STAT 8320 Spring 2015 Assignment 6

Peng Shao 14221765

May 5, 2015

▶ **1.** **Solution.** (a). Because PCA is not invariant with respect to changes in scale, that is, the variable which is measured under larger scale will have larger variance, then produce a large eigenvalue when performing singular value decomposition. So standardization avoids the problems of having one variable with large variance unduly influencing the determination of factor loadings, and the correlation matrix is the covariance matrix of standardized variable.

(b). Before PCA, we should investigate the data roughly, and we find that there may be an outlier which have the total score less than 6000. We have no idea why cause this abnormality, but we should take it out of the following analysis because the PCA is very sensitive to outliers. Then we do a PCA based on the 10 single scores.

Eigenvalues of the Correlation Matrix				
	Eigenvalue	Difference	Proportion	Cumulative
1	3.41823814	0.81184501	0.3418	0.3418
2	2.60639314	1.66309673	0.2606	0.6025
3	0.94329641	0.06527516	0.0943	0.6968
4	0.87802124	0.32139459	0.0878	0.7846
5	0.55662665	0.06539914	0.0557	0.8403
6	0.49122752	0.06063230	0.0491	0.8894
7	0.43059522	0.12379709	0.0431	0.9324
8	0.30679812	0.03984871	0.0307	0.9631
9	0.26694941	0.16509526	0.0267	0.9898
10	0.10185415		0.0102	1.0000

Figure 1: Principal Component Proportion

From Figure 1, the first 2 PCs have accounted for 60.25% variability of the total.

(c). Because the first 2 PCs account for more than 50% variability of the data, and the only these 2 PCs have the eigenvalues more than 1.

(d). According to the part of PC table (Figure 2), the first principal component

Eigenvectors

	Prin1	Prin2	Prin3	Prin4	Prin5
run100	0.415882	-.148808	-.267472	-.088332	-.442314
Ljump	0.394051	-.152082	0.168949	0.244250	-.368914
shot	0.269106	0.483537	-.098533	0.107763	0.009755
Hjump	0.212282	0.027898	0.854987	-.387944	0.001876
run400	0.355847	-.352160	-.189496	0.080575	0.146965
hurdle	0.433482	-.069568	-.126160	-.382290	-.088803
discus	0.175792	0.503335	-.046100	-.025584	-.019359
polevlt	0.384082	0.149582	-.136872	-.143965	0.716743
javelin	0.179944	0.371957	0.192328	0.600466	-.095582
run1500	0.170143	-.420965	0.222552	0.485642	0.339772

Figure 2: First 5 Principal Components

may be a kind of overall means of all scores, not exactly equal to the average because of some slightly different weights; and the second principal component may be the comparison between the scores of running add long jump and those of throwing add pole vault.

(e).

From the scatter plot Figure 3, the data points have already been labeled as the their rank. So we can see that most high rank athletes (here higher rank means higher total score but smaller value of the rank) have comparably high 1st PC score. However, it is hard to see that whether there is any pattern of 2nd PC score relative to the ranks of athletes.

Hence, to make the relationship between PC scores and the ranks (or the total scores) more clearly, we can plot the total score versus two PC scores respectively(Figure 4 and Figure 5). From Figure 4, we can see that the 1st PC score has positive relationship with total score indeed, and we also believe that there should be no significant relationship between 2nd PC score and total score.

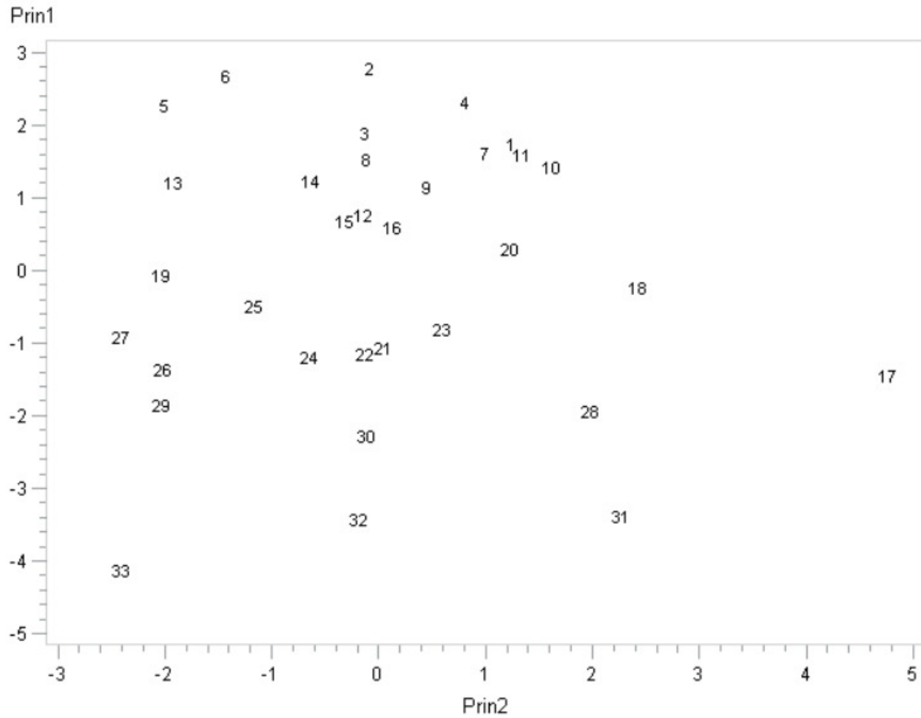

Figure 3: 1st PC v.s. 2nd PC

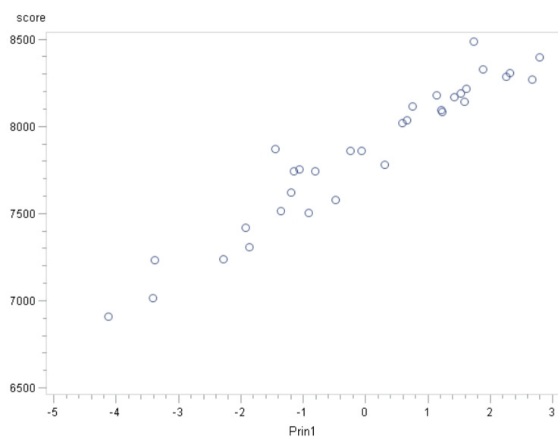

Figure 4: Total Score v.s. 1st PC

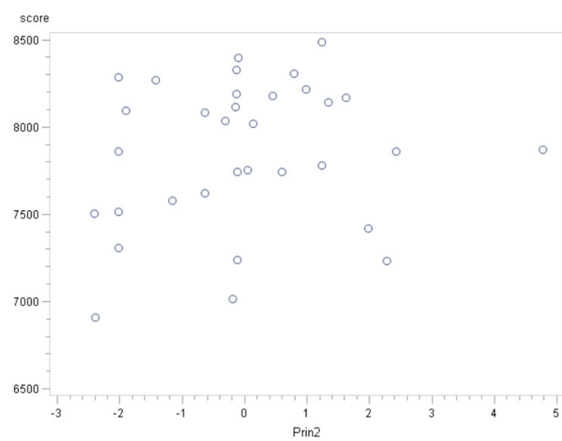

Figure 5: Total Score v.s. 2nd PC

(f).

Pearson Correlation Coefficients, N = 33 Prob > |r| under H0: Rho=0			
	score	Prin1	Prin2
score	1.00000	0.96158	0.16194
		<.0001	0.3679
Prin1	0.96158	1.00000	0.00000
	<.0001		1.0000
Prin2	0.16194	0.00000	1.00000
	0.3679	1.0000	

Figure 6: Correlation Matrix among PCs

To compute the Pearson correlation coefficients among the total score, 1st PC score and 2nd PC score. The correlation coefficient between total score and 1st PC score is 0.96158 while that between total score and 2nd PC score is only 0.16194. This is in accordance with the result of last part. Thus, we may guess that the total score should be measured in a way like 1st PC, i.e., a kind of overall mean, not some comparisons.

▶ **2. Solution.** (a). Because X_i are standardized random variables(zero mean and one standard deviation), then the covariance of X_i is exactly the correlation of X_i. So

$$\begin{aligned} corr(X_i, X_k) &= cov(X_i, X_k) = cov(a_iF + e_i, a_kF + e_K) \\ &= a_ia_kvar(F) + a_icov(F, e_k) + a_kcov(e_i, F) + cov(e_i, e_k) = a_ia_k \\ corr(X_j, X_k) &= cov(X_j, X_k) = cov(a_jF + e_j, a_kF + e_K) \\ &= a_ja_kvar(F) + a_jcov(F, e_k) + a_kcov(e_j, F) + cov(e_j, e_k) = a_ja_k \end{aligned}$$

where i, j, k are mutually different. Thus the ratio of pair of rows (i, j) is always

$$\frac{corr(X_i, X_k)}{corr(X_j, X_k)} = \frac{a_i}{a_j}$$

for all $k \neq i, j$.

(b).

$$corr(\boldsymbol{X}) = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \end{pmatrix} \begin{pmatrix} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 \end{pmatrix} + var(\boldsymbol{e})$$

(c).

$$\boldsymbol{X} = \boldsymbol{a}F + \boldsymbol{e}$$

where

- $\boldsymbol{X} = \begin{pmatrix} X_1 & X_2 & \cdots & X_6 \end{pmatrix}' \sim \mathrm{N}(\boldsymbol{0}, corr(\boldsymbol{X}))$
- $F \sim \mathrm{N}(0,1)$
- $\boldsymbol{a} = \begin{pmatrix} a_1 & a_2 & \cdots & a_6 \end{pmatrix}'$
- $\boldsymbol{e} = \begin{pmatrix} e_1 & e_2 & \cdots & e_6 \end{pmatrix}' \sim \mathrm{N}(\boldsymbol{0}, \boldsymbol{\Psi})$
- F and $\boldsymbol{e}$ are independent.

(d). For my perspective, only one factor should be included. Because

1. only the eigenvalue of the 1st factor is larger than 1, no matter how many factor we include in the analysis, and the proportion of variability accounted by the 1st factor is at least 100%.
2. to test the hypothesis

$$H_0: \text{ 1 Factor is sufficient}$$

   the P-value is 0.9805, so we cannot reject the null hypothesis, that is, more factors are not necessary.

3. the AIC and BIC of one factor are lower than those of two factors.

(e). The result of "Factor Pattern" can give the information about factor loadings of different variable, i.e. the a_i in the model in part (c). For example, the variable C has a loading of 0.95611(a very high loading) in Factor1, and the variable F has a loading of 0.87081(a moderately to high loading) in Factor1, and so on. All variables have a at least moderately to high loading in Factor1, and the loadings are not different so much.

In addition, we can know how much proportion of variance of each variable is accounted by the communality from the "Factor Pattern" output or "Final Communality" output. For example,

$$\text{Communality of C} = a_1^2 = 0.95611^2 = 0.9141$$

which means 91.41% variability of C comes from common factor and only 8.69% variability come from specific factor.

(f). No. If the rotation is help, let T become the rotation operator. Then we can have that

$$F^* = TF$$

where the F^* is the factor after rotating. So the F^* should also have the restriction of length equal to 1, i.e. $var(F^*) = 1$. This implies that $T^2 = 1$, and so $T = \pm 1$. So rotation cannot change the value of loadings except the sign. Thus, there is no need to perform a rotation.

(g). Because in part (e), the loadings of different variable are not exactly same, so the ratio of any two loadings are you 1, as part (a). states. and from the part (d), we only include one factor, so the correlation coefficient directly determined by the loadings. Then the correlation coefficients should not be same. The test result is not so consistent with the conclusion of part (d) and part (e). But we cannot say either of them is wrong, it is very probability that the test is not power enough to prove the alternative hypothesis of different correlation coefficient.

► **3. Solution.** (a).

Significance Tests Based on 123 Observations

Test	DF	Chi-Square	Pr > ChiSq
H0: No common factors HA: At least one common factor	36	400.8045	<.0001
H0: 2 Factors are sufficient HA: More factors are needed	19	58.9492	<.0001

Figure 7: Likelihood Tests

No. From the second test in Figure 7, the P-value less than 0.0001 means we should reject the null hypothesis, that is, two factors is not sufficient.

(b). Yes. From the second test in Figure 8, the P-value=0.1100 means we do not reject the null hypothesis, that is, 3 factors is sufficient.

From the Figure 9, we can see that $p1, p3, p4, p8, p9$ have high loadings in factor 1, and all there statements have mentioned doctor. So maybe the factor 1 is a "doctor factor". $p6, p7$ have high loadings in factor 2, and all these statements is about themselves without some specific reasons. So maybe the factor 2 is a "subjective personal factor". $p2, p5$ have high loadings in factor 3, and all these statements is

Significance Tests Based on 123 Observations

Test	DF	Chi-Square	Pr > ChiSq
H0: No common factors HA: At least one common factor	36	400.8045	<.0001
H0: 3 Factors are sufficient HA: More factors are needed	12	18.1926	0.1100

Figure 8: Likelihood Tests

Rotated Factor Pattern

	Factor1	Factor2	Factor3
p1	0.65061	-0.36388	0.18922
p2	-0.12303	0.19762	0.65038
p3	0.79194	-0.14394	0.11442
p4	0.72594	-0.10131	-0.08998
p5	0.01951	0.30112	0.64419
p6	-0.08648	0.82532	0.36929
p7	-0.22303	0.59741	0.32525
p8	0.81511	0.06018	-0.30809
p9	0.43540	-0.07642	-0.22784

Figure 9: Rotated Factors

also about themselves but more reasonable. So maybe the factor 3 is a "objective personal factor".

(c). From the Figure 10, we can see that the factor pattern does not change

Rotated Factor Pattern			
	Factor1	Factor2	Factor3
p3	0.77055	0.08960	-0.15725
p8	0.76602	-0.25549	0.01478
p4	0.73756	-0.09466	-0.05899
p1	0.62890	0.13760	-0.34898
p9	0.44388	-0.24117	-0.09942
p5	0.00703	0.64140	0.27762
p2	-0.11237	0.62330	0.16815
p6	-0.10471	0.41384	0.67610
p7	-0.21240	0.34957	0.61389

Figure 10: Rotated Factors

too much between MLE and principle factor method; the clusters are still same: $\{p3, p8, p4, p1, p9\}$, $\{p5, p2\}$, $\{p6, p7\}$, only the loadings change slightly. But the interpretation becomes a little different. Because principle factor method uses the philosophy of PCA, so the factor interpretation may be like the interpretation of PCs. That is, factor 1 is the overall mean of $\{p3, p8, p4, p1, p9\}$, factor 2 is the overall mean of $\{p5, p2\}$ and factor 3 is the overall mean of $\{p6, p7\}$.

(d). The overall result of principle factor method with oblique rotation is not so

Rotated Factor Pattern (Standardized Regression Coefficients)			
	Factor1	Factor2	Factor3
p8	0.80194	-0.23698	0.14769
p3	0.76327	0.15455	-0.08062
p4	0.75363	-0.05702	0.04426
p1	0.57149	0.23910	-0.31583
p9	0.44088	-0.21029	-0.02242
p2	-0.08493	0.60139	0.09812
p5	0.06408	0.60138	0.23230
p6	0.04218	0.27883	0.68179
p7	-0.08328	0.22180	0.60645

Figure 11: Oblique Rotated Factors

different from the result of other methods. It also only has some differences about factor loadings. The clusters and the interpretations of factors are same as the principle factor method with orthogonal rotation. However, for oblique rotation, we should also see the correlation between latent factors. In the problem the maximum correlation coefficient is 0.33129, which is not significant. So we can stop here. If the correlation between factors is significant, we can go on and perform a factor analysis to get the second order factor, which may contain more general information.

▶ **3. Solution.** (a). The covariance matrices for spam and not-spam seem not same but we cannot perform most test because of the rank deficiency: the 41st attribute of not-spam is all 0.

Given the different correlation matrix, we should use the quadratic discriminant analysis.

(b). Before all following analysis, we want to mention some index used in evaluating the performance of classifiers.

- **Precision**
$$P = \frac{TP}{TP + FP}$$
- **Recall**
$$R = \frac{TP}{TP + FN}$$
- **F-measure**
$$F = \frac{2 \cdot R \cdot P}{R + P}$$
- **Error Rate**
$$ER = \frac{FP + FN}{P + N}$$
- **Missing Alarm**
$$MA = 1 - R$$
- **False Alarm**[1]
$$FA = 1 - P$$

[1]Here I choose false alarm instead of false positive rate because false alarm is more sensitive to the misclassification of negative case when positive cases are more rare. For example, we have 50 non-spam emails and 9 spam emails, and only one non-spam email is misclassified as spam, then the false positive rate is 2% while false alarm is 10%.

For the first three index, the larger the better; they give us the ability of prediction of the classifier; for the last three index, the lower the better; they give us the probabilities of misclassification of the classifier.

The DISCRIM Procedure
Classification Summary for Calibration Data: DA2.H7Q4
Cross-validation Summary using Linear Discriminant Function

Number of Observations and Percent Classified into spam			
From spam	**0**	**1**	**Total**
	2374	110	2484
0	95.57	4.43	100.00
	345	1272	1617
1	21.34	78.66	100.00
	2719	1382	4101
Total	66.30	33.70	100.00
	0.60571	0.39429	
Priors			

Error Count Estimates for spam			
	0	**1**	**Total**
Rate	0.0443	0.2134	0.1109
Priors	0.6057	0.3943	

Figure 12: Cross-validation for Linear Discriminant Funciton

For cross-validation, we have

$$TP = 1272 \quad FP = 110 \quad TN = 2374 \quad FN = 345$$

then we can compute that

$$P = 92.04\% \quad R = 78.66\% \quad F = 84.82\%$$
$$ER = 11.09\% \quad MA = 21.34\% \quad FA = 7.95\%$$

For test sample, we have

$$TP = 147 \quad FP = 20 \quad TN = 284 \quad FN = 49$$

then we can compute that

$$P = 88.02\% \quad R = 75\% \quad F = 80.99\%$$
$$ER = 13.8\% \quad MA = 25\% \quad FA = 11.97605\%$$

The DISCRIM Procedure
Classification Summary for Test Data: DA2.H7Q4TEST
Classification Summary using Linear Discriminant Function

Observation Profile for Test Data	
Number of Observations Read	500
Number of Observations Used	500

Number of Observations and Percent Classified into spam			
From spam	0	1	Total
0	284	20	304
	93.42	6.58	100.00
1	49	147	196
	25.00	75.00	100.00
Total	333	167	500
	66.60	33.40	100.00
Priors	0.60571	0.39429	

Error Count Estimates for spam			
	0	1	Total
Rate	0.0658	0.2500	0.1384
Priors	0.6057	0.3943	

Figure 13: Test for Linear Discriminant Funciton

It is very reasonable that the classifier has better performance on cross-validation than on testing sample, so the linear discriminant function is not bad.

(c).

For cross-validation, we have

$$TP = 1547 \quad FP = 644 \quad TN = 1851 \quad FN = 70$$

then we can compute that

$$P = 70.61\% \quad R = 95.67\% \quad F = 81.25\%$$
$$ER = 17.36\% \quad MA = 4.33\% \quad FA = 29.39\%$$

For test sample, we have

$$TP = 187 \quad FP = 95 \quad TN = 209 \quad FN = 9$$

then we can compute that

$$P = 66.31\% \quad R = 95.41\% \quad F = 78.24\%$$
$$ER = 20.8\% \quad MA = 4.59\% \quad FA = 33.69\%$$

The DISCRIM Procedure
Classification Summary for Calibration Data: DA2.H7Q4
Cross-validation Summary using Quadratic Discriminant Function

Number of Observations and Percent Classified into spam			
From spam	0	1	Total
0	1851	633	2484
	74.52	25.48	100.00
1	70	1547	1617
	4.33	95.67	100.00
Total	1921	2180	4101
	46.84	53.16	100.00
Priors	0.60571	0.39429	

Error Count Estimates for spam			
	0	1	Total
Rate	0.2548	0.0433	0.1714
Priors	0.6057	0.3943	

Figure 14: Cross-validation for Quadratic Discriminant Funciton

The DISCRIM Procedure
Classification Summary for Test Data: DA2.H7Q4TEST
Classification Summary using Quadratic Discriminant Function

Observation Profile for Test Data	
Number of Observations Read	500
Number of Observations Used	500

Number of Observations and Percent Classified into spam			
From spam	0	1	Total
0	209	95	304
	68.75	31.25	100.00
1	9	187	196
	4.59	95.41	100.00
Total	218	282	500
	43.60	56.40	100.00
Priors	0.60571	0.39429	

Error Count Estimates for spam			
	0	1	Total
Rate	0.3125	0.0459	0.2074
Priors	0.6057	0.3943	

Figure 15: Test for Quadratic Discriminant Funciton

To compare the linear classifier, the quadratic classifier has higher recall and lower missing alarm, which indicates that it is excellent in detecting spam email; while in the meantime the false alarm become pretty large; it misclassify over 30 non-spam emails as spam emails, which could be very harmful.

(d).

The DISCRIM Procedure
Classification Summary for Test Data: DA2.H7Q4TEST
Classification Summary using Linear Discriminant Function

Observation Profile for Test Data	
Number of Observations Read	500
Number of Observations Used	500

Number of Observations and Percent Classified into spam			
From spam	0	1	Total
	280	24	304
0	92.11	7.89	100.00
	36	160	196
1	18.37	81.63	100.00
	316	184	500
Total	63.20	36.80	100.00
	0.5	0.5	
Priors			

Error Count Estimates for spam			
	0	1	Total
Rate	0.0789	0.1837	0.1313
Priors	0.5000	0.5000	

Figure 16: Test for Step-selection Linear Discriminant Funciton

For test sample, we have

$$TP = 184 \quad FP = 24 \quad TN = 280 \quad FN = 36$$

then we can compute that

$$\begin{aligned} &P = 88.46\% \quad R = 83.64\% \quad F = 85.98\% \\ &ER = 11.45\% \quad MA = 16.36\% \quad FA = 11.54\% \end{aligned}$$

Comparing to the two discriminant functions above, the linear discriminant function after stepwise selection has become a very good model. Because it has the highest F score, the lowest error rate and lowest false alarm for test sample so far,

Model	P	R	F	ER[2]	MA	FA
2-Nearest Neighbors Classifier	99.44%	99.79%	99.61%	8.46%	0.21%	0.56%
3-Nearest Neighbors Classifier	94.34%	93.11%	93.72%	5.32%	6.89%	5.66%
5-Nearest Neighbors Classifier	92.82%	89.54%	91.14%	7.07%	10.46%	7.18%
1-Radius Kernal Classifier	99.94%	98.69%	99.31%	0.83%	1.31%	0.06%
10-Radius Kernal Classifier	90%	13.38%	23.30%	34.8%	86.62%	10%

Table 1: Performances on Training data

Model	P	R	F	ER	MA	FA
2-Nearest Neighbors Classifier	93.69%	92.76%	93.23%	5.32%	7.23%	6.31%
3-Nearest Neighbors Classifier	89.40%	86.02%	87.68%	9.53%	13.98%	10.60%
5-Nearest Neighbors Classifier	89.54%	85.78%	87.62%	9.56%	14.22%	10.46%
1-Radius Kernal Classifier	99.37%	48.98%	65.61%	64.5%	51.02%	0.63%
10-Radius Kernal Classifier	88.44%	12.31%	21.61%	36.04%	87.69%	11.56%
Logistic Regression[3]	90.43%	86.73%	88.54%	8.8%	13.27%	9.57%

Table 2: Performances on Cross-Validation

which are what we are concerned about most. And its missing alarm is much less than the linear discriminant function with all attributes.

(e). The models we try are

- Nearest Neighbor Classifier, K=2,3,5;
- Kernel Method, R=1,10;
- Logistic Regression.

All result about the models are listed from Figure 17 to Figure 18. And the performances of different models on training data and cross-validation are summarized in Table 1 and Table 2.

We have two basic goals:

1. to detect the spam email more efficiently, i.e. hope the recall as large as possible;

[2]The computation of ER here and below is different from above, because non parametrical method allow classify a observation as other, which is neither false positive or false negative, but also treated as misclassification

[3]For logistic regression here, the result is the performance on test sample

2. not to misclassify a non-spam email as spam email, i.e., hope the false alarm as small as possible.

We also want to maximize the F score and minimize the error rate. Accord to these criterions, it is not so hard to find out that the "best" model should be the 2-Nearest Neighbors Classifier: the highest recall and F-score, the lowest error rate and false alarm (except the 0.63% of 1-Radius Kernel Classifier).

It is also very interesting that when the number of neighbors increases, the nearest neighbor model performs more poorly both on training data and cross-validation. We should also notice that the kernel model fits the training data better as the radius becomes larger, but it has already overfitted because of the worst performance on cross-validation.

The DISCRIM Procedure
Classification Summary for Calibration Data: DA2.H7Q4
Resubstitution Summary using 2 Nearest Neighbors

Number of Observations and Percent Classified into spam				
From spam	0	1	Other	Total
	2321	8	155	2484
0	93.44	0.32	6.24	100.00
	3	1433	181	1617
1	0.19	88.62	11.19	100.00
	2324	1441	336	4101
Total	56.67	35.14	8.19	100.00
	0.60571	0.39429		
Priors				

Error Count Estimates for spam			
	0	1	Total
Rate	0.0656	0.1138	0.0846
Priors	0.6057	0.3943	

The DISCRIM Procedure
Classification Summary for Calibration Data: DA2.H7Q4
Cross-validation Summary using 2 Nearest Neighbors

Number of Observations and Percent Classified into spam			
From spam	0	1	Total
	2383	101	2484
0	95.93	4.07	100.00
	117	1500	1617
1	7.24	92.76	100.00
	2500	1601	4101
Total	60.96	39.04	100.00
	0.60571	0.39429	
Priors			

Error Count Estimates for spam			
	0	1	Total
Rate	0.0407	0.0724	0.0532
Priors	0.6057	0.3943	

Figure 17: Confusion Table of 2-Nearest Neighbors Classifier

Table of F_spam by I_spam				
		I_spam(Into: spam)		
		0	1	Total
F_spam(From: spam)				
0	Frequency	286	18	304
1	Frequency	26	170	196
Total	Frequency	312	188	500

Figure 18: Confusion Table of Logistic Regression

The DISCRIM Procedure
Classification Summary for Calibration Data: DA2.H7Q4
Resubstitution Summary using 3 Nearest Neighbors

Number of Observations and Percent Classified into spam

From spam	0	1	Other	Total
0	2383	90	11	2484
	95.93	3.62	0.44	100.00
1	111	1500	6	1617
	6.86	92.76	0.37	100.00
Total	2494	1590	17	4101
	60.81	38.77	0.41	100.00
Priors	0.60571	0.39429		

Error Count Estimates for spam

	0	1	Total
Rate	0.0407	0.0724	0.0532
Priors	0.6057	0.3943	

The DISCRIM Procedure
Classification Summary for Calibration Data: DA2.H7Q4
Cross-validation Summary using 3 Nearest Neighbors

Number of Observations and Percent Classified into spam

From spam	0	1	Total
0	2319	165	2484
	93.36	6.64	100.00
1	226	1391	1617
	13.98	86.02	100.00
Total	2545	1556	4101
	62.06	37.94	100.00
Priors	0.60571	0.39429	

Error Count Estimates for spam

	0	1	Total
Rate	0.0664	0.1398	0.0953
Priors	0.6057	0.3943	

Figure 19: Confusion Table of 3-Nearest Neighbors Classifier

The DISCRIM Procedure
Classification Summary for Calibration Data: DA2.H7Q4
Resubstitution Summary using 5 Nearest Neighbors

Number of Observations and Percent Classified into spam

From spam	0	1	Other	Total
0	2364	112	8	2484
	95.17	4.51	0.32	100.00
1	169	1447	1	1617
	10.45	89.49	0.06	100.00
Total	2533	1559	9	4101
	61.77	38.02	0.22	100.00
Priors	0.60571	0.39429		

Error Count Estimates for spam

	0	1	Total
Rate	0.0483	0.1051	0.0707
Priors	0.6057	0.3943	

The DISCRIM Procedure
Classification Summary for Calibration Data: DA2.H7Q4
Cross-validation Summary using 5 Nearest Neighbors

Number of Observations and Percent Classified into spam

From spam	0	1	Total
0	2322	162	2484
	93.48	6.52	100.00
1	230	1387	1617
	14.22	85.78	100.00
Total	2552	1549	4101
	62.23	37.77	100.00
Priors	0.60571	0.39429	

Error Count Estimates for spam

	0	1	Total
Rate	0.0652	0.1422	0.0956
Priors	0.6057	0.3943	

Figure 20: Confusion Table of 5-Nearest Neighbors Classifier

The DISCRIM Procedure
Classification Summary for Calibration Data: DA2.H7Q4
Resubstitution Summary using Uniform Kernel Density

Number of Observations and Percent Classified into spam				
From spam	0	1	Other	Total
	2479	1	4	2484
0	99.80	0.04	0.16	100.00
	21	1588	8	1617
1	1.30	98.21	0.49	100.00
	2500	1589	12	4101
Total	60.96	38.75	0.29	100.00
	0.60571	0.39429		
Priors				

Error Count Estimates for spam			
	0	1	Total
Rate	0.0020	0.0179	0.0083
Priors	0.6057	0.3943	

The DISCRIM Procedure
Classification Summary for Calibration Data: DA2.H7Q4
Cross-validation Summary using Uniform Kernel Density

Number of Observations and Percent Classified into spam				
From spam	0	1	Other	Total
	664	5	1815	2484
0	26.73	0.20	73.07	100.00
	29	792	796	1617
1	1.79	48.98	49.23	100.00
	693	797	2611	4101
Total	16.90	19.43	63.67	100.00
	0.60571	0.39429		
Priors				

Error Count Estimates for spam			
	0	1	Total
Rate	0.7327	0.5102	0.6450
Priors	0.6057	0.3943	

Figure 21: Confusion Table of 1-Radius Kernal Classifier

The DISCRIM Procedure
Classification Summary for Calibration Data: DA2.H7Q4
Resubstitution Summary using Uniform Kernel Density

Number of Observations and Percent Classified into spam				
From spam	0	1	Other	Total
	2458	24	2	2484
0	98.95	0.97	0.08	100.00
	1398	216	3	1617
1	86.46	13.36	0.19	100.00
	3856	240	5	4101
Total	94.03	5.85	0.12	100.00
	0.60571	0.39429		
Priors				

Error Count Estimates for spam			
	0	1	Total
Rate	0.0105	0.8664	0.3480
Priors	0.6057	0.3943	

The DISCRIM Procedure
Classification Summary for Calibration Data: DA2.H7Q4
Cross-validation Summary using Uniform Kernel Density

Number of Observations and Percent Classified into spam				
From spam	0	1	Other	Total
	2424	26	34	2484
0	97.58	1.05	1.37	100.00
	1401	199	17	1617
1	86.64	12.31	1.05	100.00
	3825	225	51	4101
Total	93.27	5.49	1.24	100.00
	0.60571	0.39429		
Priors				

Error Count Estimates for spam			
	0	1	Total
Rate	0.0242	0.8769	0.3604
Priors	0.6057	0.3943	

Figure 22: Confusion Table of 10-Radius Kernal Classifier