Εφαρμοσμένη Πολυμεταβλητή Ανάλυση και Big Data

Καλλιαμπάκος Μάριος, Κατσιγιάννης Θεόφιλος, Τριανταφυλλάκης Εμμανουήλ

17 Ιουνίου 2025

Εργασία 1

Αρχικά φορτώνουμε τα δεδομένα από μορφή csv, κατόπιν μιας μικρής τροποποίησης των δεδομένων, η μετατροπή των ονομάτων των στηλών σε λατινικούς χαρακτήρες και διαγραφή της στήλης Α/Α. Η φόρτωση των δεδομένων έγινε με τον κώδικα

```
  # Load csv file
data <- read.csv("data.csv")
```

Listing 1: R code

Τα δεδομένα φορτώνονται στην μεταβλητή `data`.

(α') Στο ερώτημα αυτό θα κάνουμε μια γραμμική παλινδρόμηση και θα ελέγξουμε όλες τις βασικές προϋποθέσεις. Τις χωρίσαμε κατά ομάδες ως εξής:

1. Εκτίμηση του μοντέλου. Για τη δημιουργία του μοντέλου χρειαζόμαστε τον κώδικα

```
 # Set the linear model
  model <-lm(Y   P + M1 + M2 + M3 + M4 + SCORE + I, data = data)
  summary(model)

  # Anova Table
  anova(model)
```

Listing 2: R code

όπου P είναι η μεταβλητή Π. Το αποτέλεσμα της `summary{model}` είναι

```
Call:
lm(formula = Y ~ P + M1 + M2 + M3 + M4 + SCORE + I, data = data)

Residuals:
    Min      1Q  Median      3Q     Max
-1.4762 -0.2820 -0.1562  0.4745  0.9508

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.43495    3.37626   0.425 0.679828
P            1.79567    0.88522   2.029 0.069978 .
M1           0.19348    0.21570   0.897 0.390811
M2           0.22718    0.24095   0.943 0.367979
M3          -0.10842    0.34068  -0.318 0.756853
M4           0.02837    0.24127   0.118 0.908713
```

```
SCORE          0.07774    0.08022   0.969 0.355364
I              0.99673    0.16976   5.872 0.000157 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.8053 on 10 degrees of freedom
  (3 observations deleted due to missingness)
Multiple R-squared:  0.8702, Adjusted R-squared:  0.7794
F-statistic:  9.58 on 7 and 10 DF,  p-value: 0.0009738
```

Επίσης λάβαμε τον πίνακα `anova`

```
Analysis of Variance Table

Response: Y
          Df  Sum Sq Mean Sq F value    Pr(>F)
P          1 18.3405 18.3405 28.2819 0.0003387 ***
M1         1  0.8080  0.8080  1.2459 0.2904323
M2         1  0.4461  0.4461  0.6878 0.4262555
M3         1  0.0091  0.0091  0.0141 0.9079527
M4         1  0.0093  0.0093  0.0143 0.9071752
SCORE      1  1.5161  1.5161  2.3379 0.1572563
I          1 22.3567 22.3567 34.4752 0.0001570 ***
Residuals 10  6.4849  0.6485
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Στη συνέχεια σχεδιάζουμε τα γραφήματα που αντιστοιχούν στο γραμμικό μοντέλο με τη βοήθεια του κώδικα

```r
# Graphs
par(mfrow = c(2,2))
plot(model)
```

Listing 3: R code

γραπη1.πνγ

Σχήμα 1: Όλα τα βασικά `plots` που αντιστοιχούν στο γραμμικό μοντέλο.

Θα περιγράψουμε κάθε ένα από τα παραπάνω γραφήματα

Το διάγραμμα Residual vs Fitted. Το διάγραμμα αυτό περιγράφει τη σχέση μεταξύ των προβλεπόμενων τιμών $\hat{Y}_i$ και των αντίστοιχων υπολοίπων $e_i = Y_i - \hat{Y}_i$. Το διάγραμμα αυτό μπορεί να μας δώσει οπτικοποίηση για το αν η σχέση $Y \sim X$ είναι γραμμική, όπου X το σύνολο των ανεξαρτήτων μεταβλητών στο μοντέλο της γραμμικής παλινδρόμησης. Δηλαδή αν η υπόθεση

$$H_0 : \text{η συσχέτιση } Y \sim X \text{ είναι γραμμική}$$

μπορεί να απορριφθεί ή όχι. Από γραφικής άποψης όσο τα σημεία είναι κοντά στην ευθεία γραμμή $y = 0$ τόσο κοντά είναι να δεχθούμε την υπόθεση H_0. Επίσης μπορούμε

να ελέγξουμε την ομοσκεδαστικότητα. Η ομοσκεδαστικότητα μπορεί γραφικά να αναπαρασταθεί με την κυρτότητα της καμπύλης των υπολοίπων. Όσο μεταβάλλεται η καμπύλη τόσο η διασπορά των υπολοίπων είναι ανόμοια. Επομένως σε αυτόν την περίπτωση έχουμε ετεροσκεδαστικότητα.
Στην περίπτωση του γραφήματος της γραμμικής παλινδρόμησης φαίνεται μια μικρή κυρτότητα της καμπύλης των υπολοίπων το οποίο υποδηλώνει την πιθανή μη γραμμική σχέση της μεταβλητής Y και των υπολοίπων ανεξάρτητων μεταβλητών ή πιθανή ετεροσκεδαστικότητα στα υπόλοιπα.

Το διάγραμμα Q-Q plot. Με το διάγραμμα αυτό ελέγχουμε την κανονικότητα των υπολοίπων. Στο μοντέλο παλινδρόμησης τα υπόλοιπα συγκεντρώνονται κοντά στην γραμμή της κανονικής κατανομής αλλά υπάρχουν και σημεία που αποκλείνουν από αυτήν αρκετά. Άρα τα υπόλοιπα είναι περίπου κανονικά, αλλά η κανονικότητα θα εξετασθεί παρακάτω με τη χρήση του ελέγχου Shapiro-Wilk.

Το διάγραμμα Scale-Location. Το συγκεκριμένο διάγραμμα ελέγχει με γραφικό τρόπο την ομοσκεδαστικότητα των υπολοίπων. Υπολογίζει την σχέση των τυποποιημένων υπολοίπων $\sqrt{|\hat{e}_i|}$ με πεδίο ορισμού τις προβλεπόμενες τιμές του μοντέλου.
Μια οριζόντια 'νεφέλη' σημείων υποδηλώνει την ομοσκεδαστικότητα των υπολοίων, ενώ μια αύξουσα ή φθίνουσα τάση των σημείων υποδηλώνει ετεροσκεδαστικότητα.
Στην περίπτωση του μοντέλου που μελετάμε τα υπόλοιπα έχουν αυξητική τάση για μεγάλες τιμές τιμές του $\hat{Y}$. Πράγμα που υποδηλώνει τάση για ετεροσκεδαστικότητα.

Γράφημα Residuals-Leverage. Εντοπίζει σημεία με υψηλή επιρροή στο μοντέλο. Το σημείο 1 παρουσιάζει μεγάλο leverage και υψηλό standarized residual άρα είναι πιθανό σημείο επιρροής. Το σημείο 9 παρουσιάζει τη μικρότερη αρνητική τιμή ως υπόλοιπο και ταυτόχρονα έχει μικρό leverage. Άρα είναι outlier. Γενικά όλα τα υπόλοιπα δεν είναι επιδραστικά για το γραμμικό μοντέλο αφού είναι κάτω από τη γραμμή του Cook εκτός από το σημείο 1.

2. **Έλεγχος βασικών υποθέσεων (στατιστικά και γραφικά).** Θα κάνουμε τους παρακάτω στατιστικούς ελέγχους.

   i. **Κανονικότητα των υπολοίπων.** Για τον έλεγχο κανονικότητας των υπολοίπων λαμβάνουμε τα `residuals` του μοντέλου

```
res <- residuals(model)
```

Listing 4: R code

και εκτελούμε:

(α΄) **shapiro-wilk έλεγχο**

```
# Normality of the residuals
shapiro.test(res)
```

Listing 5: R code

από τον οποίο λαμβάνουμε

```
Shapiro-Wilk normality test
data:  res
W = 0.93361, p-value = 0.2246
```

Το παραπάνω τεστ δείχνει ότι σε επίπεδο σημαντικότητας $\alpha = 0.05$ επειδή $p - value = 0.2246 > \alpha = 0.05$ **δεν** απορρίπτουμε την υπόθεση της κανονικότητας των υπολοίπων.

(β΄) **Lilliefors έλεγχο**. Έχουμε τον κώδικα

```r
library(nortest)
lillie.test(res)
```

Listing 6: R code

Από το οποίο λαμβάνουμε το αποτέλεσμα

```
Lilliefors (Kolmogorov-Smirnov) normality test

data:  res
D = 0.15189, p-value = 0.3327
```

Όπως και προηγουμένως επειδή $p - value = 0.3327 > \alpha = 0.05$ δεν μπορούμε να απορρίψουμε την υπόθεση της κανονικότητας των υπολοίπων.

(γ΄) **Anderson-Darling έλεγχο**. Έχουμε τον κώδικα

```r
ad.test(res)
```

Listing 7: R code

Από το οποίο λαμβάνουμε το αποτέλεσμα

```
Anderson-Darling normality test

data:  res
A = 0.5091, p-value = 0.1721
```

Εντελώς με παράμοια λογική και εδώ δεν απορρίπτουμε της υπόθεση της κανονικότητας για τα υπόλοιπα.

Συμπέρασμα. Συνεπώς με βάση τα τρία παραπάνω τέστ δεν μπορούμε να απορρίψουμε την υπόθεση της κανονικότητας των υπολοίπων.

ii. **Ομοσκεδαστικότητα (σταθερή διασπορά).** Για την ομοσκεδαστικότητα θα εκτελούμε έναν `Bruesch-Pagan` έλεγχο με τον παρακάτω κώδικα

```r
library(zoo)
library(lmtest)
bptest(model)
```

Listing 8: R code

Από το οποίο λαμβάνουμε το αποτέλεσμα

```
studentized Breusch-Pagan test

data:  model
BP = 6.8618, df = 7, p-value = 0.4434
```

Επειδή $p - value = 0.4434 > \alpha = 0.05$ δεν παραβιάζεται ο κανόνας της σταθερής διασποράς των υπολοίπων.

iii. **Γραμμικότητα.** Για τη μελέτη της γραμμικότητας εκτελούμε δύο διαφορετικούς ελέγχους

Γραφικό έλεγχο με avPlots Χρειαζόμαστε τον κώδικα

```r
# Linearity av Plots
```

```r
library(car)
avPlots(model, ask = FALSE)
```

Listing 9: R code

Από το οποίο λαμβάνουμε το σύστημα των γραφημάτων.

γραπη3.πνγ

Σχήμα 2: Γράφηματα από τη χρήση της `avPlots`.

Τα παραπάνω γραφήματα παρουσιάζουν τη συσχέτιση κάθε μεταβλητής με την εξαρτημένη αν εξαιρεθούν οι υπόλοιπες μεταβλητές. Με βάση τα παραπάνω γραφήματα μπορούμε να συμπεράνουμε τα εξής.

Η μεταβλητή P**.** Παρουσιάζει μια θετική κλίση προς τα πάνω, η οποία αποτελεί ισχυρή ένδειξη γραμμικής σχέσης.

Η μεταβλητή $M1$**.** Παρουσιάζει ελαφρά θετική σχέση με την εξαρτημένη μεταβλητή και με υψηλή διασπορά σημείων. Επομένως υπάρχει ασθενής γραμμική σχέση.

Η μεταβλητή $M2$**.** Παρουσιάζει ελαφρά θετική σχέση και με υψηλη διασπορά σημείων. Ομοίως υπάρχει ασθενής γραμμική σχέση.

Η μεταβλητή $M3$**.** Παρουσιάζει ελαφρά αρνητική κλίση και υπάρχει αρκετός θόρυβος στα δεδομένα. Συνέπεια αυτού είναι ότι δεν υπάρχει σαφής γραμμική σχέση.

Η μεταβλητή $M4$**.** Παρουσιάζει μια ευθεία σχεδόν οριζόντια. Άρα έχει μηδενική ή αμελητέα γραμμική σχέση.

Η μεταβλητή $SCORE$**.** Με την ίδια λογική η μεταβλητή έχει μια συσχέτιση ελαφρά θετικής κλίσης. Επομένως εμφανίζει μια μικρή γραμμική σχέση.

Η μεταβλητή I**.** Παρουσιάζει συσχέτιση ισχυρή θετική κλίση και επομένως έχει μια ισχυρή γραμμική συσχέτιση με την Y.

3. Έλεγχος πολυσυγγραμμικότητας. Κάνουμε ένα `vif` έλεγχο και έχουμε

```r
# Multi-linearity Check
library(car)
vif(model)
```

Listing 10: R code

Από το οποίο λαμβάνουμε τα αποτελέσματα

```
       P       M1       M2       M3       M4    SCORE        I
3.211168 1.702984 1.712230 1.958790 1.622011 1.308837 1.066999
```

Με βάση τα παραπάνω αποτελέσματα έχουμε ότι όλες οι μεταβλητές έχουν ανεκτή συσχέτιση μεταξύ τους αφού οι `vif` τιμές είναι μικρότερες του 5 και πολύ μακριά του 10.

(β') Αρχικά θα κατασκευάσουμε τον covariance πίνακα και τον correlation πίνακα. Ο κώδικας είναι

```r
# Find the covariance matrix
S <- cov(data, use = "pairwise.complete.obs")
print(round(S, 4))
```

```r
# Find the correlation matrix
P <- cor(data, use = "pairwise.complete.obs")
print(round(P,4))
```

Listing 11: R code

Από τον οποίο λαμβάνουμε τα αποτελέσματα. Για τον πίνακα S έχουμε:

```
              P      M1      M2      M3      M4   SCORE      I       Y
P        0.1450  0.0684  0.1555  0.0888  0.1496 -0.2825 0.1333  0.4112
M1       0.0684  1.2071 -0.1750 -0.2768 -0.0554  0.0989 0.0668  0.4203
M2       0.1555 -0.1750  1.0083  0.0583 -0.0208 -0.7917 0.0752  0.5028
M3       0.0888 -0.2768  0.0583  0.6119  0.0399  0.4183 0.0448  0.1320
M4       0.1496 -0.0554 -0.0208  0.0399  0.9655 -0.2361 0.0472  0.3264
SCORE   -0.2825  0.0989 -0.7917  0.4183 -0.2361  7.7590 0.1007 -0.0165
I        0.1333  0.0668  0.0752  0.0448  0.0472  0.1007 1.3843  1.6146
Y        0.4112  0.4203  0.5028  0.1320  0.3264 -0.0165 1.6146  2.8252
```

Για τον πίνακα P έχουμε

```
             P       M1       M2       M3       M4    SCORE       I        Y
P       1.0000  0.1634  0.4067  0.2981  0.3999 -0.2565 0.2976  0.6425
M1      0.1634  1.0000 -0.1586 -0.3220 -0.0513  0.0300 0.0517  0.2276
M2      0.4067 -0.1586  1.0000  0.0743 -0.0211 -0.2680 0.0637  0.2979
M3      0.2981 -0.3220  0.0743  1.0000  0.0519  0.1872 0.0486  0.1004
M4      0.3999 -0.0513 -0.0211  0.0519  1.0000 -0.0822 0.0408  0.1976
SCORE  -0.2565  0.0300 -0.2680  0.1872 -0.0822  1.0000 0.0304 -0.0035
I       0.2976  0.0517  0.0637  0.0486  0.0408  0.0304 1.0000  0.8164
Y       0.6425  0.2276  0.2979  0.1004  0.1976 -0.0035 0.8164  1.0000
```

Στη συνέχεια θα κατασκευάσουμε τον πίνακα με τις `p-values` με τον κώδικα

```r
  # Create the p-value matrix for each pair
p_matrix <- matrix(NA, nrow = length(vars), ncol = length(vars))
colnames(p_matrix) <- vars
rownames(p_matrix) <- vars

for(i in 1:(length(vars)-1)){
  for(j in (i+1):length(vars)){
    test <- cor.test(data[[i]], data[[j]])
    p_matrix[i, j] <- test$p.value
    p_matrix[j, i] <- test$p.value
  }
}

print(round(p_matrix, 4))
```

Listing 12: R code

Από τον οποίο παίρνουμε τον πίνακα `p_matrix`

```
             P     M1     M2     M3     M4  SCORE      I      Y
P           NA 0.4791 0.0673 0.1893 0.0725 0.3042 0.1902 0.0017
M1      0.4791     NA 0.4922 0.1545 0.8253 0.9058 0.8240 0.3211
M2      0.0673 0.4922     NA 0.7490 0.9276 0.2824 0.7840 0.1896
M3      0.1893 0.1545 0.7490     NA 0.8233 0.4571 0.8342 0.6650
M4      0.0725 0.8253 0.9276 0.8233     NA 0.7457 0.8605 0.3905
SCORE   0.3042 0.9058 0.2824 0.4571 0.7457     NA 0.9046 0.9891
I       0.1902 0.8240 0.7840 0.8342 0.8605 0.9046     NA 0.0000
Y       0.0017 0.3211 0.1896 0.6650 0.3905 0.9891 0.0000     NA
```

Πάνω στον πίνακα αναζήτηση για τιμές που είναι κάτω από τον συντελεστή εμπιστοσύνης 0.05. Έτσι γράφουμε

```r
  which(p_matrix < 0.05, arr.ind = TRUE)
```

Listing 13: R code

Από το οποίο λαμβάνουμε το αποτέλεσμα

```
     row col
Y    8   1
Y    8   7
P    1   8
I    7   8
```

Συμπέρασμα. Με βάση το προηγούμενο αποτέλεσμα οι μόνες ισχυρές συσχετίσεις με την εξαρτημένη μεταβλητή Y είναι αυτή των P, I. Όλες οι υπόλοιπες μεταβλητές θα μπορούσαν να παραληφθούν.

Στο τέλος θα σχεδιάσουμε το γράφημα συσχέτισης με τη βοήθεια του κώδικα

```r
library(corrplot)
corrplot(P, method = "circle", type = "upper", tl.cex = 0.8)
```

Listing 14: R code

Από το οποίο λαμβάνουμε το γράφημα

γραπη6.πνγ

Σχήμα 3: Γράφημα συσχέτισης για το γραμμικό μοντέλο.

Από το παραπάνω γράφημα παρατηρούμε ότι

Συμπέρασμα.. Υπάρχει γραμμική συσχέτιση μεταξύ των ανεξαρτήτων μεταβλητών που πρέπει να ληφθεί υπόψην στην υπόθεση της πολυσυγγραμμικότητας. Πιο συγκεκριμένα:

Η μεταβλητή $M2$ Συσχετίζεται ισχυρά με την P.

Η μεταβλητή $M4$ Συσχετίζεται ισχυρά με την P.

(γ΄) Για την εφαρμογή της PCA επί του εκτιμητή του πίνακα Σ χρειαζόμαστε πρώτα να αποκλείσουμε την εξαρτημένη μεταβλητή Y από τον αρχικό πίνακα των δεδομένων, γράφοντας τον κώδικα

```r
 X <-data[, !(names(data) %in% "Y")]
```

Listing 15: R code

Στη συνέχεια υπολογίζουμε τον εκτιμητή πίνακα S

```r
S <- cov(X)
print(round(S, 4))
```

Listing 16: R code

Από τον οποίο λαμβάνουμε τον πίνακα S

```
              P       M1       M2       M3       M4  SCORE       I
P      0.1450  0.0684  0.1555  0.0888  0.1496     NA 0.1333
M1     0.0684  1.2071 -0.1750 -0.2768 -0.0554     NA 0.0668
M2     0.1555 -0.1750  1.0083  0.0583 -0.0208     NA 0.0752
M3     0.0888 -0.2768  0.0583  0.6119  0.0399     NA 0.0448
M4     0.1496 -0.0554 -0.0208  0.0399  0.9655     NA 0.0472
SCORE      NA      NA      NA      NA      NA     NA     NA
I      0.1333  0.0668  0.0752  0.0448  0.0472     NA 1.3843
```

Listing 17: R code

Εδώ δυστυχώς έχουμε τη μεταβλητή SCORE που διαθέτει απροσδιόριστες τιμές NA. Άρα για τον υπολογισμό των ιδιοτιμών-ιδιοδιανυσμάτων πρέπει να απαλείψουμε τις τιμές αυτές

```
vars_to_keep <- colnames(S)[colSums(is.na(S)) <= 1]
S_clean <- S[vars_to_keep, vars_to_keep]
print(round(S_clean, 4))
```

Listing 18: R code

Από την οποία ο πίνακας έχει καθαριστεί και είναι ο

```
         P      M1      M2      M3      M4      I
P   0.1450  0.0684  0.1555  0.0888  0.1496 0.1333
M1  0.0684  1.2071 -0.1750 -0.2768 -0.0554 0.0668
M2  0.1555 -0.1750  1.0083  0.0583 -0.0208 0.0752
M3  0.0888 -0.2768  0.0583  0.6119  0.0399 0.0448
M4  0.1496 -0.0554 -0.0208  0.0399  0.9655 0.0472
I   0.1333  0.0668  0.0752  0.0448  0.0472 1.3843
```

Στη συνέχεια υπολογίζουμε τις ιδιοτιμές και τα ιδιοδιανύσματα.

```
eigens <- eigen(S_clean)
values <- eigens$values
vectors <- eigens$vectors

print(round(values, 4))
print(round(vectors, 4))
```

Listing 19: R code

Από την εκτέλεση του παραπάνω κώδικα προκύπτουν οι ιδιοτιμές

```
 1.4344 1.4101 0.9944 0.9093 0.5167 0.0573
```

Και τα αντίστοιχα ιδιοδιανύσματα

```
        [,1]    [,2]    [,3]    [,4]    [,5]    [,6]
[1,] -0.1472 -0.0275 -0.0633  0.2061 -0.1918  0.9457
[2,] -0.0276  0.8430  0.0719  0.3906 -0.3380 -0.1287
[3,] -0.2147 -0.3978  0.4775  0.7341  0.0381 -0.1652
```

```
[4,] -0.0803 -0.3237 -0.0808 -0.1400 -0.9109 -0.1815
[5,] -0.1368 -0.1126 -0.8691  0.4243  0.0951 -0.1559
[6,] -0.9520  0.1130  0.0317 -0.2579  0.0940 -0.0675
```

(δ΄) Στον πίνακα `data` εφαρμόζουμε την μέθοδο PCA. Αρχικά επιλέγουμε τις ανεξάρτητες μεταβλητές και στη συνέχεια εκτελούμε κανονικοποίηση σε αυτόν

```r
   X <- data[ , c("P", "M1", "M2", "M3", "M4", "I")]
X_scaled <- scale(X)
```

Listing 20: R code

Στη συνέχεια θα κάνουμε χρήση της μεθόδου PCA που είναι υλοποιημένη στην R και λαμβάνουμε τις ιδιοτιμές και τα αντίστοιχα ιδιοδιανύσματα. Στη συνέχεια επιλέγουμε τις ιδιοτιμές και τα αντίστοιχα ιδιοδιανύσματα με την μεγαλύτερη συνεισφορά στην συνολική διακύμανση. Εχουμε

```r
pca <- prcomp(X_scaled)
eig_vals <- pca$sdev ^2
explained_var <- eig_vals / sum(eig_vals)
cum_var <- cumsum(explained_var)
k <- which(cum_var >= 0.8)[1]

print(k)
print(cum_var)
```

Listing 21: R code

Από τον οποίο λαμβάνουμε τα αποτελέσματα

```
0.2964572 0.5139085 0.6835467 0.8365071 0.9615865 1.0000000
```

Από το αποτέλεσμα είναι φανερό ότι **οι τέσσερεις πρώτοι** κύριοι άξονες είναι αυτοί με την μεγαλύτερη επιρροή στην συνολική διακύμανση.

(ε΄) Θα κάνουμε χρήση της τεχνικής των φορτίσεων. Δηλαδή θα δούμε ποιά είναι η επίδραση της κάθε συνιστώσας πάνω σε κάθε κύρια συνιστώσα. Για κάθε μεταβλητή X_j και κάθε κύρια συνιστώσα Y_k, η συνεισφορά δίνεται από τις τιμές e_{jk} που δίνονται από τη σχέση

$$Corr(X_j, Y_k) = \frac{e_{ij} \cdot \sqrt{\lambda_k}}{\sqrt{\sigma_{jj}}}$$

Άρα ακολουθούμε τα εξής βήματα:

(α΄) Υπολογίζουμε τα ιδιοδιανύσματα-φορτίσεις και τις αντίστοιχες ιδιοτιμές.

(β΄) Υπολογίζουμε για κάθε μεταβλητή

- Πόσο φορτίζει τις πρώτες k κύριες συνιστώσες.
- Τη συνολική συνεισφορά που είναι το άθροισμα των τετραγώνων των φορτίσεων στις πρώτες k συνιστώσες.

(γ΄) Αφαιρούμε εκείνες τις μεταβλητές με πολύ χαμηλή συνολική συνεισφορά.

Αρχικά επιλέγουμε τις ανεξάρτητες στήλες και χωρίς μηδενική συνδιακύμανση. Και εκτελούμε κανονικοποίηση των δεδομενών.

```
  X <- data[ , c("P", "M1", "M2", "M3", "M4", "I")]
  X_scaled <- scale(X)
```

Listing 22: R code

Στη συνέχεια εφαρμόζουμε PCA ώστε να βρούμε του κύριους άξονες και να κάνουμε επιλογή των αξόνων που περιγράφουν πάνω από το 80% της διασποράς.

```
pca <- prcomp(X_scaled)
eig_vals <- pca$sdev ^2
explained_var <- eig_vals / sum(eig_vals)
cum_var <- cumsum(explained_var)
k <- which(cum_var >= 0.8)[1]
print(k)
```

Listing 23: R code

Σαν αποτέλεσμα προκύπτουν οι **4 πρώτοι** κύριοι άξονες. Στη συνέχεια υπολογίζουμε τις φορτίσεις των αρχικών μεταβλητών πάνω στους κύριους άξονες που επιλέξαμε και διατάσσουμε αυτές τις φορτίσεις κατά αύξουσα σειρά

```
  loadings <- pca$rotation[ , 1:k]
squared_contrib <- rowSums(loadings^2)

# Create give the order of the variable
contrib_table <- data.frame(
  Variable = rownames(loadings),
  Contribution = squared_contrib
)
print(contrib_table)

contrib_table <- contrib_table[order(contrib_table$Contribution), ]
print(contrib_table)
```

Listing 24: R code

Από τον παρακάτω κώδικα προκύπτει το εξής `frame`

```
    Variable Contribution
P          P    0.5038607
M3        M3    0.5296890
M1        M1    0.5789086
M4        M4    0.7609881
I          I    0.8033860
M2        M2    0.8231677
```

Επιλέγουμε ένα κατώφλι για να επιλέξουμε τις καλύτερες επιρροές. Και διαλέγουμε τις καλύτερες μεταβλητές

```
threshold <- 0.6

selected_variables <- contrib_table$Variable[
```

```
            contrib_table$Contribution >= threshold]

print(selected_variables)
```

Listing 25: R code

Από τον κώδικα προκύπτουν τα αποτελέσματα

```
"M4" "I"  "M2"
```

Οι οποίες είναι και οι σημαντικότερες μεταβλητές.

(ϛ΄) Η ανάλυση αντιστοιχιών είναι είναι μια πολυμεταβλητή τεχνική που χρησιμοποιείται για την ανάλυση πίνακα συχνοτήτων, δηλαδή κατηγορικών δεδομένων, με στόχο τη γραφική απεικόνιση των σχέσεων μεταξύ των γραμμών και των στηλών του πίνακα. Η μέθοδος εφαρμόζεται κυρίως σε ποιοτικά δεδομένα, σε αντίθεση με την Ανάλυση Κύριων Συνιστωσών, που αφορά ποσοτικά δεδομένα. Τα βασικά χαρακτηριστικά της είναι

- Βασίζεται στην απόσταση $\mathcal{X}^2$ μεταξύ των γραμμών και των στηλών ενός πίνακα συχνοτήτων που αφορά ποιοτικές μεταβλητές.
- Δίνει συντεταγμένες σε χαμηλότερης διάστασης χώρο, έτσι ώστε να μπορούμε να αναπαραστήσουμε τα δεδομένα σε μικρές διαστάσεις.
- Κάθε γραμμή ή στήλη του πίνακα τοποθετείται στον χώρο, επιτρέποντας την ταυτόχρονη ερμηνεία των κατηγοριών.

Οι ομοιότητες με την PCA είανι οι εξής:

- Και οι δύο κάνουν υποβιβασμό διάστασης δεδομένων.
- Έχουν την δυνατότητα να δώσουν γραφική αναπαράσταση των μεταβλητών.
- Κάνουν χρήση της έννοιας της ορθοκανονικότητας και του γραμμικού συνδυασμού των αξόνων.

Τελικό συμπέρασμα. Η μέθοδος αυτή δεν μπορεί να εφαρμοσθεί στα δεδομένα μας γιατί χειρίζεται **μόνο** κατηγορικά δεδομένα. Τα δικά μας δεδομένα είναι αριθμητικά.