl9p8@missouri.edu hswv8@missouri.edu fthc8@missouri.edu*

**1 Investigators**

**1.1 Chanmann Lim**

To set off our investigation we split the dataset into training and test set using our heuristic function then PCA and Feature Selection methods are performed iteratively to reduce the numbers of features or dimensions of feature spaces before evaluation. In addition the graphs of two features projection from both methods are plotted for our exploratory analysis followed by a series evaluations.

One method conducted in our experiment is Feedforward Neural Network which is one of the most popular methods for classification and statistical pattern recognition task.

**1.2 Haidong Shen**

Using the dataset after PCA and Forward Feature Selection, we do the Minimum Error Classification to analyze the data. For Minimum Error Classification, we do Bayesian Parameter Estimation and Maximum Likelihood Estimation. According to the limit training data, the highest dimension that we do for the MLE and Bayesian Parameter Estimation is 5. We compute the error rate for all the features that we can get and find the best case.

**1.3 Fernando Torre Mora**

To better compare our results to the existing research in the area, and properly evaluate it, we perform the K-Nearest Neighbors Classification algorithm (KNN). For a thorough evaluation of the results that can be obtained using KNN, we perform the algorithm varying K with all possible values and interpret our results both in terms of KNN and in terms of our dimension reduction methods.

Finally, we compare our results from all algorithms with the current state of the art and explain our differences.

**2 Real-world task**

**2.1 Dataset**

We will use the leaf dataset [4], the same one used in [2] and [3]. This dataset was collected by Pauwels et al. using algorithmic image processing methods and tries to encompass most of the features found in the botanical literature. The dataset was made available by Silva et al. (2014)

The dataset classifies 340 leaf specimens into 30 classes, with 14 features computed from quantitative measurements. Each class contains between 8 and 16 samples, with most classes containing 12. A summary of the classes can be seen in Table 1. A list of the features can be seen in Table 2.

Table 1: Number of samples for each class

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | Id | Scientific Name | # | | 1 | Quercus suber | 12 | | 2 | Salix atrocinera | 10 | | 3 | Populus nigra | 10 | | 4 | Alnus sp. | 8 | | 5 | Quercus robur | 12 | | 6 | Crataegus monogyna | 8 | | 7 | Ilex aquifolium | 10 | | 8 | Nerium oleander | 11 | | 9 | Betula pubescens | 14 | | 10 | Tilia tomentosa | 13 | | 11 | Acer palmatum | 16 | | 12 | Celtis sp. | 12 | | 13 | Corylus avellana | 13 | | 14 | Castanea sativa | 12 | | 15 | Populus alba | 10 | | |  |  |  | | --- | --- | --- | | Id | Scientific Name | # | | 22 | Primula vulgaris | 12 | | 23 | Erodium sp. | 11 | | 24 | Bougainvillea sp. | 13 | | 25 | Arisarum vulgare | 9 | | 26 | Euonymus japonicus | 12 | | 27 | Ilex perado ssp. | 11 | | 28 | Magnolia soulangeana | 12 | | 29 | Buxus sempervirens | 12 | | 30 | Urtica dioica | 12 | | 31 | Podocarpus sp. | 11 | | 32 | Acca sellowiana | 11 | | 33 | Hydrangea sp. | 11 | | 34 | Pseudosasa japonica | 11 | | 35 | Magnolia grandiflora | 11 | | 36 | Geranium sp. | 10 | |

Table 2: Attributes of the Data Samples

|  |  |
| --- | --- |
| Identification | Class (Species), Specimen Number |
| Features | Eccentricity, Aspect Ratio, Elongation, Solidity, Stochastic Convexity, Isoperimetric Factor, Maximal Indentation Depth, Lobedness, Average Intensity, Average Contrast, Smoothness, Third moment, Uniformity, Entropy |

**2.2 Background**

Correct identification and classification of leaves is an essential part of outdoor outings, whether they are academic or recreational, as they can be used for food and medicine, and are essential to protection of the environment. [1]. In 2009, Pauwels et al. [2] developed a web service which allowed identifying 36 species using 16 attributes (these attributes can automatically be extracted from a photograph of the leaf). Silva et al. [3] performed an evaluation of how well 8 of these attributes identified 15 of these species, but this evaluation seems incomplete. We will attempt to perform a similar evaluation, and either provide additional support for Silva et al.’s result, or show how they can be improved.

Our base approach is to perform a dimension reduction of the 16 attributes. We will use Principal Component Analysis (PCA) to extract the d’ principal components. We will vary d’ to several values ≤16. This is in contrast to Silva et al. [3] who only performed PCA with d’=2, and with Pauwels et al. [2] whose attempt at dimension reduction consisted in sorting the attributes from most significant to least.

We will evaluate each extracted set of components by performing Maximum Likelihood Classification (MLE), K-Nearest Neighbors (KNN) and Feedforward Neural Network. Pauwels et al. [2] performed an KNN-based classification while Silva et al. [3] performed K-means. However, basic MLE or KNN have not been performed on this dataset

2**.3 State of the Art performance**

A misclassification rate of 12.7% was gotten in [3] by using the Linear Discriminant Analysis (LDA) based on the entire dataset, which was splitted into training data (70%) and testing data (30%).

**3 Methods**

To be able to evaluate all classes equally, we first partitioned the data into a training and a testing set by taking the two last specimens from each dataset. We call the larger set the training set (containing 82 percent of the data) and the smaller set the testing set (containing 18 percent of the data). We performed dimension reduction with both Principal Component Analysis (PCA) and Forward Features Selection (FFS) for all possible dimensions (1 ≤ *d*’ < 14 where *d’* is the number of target dimensions). It is of note that while we did perform PCA for *d*’=1, we did not perform FFS.

We then performed a series of classification algorithms on each of our dimension reduction results to evaluate them. We began with a K-Nearest neighbors algorithm (KNN) as the closest algorithm to [3]. We then performed Minimum Error Classification (MEE) both with Bayesian Error Estimation and Maximum Likelihood Estimation. We also implemented a single-hidden layer Feedforward Neural Network. Finally, we evaluated the accuracy of each trained classifier on the test set.

**3.1 Dimension Reduction**

As exploratory analysis of the data with reduced features or dimensionalities we provide the dispersion when d=2, the result of which is obtained by PCA transformation may be seen in Figure 1. Note that this shows that the classes are not linearly separable.

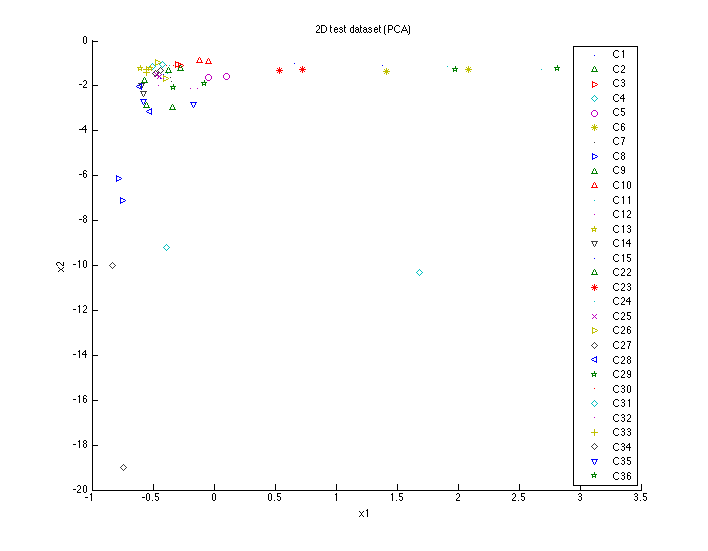


Figure 1: 2D test dataset after performing PCA

For FFS, we use the Least Squares Algorithm to select the most useful features conducive to classification tasks. The features selected are shown in Table 3. Again, we show the dispersion when d=2 in Figure 2.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| # features | Eccentricity | Aspect Ratio | Elongation | Solidity | Stochastic Convexity | Isoperimetric Factor | Maximal Indentation Depth | Lobedness | Average Intensity | Average Contrast | Smoothness | Third moment | Uniformity | Entropy |
| 2 |  |  |  |  |  |  |  |  |  |  |  |  | X | X |
| 3 | X |  |  |  |  |  |  |  |  |  |  |  | X | X |
| 4 | X | X |  |  |  |  |  |  |  |  |  |  | X | X |
| 5 | X | X | X |  |  |  |  |  |  |  |  |  | X | X |
| 6 | X | X | X | X |  |  |  |  |  |  |  |  | X | X |
| 7 | X | X | X | X | X |  |  |  |  |  |  |  | X | X |
| 8 | X | X | X | X | X | X |  |  |  |  |  |  | X | X |
| 9 | X | X | X | X | X | X | X |  |  |  |  |  | X | X |
| 10 | X | X | X | X | X | X | X | X |  |  |  |  | X | X |
| 11 | X | X | X | X | X | X | X | X | X |  |  |  | X | X |
| 12 | X | X | X | X | X | X | X | X | X | X |  |  | X | X |
| 13 | X | X | X | X | X | X | X | X | X | X | X |  | X | X |

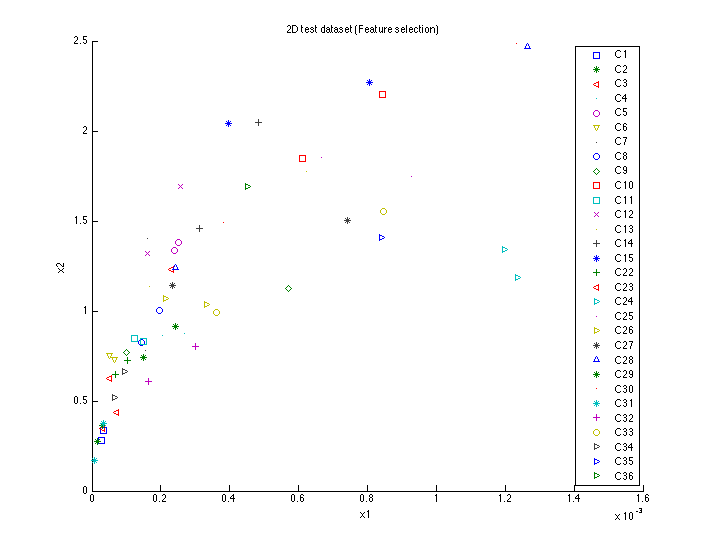


Figure 2. 2D test dataset after performing FFS

**3.2 Classification Algorithms**

For Minimum Error Classification, we do Bayesian Parameter Estimation and Maximum Likelihood Estimation. The prior possibility of training data are different from each other, which is the reason why we do not choose the Maximum likelihood Classification. For Minimum Risk Classification, we have to know the α, which is unknown in this dataset. So we choose Minimum Error Classification.

For K-Nearest neighbors, we also varied the number of neighbors between 1 and 12, since 12 is the mode and average for the number of samples in a class. We list the results of using the full dataset under FFS with 14 dimensions.

For Feedforward Neural Network, we uses single hidden layer neural networks with the number of neurons proportional to the ratio of the number of sample in the training set square and vary by the number of dimensionalities being used

where is the number of the training samples

is the total number of samples

is the number of classes

is the number of dimensionalities used

With the hyperbolic tangent as activation function we empirically tune the learning rate with the fixed decay rate of one of a thousand(0.001) for different number of input features then we train the networks for 1000 epochs with 20 batchsize backpropagation.

**4 Experimental results**

## 4.1 K-Nearest Neighbors Classification

Our best result was an error of 38.33 percent for KNN. This was found both when using 10 neighbors with all classifications performed with 10 FFS-selected features or more, and using 10 neighbors with all classifications performed with 5 PCA components or more. It is possible that 10 is the best number of neighbors because it allows the small classes (8 samples) to have all their members, as well as enough non-members, to correctly classify themselves, while allowing the large classes to have all their neighbors be class members. Our second-best result with KNN was an error of 40 percent. This was found both when using 1 neighbor with 8 FFS-selected features, and using 8 neighbors with 5 PCA components. Several other combinations resulted in this error of 40 percent, however, we consider 8 neighbors and 5 components our best since it gives the smallest dimension. The combinations are listed in Table 4.

Table 4: List of combinations yielding a. an error of 38.33% and b. an error of 40%

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | Dimension Reduction | Neighbors | Dimensions | | FFS | 10 | 10 | |  | 10 | 11 | |  | 10 | 12 | |  | 10 | 13 | |  | 10 | 14 | | PCA | 10 | 5 | |  | 10 | 6 | |  | 10 | 7 | |  | 10 | 8 | |  | 10 | 9 | |  | 10 | 10 | |  | 10 | 11 | |  | 10 | 12 | |  | 10 | 13 | | |  |  |  | | --- | --- | --- | | Dimension Reduction | Neighbors | Dimensions | | FFS | 1 | 8 | |  | 1 | 9 | |  | 9 | 11 | |  | 7 | 12 | |  | 7 | 13 | |  | 7 | 14 | | PCA | 8 | 5 | |  | 7 | 6 | |  | 8 | 6 | |  | 9 | 7 | |  | 9 | 8 | |  | 7 | 9 | |  | 7 | 10 | |  | 7 | 11 | |  | 7 | 12 | |  | 7 | 13 | |

In general, as the number of features and neighbors increased, the error decreased. However, we observed that, after a certain *d*, the error no longer decreased. Rather, it stayed the same, the additional features adding no new information, but also not scattering the points among the dimensions far enough to increase the error. This can be observed as a general flatness in the surfaces plotted in Figure 3, while the best numbers of neighbors can be seen as valleys.

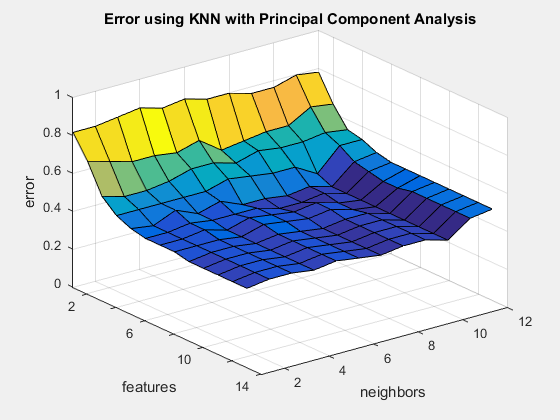
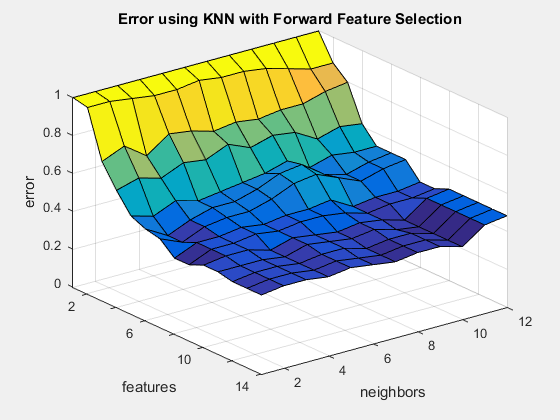
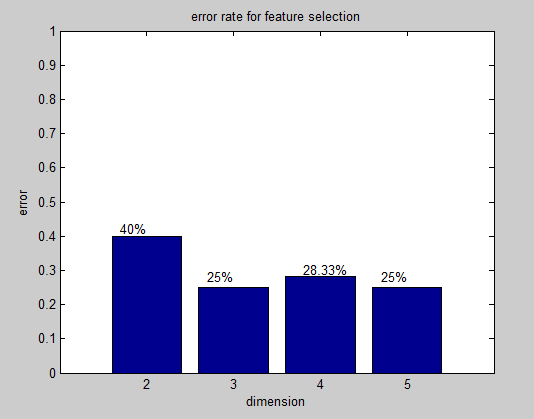
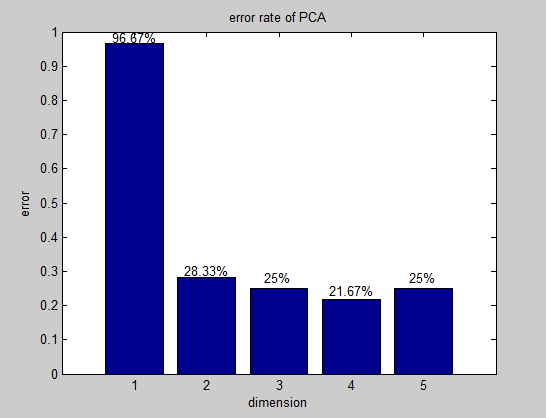


Figure 3: KNN classification results for all values of *k* using a. the results from FFS and b. the results from PCA.

In the FFS results, we additionally observed that, for all values of *k* (the number of neighbors) there was an increase in the error when *d*’ was increased from 5 to 6; an error that is not corrected until *d*’ is raised to 8. This occurs because the first 5 features selected (Eccentricity, Uniformity, Entropy, Aspect Ratio, and Elongation) all measure significantly different features of the leaf. However, the sixth and seventh features (Solidity, Stochastic Convexity) are both defined as measures of convexity [4]. Because convexity is an aspect ratio [6], these features can be seen as redundant with those previously selected. Because the new features do not add useful information, they only serve to spread the data points out among more dimensions, making it harder for KNN to find the appropriate neighbors; thus, the error increases. Once we introduce a non-redundant feature that does add useful information, the error decreases again. This increase in the error can be seen as a ridge in the center of the surface graphed in Figure 3.a.

## 4.2 Minimum Error Classification



Two of the data are used for test data, which means that the largest number of training data, which has the least training data is 6. The training data must at least larger than the dimension. So the largest dimension that we can use for Maximum Likelihood Estimation is 5. We analyze the result from dimension 1 to 5 for the dataset after PCA and the result from dimension 2 to 5 for the dataset after FFS.

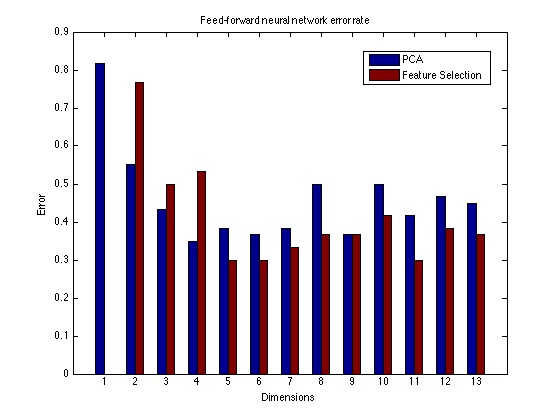
According to the figure above, the X-ray is the number of feature after the PCA and Forward Feature Selection and the Y-ray is the error rate.

For the Maximum Likelihood Estimation, the best result of the dataset after the PCA is dimension 4, the error rate of which is 21.67%. This means we can reduce the features to 4 allowing 21.67% data error rate.

For the Maximum Likelihood Estimation, the best result of the dataset after the Forward Feature Selection is dimension 3, which has the same error rate as dimension 5 but has fewer dimensions. The error rate of dimension 3 is 25%. This means we can reduce the features to 3 allowing 25% data error rate.

For Bayesian parameter estimation, the dimension 5 error rate of the dataset after PCA is 51.67% and of the dataset after FFS is 26.67%.

## 4.3 Feed-Forward Neural Network Classification



**5 Discussion**

When evaluating with PCA, we found our best approximations occurred with either four or five features, depending on the classification algorithm, yielding errors of 21 to 38 percent. When evaluating with FFS, we found our best approximation occurred between three and eight features, yielding errors of 25 to 40 percent. This shows that it is possible to get a good classification (error 25 percent) with less than 8 features, especially if we allow the error to be slightly larger..

To perform a more complete evaluation, we find that more data is needed. MLE as evaluated in our work could not be performed for the PCA and FFS results for dimensions 6-14; a more complete dataset would allow evaluating the accuracy for these reductions in search of an optimal. A greater dataset would also allow us to perform cross-validation in our training, which may yield better accuracies for all algorithms that use it.

However, we must acknowledge that we were unable to improve on the 13 percent error in Silva et al. [3]. We believe this to be because of two reasons. First is that Silva et al. used only 15 classes. It is also possibly influenced by a greater testing set size (30 percent as opposed to 18 percent). The other reason is that the LDA that performed by Silva et al was based on the entire dataset, instead of the dataset after the PCA. This will definitely decrease the error rate. In our future work, we would like to vary the number of classes, as well as the proportion of training and testing set , to evaluate if the 13 percent error given by Silva et al. can be improved.

**References**

[1] Chia-Ling Lee and Shu-Yuan Chen (2006) Classification of leaf images. *International Journal of Imaging Systems and Technology*, vol. 16, issue 1. Wiley Periodicals, Inc. http://onlinelibrary.wiley.com/doi/10.1002/ima.20063/abstract

[2] Pauwels, E.J., de Zeeuw, P.M., and Ranguelova, E. (2009) Computer-assisted tree taxonomy by automated image recognition. *Engineering Applications of Artificial Intelligence*. Elsevier. http://oai.cwi.nl/oai/asset/13914/13914D.pdf

[3] Silva, Pedro F. B.; Marçal, André R. S., and Almeida da Silva, Rubim M. (2013) Evaluation of Features for Leaf Discrimination. *10th International Conference on Image Analysis and Recognition*. Springer. Póvoa do Varzim, Portugal, June 26, 2013 http://link.springer.com/chapter/10.1007%2F978-3-642-39094-4\_23

[4] —— (2014) Leaf Data Set. Machine Learning Repository. *University of California, Irvine*. http://archive.ics.uci.edu/ml/datasets/Leaf

[5] Berg Palm, R (2012). Prediction as a candidate for learning deep hierarchical models of data. Technical *University of Denmark, DTU Informatics*. http://www2.imm.dtu.dk/pubdb/views/publication\_details.php?id=6284

[6] Lee Kamentsky. (2009) “Explanation for Solidity and Extent.” *Cell Profiler.* Broad Insittute. http://www.cellprofiler.org/forum/viewtopic.php?f=2&t=663

[7] Richard O. Duda, Peter E. Hart, David G. Stork (2001) Pattern Classification, 2nd Edition. *A Wiley-Interscience Publication*. John Wiley & Son Inc.