The Application of Principal Component Analysis and Discriminant Analysis on the Classification of Rice Species

.

Introduction

Rice, an edible starchy cereal grain, is one of the most widely consumed staple foods worldwide. In 2019, a study was conducted in Turkey to distinguish between two of their extensively grown rice species, the Osmancik and Cammeo. As a result, 3810 rice grains images were obtained and processed using various image processing techniques. These rice images were obtained using a computer vision system; the rice samples were placed inside an enclosed box with a camera and a lighting system attached. Images of the rice samples were captured and sent to a computer system.

These images were processed using MATLAB, a programming platform. They were then converted to grayscale and binary images, the rice grains on each image were treated separately, and seven morphological features were gathered for each grain. These features were obtained from the shapes found in the images. The description of the features is as follows:

- **Area**: The area returns the number of pixels within the boundaries of the rice grain.
- **Perimeter**: The perimeter calculates the circumference by calculating the distance between pixels around the boundaries of the rice grain.
- **MajorAxisLength**: The longest line that can be drawn on the rice grain.
- **MinorAxisLength**: The shortest line that can be drawn on the rice grain.
- **Eccentricity**: A measure of how round the ellipse of the rice grain is.
- **Convex Area**: This returns the pixel count of the smallest convex shell of the region formed by the rice grain.
- **Extent**: This returns the ratio of the region formed by the rice rain to the bounding box pixels.

With these features, the 3,810 rice grains were distributed into Osmancik and Cammeo species, with 2,180 of the rice grains being Osancik and 1,630 of the rice grains being Cammeo.

Figure: Description of the Dataset

Area	Perimeter	Major_Axis_Length	Minor_Axis_Length	Eccentricity	Convex_Area	Extent	Class
15231	525.5789795	229.7498779	85.09378815	0.928882003	15617	0.572895527	Cammeo
14656	494.3110046	206.0200653	91.73097229	0.895404994	15072	0.615436316	Cammeo
14634	501.1220093	214.106781	87.76828766	0.912118077	14954	0.693258822	Cammeo
13176	458.3429871	193.3373871	87.44839478	0.891860902	13368	0.640669048	Cammeo
14688	507.1669922	211.7433777	89.31245422	0.906690896	15262	0.646023929	Cammeo
13479	477.0159912	200.0530548	86.65029144	0.901328325	13786	0.657897294	Cammeo
15757	509.2810059	207.2966766	98.33613586	0.88032347	16150	0.58970809	Cammeo
16405	526.5700073	221.6125183	95.43670654	0.902520597	16837	0.65888828	Cammeo
14534	483.6409912	196.6508179	95.05068207	0.875428557	14932	0.649651349	Cammeo
13485	471.5700073	198.272644	87.72728729	0.896789312	13734	0.572319865	Cammeo
14930	499.9249878	212.2458191	90.01747894	0.905606449	15248	0.624372721	Cammeo
14626	496.5859985	204.5341339	92.97486877	0.890711546	15070	0.57021445	Cammeo
15926	522.7399902	225.7360535	91.05709076	0.915033162	16240	0.779768884	Cammeo
14076	479.677002	199.489151	90.70998383	0.890638888	14434	0.781218767	Cammeo
13500	476.9150085	202.5466766	85.4054718	0.906754851	13800	0.717703342	Cammeo
14349	496.9460144	213.5440216	86.16077423	0.914988399	14678	0.666837096	Cammeo
15209	496.5650024	214.0500793	91.02632141	0.90507257	15395	0.569369555	Cammeo
15238	496.8710022	208.5317841	93.82839966	0.893054903	15487	0.732314467	Cammeo
13509	480.4660034	207.1371613	83.94016266	0.914210558	13732	0.595634937	Cammeo

Project Goal

This project aims to use principal component and discriminant analysis as dimension reduction techniques to create an optimal rule of classifying rice grains as either Osmancik or Cammeo species.

Methods

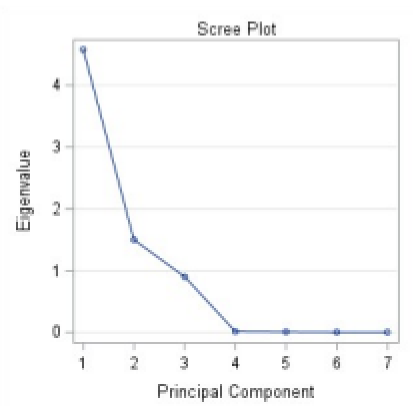
Principal Component Analysis

The principal component analysis is a technique for discovering the true dimensionality of the space in which the data lies. It does so by transforming the set of original variables into a new set containing variables uncorrelated with each other. These variables, called principal components, are created so that the first few account for the total variation in the original dataset. The PCA technique was performed on the correlation matrix as the variables were not on equal footing (different measurements).

Correlation Matrix								
		Area	Perimeter	Major_Axis_Length	Minor_Axis_Length	Eccentricity	Convex_Area	Extent
Area	Area	1.0000	0.9665	0.9030	0.7878	0.3521	0.9989	-.0612
Perimeter	Perimeter	0.9665	1.0000	0.9719	0.6298	0.5446	0.9699	-.1309
Major_Axis_Length	Major_Axis_Length	0.9030	0.9719	1.0000	0.4521	0.7109	0.9034	-.1396
Minor_Axis_Length	Minor_Axis_Length	0.7878	0.6298	0.4521	1.0000	-.2917	0.7873	0.0634
Eccentricity	Eccentricity	0.3521	0.5446	0.7109	-.2917	1.0000	0.3527	-.1986
Convex_Area	Convex_Area	0.9989	0.9699	0.9034	0.7873	0.3527	1.0000	-.0658
Extent	Extent	-.0612	-.1309	-.1396	0.0634	-.1986	-.0658	1.0000

To choose the number of principal components, we implemented two methods:

- We looked at the Scree plot, a line plot of the eigenvalues of the principal components.



The scree plot shows that the elbow is at 3, so we chose two principal components by using the equation: k=elbow-1, where k is the number of principal components.

- We looked at the Eigenvalues of the Correlation matrix table. From this, we excluded the components with eigenvalues less than 1.

Eigenvalues of the Correlation Matrix				
	Eigenvalue	Difference	Proportion	Cumulative
1	4.57897926	3.07922059	0.6541	0.6541
2	1.49975867	0.59895326	0.2143	0.8684
3	0.90080541	0.88904959	0.1287	0.9971
4	0.01175581	0.00553916	0.0017	0.9988
5	0.00621666	0.00416399	0.0009	0.9996
6	0.00205266	0.00162113	0.0003	0.9999
7	0.00043154		0.0001	1.0000

After selecting the number of principal components, we computed the equation for the first two principal components using this formula: $y_i = a_i^T x$ where i=1, 2

Eigenvectors

	Prin1	Prin2
Area	0.461252	0.124377
Perimeter	0.464408	-.055751
Major_Axis_Length	0.447076	-.213456
Minor_Axis_Length	0.321752	0.567105
Eccentricity	0.227329	-.673152
Convex_Area	0.461694	0.122535
Extent	-.057716	0.382232

- The first principal component:

  $$y_1 = 0.461Area + 0.464Perimeter + 0.447Major\_Axis\_Length + 0.322Minor\_Axis\_Length + 0.227Eccentricity + 0.462Convex\_Area - 0.0577Extent$$

  The first principal component accounts for 65% of the total variation in the data.

- The second principal component:

$$y_2 = 0.124Area - 0.0558Perimeter - 0.214Major\_Axis\_Length + 0.567Minor\_Axis\_Length - 0.673Eccentricity + 0.123Convex\_Area + 0.382Extent$$

The second principal component is the contrast between the MajorAxisLength, Perimeter, Eccentricity, and Area, MinorAxisLength, ConvexArea, and Extent. This is due to the Eigenvalues corresponding to the first three features being negative and those corresponding to the last four features being positive. We can also see from the equation that Perimeter has little effect on the second principal component.

After acquiring the principal components, we calculated the principal component scores, which are their values for each experimental unit in the dataset.

Values of the first 2 PC Scores

Obs	Prin1	Prin2
1	3.81213	-2.16505
2	2.47683	0.04529
3	2.63821	-0.62153
4	0.54779	-0.15138
5	2.81366	-0.48240
6	1.19845	-0.51207
7	3.50352	1.17370
8	4.47479	0.40302
9	1.97141	1.30204
10	1.14473	-0.65505

Discriminant and Classification Analysis

These multivariate techniques involve separating two or more groups of observations based on the variables measured on each experimental unit and creating an optimal rule for allocating new observations into the labeled classes.

Holdout Method

This technique involves partitioning the dataset into two parts; the calibration (training) and the holdout (testing). The calibration dataset is used to create the discriminant function, and the holdout dataset is used to evaluate the performance of the discriminant function.

Analysis of the Principal Component Scores

The data was partitioned so that 50% of the dataset was for training, while the other 50% was for testing. We tested the following hypotheses with the training dataset to determine which classification rule was more convenient:

H_0: $\mu_1 = \mu_2 vs\ H_1: \mu_1 \neq \mu_2$ and H_0: $\varepsilon_1 = \varepsilon_2 vs\ H_1: \varepsilon_1 \neq \varepsilon_2 \neq \varepsilon_k$

Using the **pool=test option,** which provides a test of the equality of the within covariance matrices**,** and **the manova option,** which provides a test of the equality of the mean vectors, we got the following output:

Multivariate Statistics and Exact F Statistics					
S=1 M=0 N=950					
Statistic	Value	F Value	Num DF	Den DF	Pr > F
Wilks' Lambda	0.29150413	2311.39	2	1902	<.0001
Pillai's Trace	0.70849587	2311.39	2	1902	<.0001
Hotelling-Lawley Trace	2.43048317	2311.39	2	1902	<.0001
Roy's Greatest Root	2.43048317	2311.39	2	1902	<.0001

Chi-Square	DF	Pr > ChiSq
39.541131	3	<.0001

From the output, we rejected the null hypothesis H_0: $\mu_1 = \mu_2$, implying that the mean vectors are unequal. This rejection further implies that discrimination analysis is applicable. We also rejected the null hypothesis H_0: $\varepsilon_1 = \varepsilon_2$, indicating that the within covariance matrices are unequal. These assumptions show that a quadratic classification rule is best suited.

Analysis of the Original Dataset and Subset of the Variables

Here, we performed the same holdout technique on the original data set with seven continuous variables. After this, we tested the hypotheses using the calibration data:

H_0: $\mu_1 = \mu_2 vs\ H_1: \mu_1 \neq \mu_2$ and H_0: $\varepsilon_1 = \varepsilon_2 vs\ H_1: \varepsilon_1 \neq \varepsilon_2$

Multivariate Statistics and Exact F Statistics					
S=1 M=0.5 N=949.5					
Statistic	Value	F Value	Num DF	Den DF	Pr > F
Wilks' Lambda	0.27557511	1665.77	3	1901	<.0001
Pillai's Trace	0.72442489	1665.77	3	1901	<.0001
Hotelling-Lawley Trace	2.62877471	1665.77	3	1901	<.0001
Roy's Greatest Root	2.62877471	1665.77	3	1901	<.0001

Chi-Square	DF	Pr > ChiSq
162.117600	6	<.0001

From the output, we rejected both hypotheses implying that the mean vectors and the covariance matrices are unequal. These assumptions show that a quadratic classification rule is best suited.

Using the Calibration data, we also performed the **stepdisc** procedure to see which variables were necessary for effective discrimination.

Stepwise Selection Summary											
Step	Number In	Entered	Removed	Label	Partial R-Square	F Value	Pr > F	Wilks' Lambda	Pr < Lambda	Average Squared Canonical Correlation	Pr > ASCC
1	1	Major_Axis_Length		Major_Axis_Length	0.7139	4747.86	<.0001	0.28612849	<.0001	0.71387151	<.0001
2	2	Perimeter		Perimeter	0.0052	9.89	0.0017	0.28464876	<.0001	0.71535124	<.0001
3	3	Minor_Axis_Length		Minor_Axis_Length	0.0032	6.16	0.0132	0.28373003	<.0001	0.71626997	<.0001
4	4	Convex_Area		Convex_Area	0.0224	43.44	<.0001	0.27738739	<.0001	0.72261261	<.0001
5	5	Area		Area	0.0071	13.51	0.0002	0.27542836	<.0001	0.72457164	<.0001
6	4		Major_Axis_Length	Major_Axis_Length	0.0005	0.87	0.3522	0.27555394	<.0001	0.72444606	<.0001
7	3		Perimeter	Perimeter	0.0001	0.15	0.7024	0.27557511	<.0001	0.72442489	<.0001

The table shows that the most important variables were the Minor Axis length, Convex Area, and Area. We created classification rules using the seven continuous variables and the essential variables obtained from the selection procedure as discriminators.

Results

To evaluate the performance of the quadratic discriminant rule on the holdout data, we created a confusion matrix to compute the misclassification rate. The prior probabilities were accounted for with $p_1 = 0.428\ and\ p_2 = 0.572$.

Performance on the Principal Component Scores

Number of Observations and Percent Classified into Class			
From Class	Cammeo	Osmancik	Total
Cammeo	715 87.73	100 12.27	815 100.00
Osmancik	69 6.33	1021 93.67	1090 100.00
Total	784 41.15	1121 58.85	1905 100.00
Priors	0.42782	0.57218	

From the table, we got the following conclusions:

- 100 rice samples were misclassified as Osmancik
- 69 rice samples were misclassified as Cammeo

- The Error misclassification rate:

$$ECM = 0.4782 * 0.1227 + 0.5722 * 0.0633 = 0.089$$

Performance on the Original data with all variables as discriminators

Number of Observations and Percent Classified into Class			
From Class	Cammeo	Osmancik	Total
Cammeo	736 90.31	79 9.69	815 100.00
Osmancik	82 7.52	1008 92.48	1090 100.00
Total	818 42.94	1087 57.06	1905 100.00
Priors	0.42782	0.57218	

From the table, we got the following conclusions:

- 79 rice samples were misclassified as Osmancik
- 82 rice samples were misclassified as Cammeo
- The Error misclassification rate:

$$ECM = 0.4782 * 0.097 + 0.5722 * 0.075 = 0.085$$

Performance on the selected variables

Number of Observations and Percent Classified into Class			
From Class	Cammeo	Osmancik	Total
Cammeo	715 87.73	100 12.27	815 100.00
Osmancik	73 6.70	1017 93.30	1090 100.00
Total	788 41.36	1117 58.64	1905 100.00
Priors	0.42782	0.57218	

From the table, we got the following conclusions:

- 100 rice samples were misclassified as Osmancik

- 73 rice samples were misclassified as Cammeo
- The Error misclassification rate:

$$ECM = 0.4782 * 0.1227 + 0.5722 * 0.0670 = 0.091$$

Conclusion

In this study, we examined whether dimension-reduction techniques such as principal component analysis and discrimination analysis can effectively create discriminators to derive an optimal rule for classifying rice grains as Osmanick or Cammeo. We created principal component scores using the PCA technique and implemented the holdout method on both the PCA scores and the original datasets to partition the datasets into two; training and testing datasets.

We used the stepwise selection procedure on the training dataset to acquire the best variables that can be used as discriminators. Then, using those selected variables and the principal component scores, we created a quadratic classification rule, and its performance was tested using the holdout dataset.

We calculated the error misclassification rate for each classification rule. We got a value of 0.09, indicating that the classification rule did a good job classifying rice grains as Osmancik or Cammeo.

References

Cinar, I., & Koklu, M. (2019). Classification of rice varieties using artificial intelligence methods. *International Journal of Intelligent Systems and Applications in Engineering*, *7*(3), 188–194. https://doi.org/10.18201/ijisae.2019355381

Chatfield, C., & Collins, A. J. (1980). *Introduction to multivariate analysis*. Chapman and Hall.

Lesson 11: Principal Components Analysis (PCA): Stat 505. PennState: Statistics Online Courses. (n.d.). Retrieved November 20, 2022, from https://online.stat.psu.edu/stat505/lesson/11

Lesson 10: Discriminant Analysis: Stat 505. PennState: Statistics Online Courses. (n.d.). Retrieved November 20, 2022, from https://online.stat.psu.edu/stat505/lesson/10

APPENDIX

SAS CODE

A. Principal Component Analysis

```
PROC IMPORT OUT= WORK.Rice
            DATAFILE= "Q:\Rice_Cammeo_Osmancik.xlsx"
            DBMS=EXCEL REPLACE;
     RANGE="Rice$";
     GETNAMES=YES;
     MIXED=NO;
     SCANTEXT=YES;
     USEDATE=YES;
     SCANTIME=YES;
RUN;

PROC PRINCOMP DATA=WORK.Rice  OUT=PCSCORES ;
VAR Area--Extent;
run;

proc print data=PCSCORES;
VAR  PRIN1-PRIN2;
Title 'Values of the first 2 PC Scores';
RUN;


PROC EXPORT
DATA=PCSCORES
outfile="Q:\Pcscores2.xlsx"
dbms=xlsx;
run;
```

B. DISCRIMINANT AND HOLDOUT PROCEDURE - PC SCORES

```
PROC IMPORT OUT= WORK.Pscores
            DATAFILE= "O:\Pcscores2.xlsx"
            DBMS=EXCEL REPLACE;
     RANGE="Pcscores2";
     GETNAMES=YES;
     MIXED=NO;
     SCANTEXT=YES;
     USEDATE=YES;
     SCANTIME=YES;
RUN;

data Work.Pscores;
set Work.Pscores;
id=_N_;
run;

data Pc;
set Work.Pscores; n1=1630; n2=2180; n1w=815; n2w=1090;

data tr1; set Pc;
if(id>0 & id<=n1w);

data tr2; set Pc;
if(id>n1 & id<=(n1+n2w));

data tra; set tr1 tr2;
proc print data=tra;

data t1; set Pc;
if (id>n1w & id<=n1);

data t2; set Pc;
if (id>(n1+n2w) & id<=(n1+n2));

data tes; set t1 t2;


proc discrim data=tra
method=normal pool=test manova testdata= tes;
class Class;
priors prop;
var Prin1 Prin2;
```

C. VARIABLE SELECTION PROCEDURE AND PROC DISCRIM

```
    SCANTIME=YES;
RUN;

data Work.RiceProject;
set Work.RiceProject;
id=_N_;
run;

data Rice;
set Work.RiceProject; n1=1630; n2=2180; n1w=815; n2w=1090;

data tr1; set Rice;
if(id>0 & id<=n1w);

data tr2; set Rice;
if(id>n1 & id<=(n1+n2w));

data tra; set tr1 tr2;
proc print data=tra;

data t1; set Rice;
if (id>n1w & id<=n1);

data t2; set Rice;
if (id>(n1+n2w) & id<=(n1+n2));

data tes; set t1 t2;


PROC STEPDISC DATA=tra sle=0.05 sls=0.05;
class Class;
var Area--Extent;
run;

proc discrim data=tra
method=normal pool=test manova testdata= tes;
class Class;
priors prop;
var Area Minor_Axis_Length Convex_Area;
run;
```