Clustering

Come giocano i tennisti?

Giuseppe Pio Zito 583233

1 Introduzione

1.1 Scopo dello studio

L'associazione tennistica internazionale *ATP* ha contattato la nostra agenzia perché molto interessata ai contenuti dell'analisi svolta per selezionare i partecipanti al *Six Kings Slam*, in particolare vorrebbe approfondire la questione del suddividere i tennisti in base al loro stile di gioco tra quelli abili al servizio o in risposta. Lo scopo è quello di aggiungere tali informazioni nei profili dei tennisti disponibili nel proprio sito web, dove al momento le uniche descrizioni tecniche riguardano la mano forte (sinistra o destra) e la tipologia del rovescio (ad una o due mani). Ad esempio, dalla scheda di Jannik Sinner (`https://www.atptour.com/en/players/jannik-sinner/s0ag/overview`) vediamo nella sezione *Plays* che usa la mano destra e il rovescio a due mani.

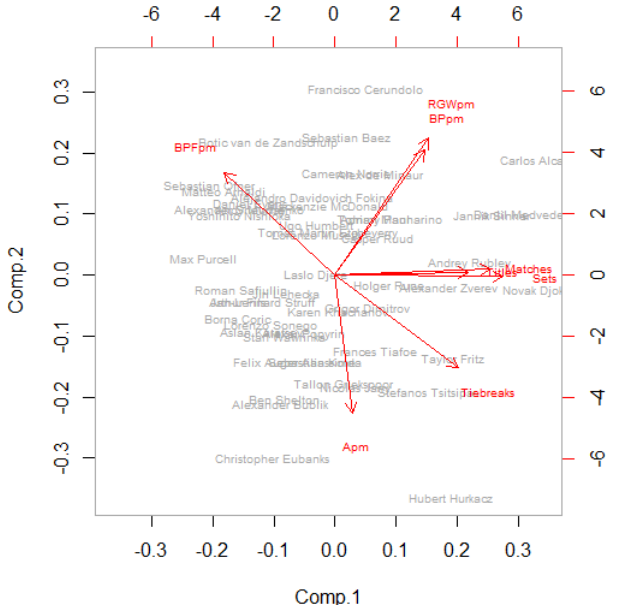
1.2 Caratteristiche della tabella

Utilizzeremo la stessa tabella usata per la prima analisi sulle prestazioni nella stagione 2023 dei tennisti nella top 50 del ranking ATP al 01/01/2024 (dati disponibili su `https://it.wikipedia.org/wiki/ATP_Tour_2023#Titoli_vinti_per_giocatore` e `https://www.ultimatetennisstatistics.com/statsLeaders`), ma considerando solamente i fattori riguardanti la prestazione e le abilità al servizio e in risposta:
- ***Matches***, partite giocate; - ***Titles***, tornei vinti; - ***Sets***, set vinti; - ***Tiebreaks***, tiebreak vinti; - ***Aces per match (Apm)***, servizi vincenti a partita; - ***Break Points Faced per match (BPFpm)***, palle break concesse a partita; - ***Return Games Won per match (RGWpm)***, game in risposta vinti a partita; - ***Break Points per match (BPpm)***, palle break a favore a partita.
Per rappresentare graficamente i cluster useremo il piano dato dalle prime due componenti principali (piano principale). Nonostante per condurre questa analisi non useremo

il fattore *UEpm* utilizzato nella prima analisi, possiamo osservare che l'intrepretazione delle prime due componenti principali resta la stessa, in cui la prima componente è un *indice di prestazione* mentre la seconda un *indice di tipologia del tennista*. In particolare, le prime due componenti rappresentano quasi l'80% della proporzione di varianza cumulata.

```
> summary(tab.pca)
Importance of components:
                          Comp.1    Comp.2     Comp.3     Comp.4     Comp.5     Comp.6      Comp.7      Comp.8
Standard deviation     1.953694 1.5543529 0.75937318 0.69510390 0.49417681 0.46076875 0.268860022 0.135815415
Proportion of Variance 0.486852 0.3081649 0.07355199 0.06162875 0.03114933 0.02708008 0.009220116 0.002352784
Cumulative Proportion  0.486852 0.7950169 0.86856894 0.93019769 0.96134702 0.98842710 0.997647216 1.000000000
> loadings(tab.pca)

Loadings:
          Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7 Comp.8
Matches    0.459         0.311  0.326  0.219  0.472  0.227  0.513
Titles     0.394        -0.212 -0.855         0.143  0.175  0.125
Sets       0.496         0.121  0.127         0.235        -0.816
Tiebreaks  0.363 -0.345 -0.190  0.210 -0.756 -0.147 -0.178  0.208
Apm               -0.511 -0.680  0.171  0.462  0.151
BPFpm     -0.327  0.380 -0.405        -0.367  0.651  0.123
RGWpm      0.277  0.508 -0.155         0.159        -0.778  0.101
BPpm       0.265  0.465 -0.396  0.250        -0.483  0.507
```



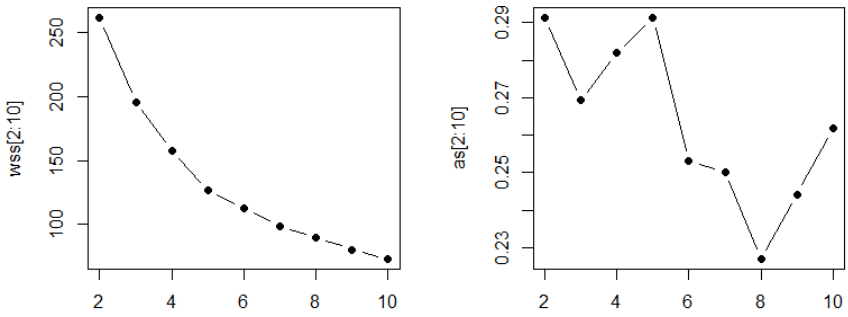
2 Scelta del metodo di clustering

Idealmente la suddivisione che vorremmo ottenere è quella in due cluster: quello dei tennisti abili nei game al servizio e quello dei tennisti abili nei game in risposta. In particolare, l'*indice di tipologia del tennista* e il piano principale (sopra) ci suggeriscono che in tal caso i due cluster dovrebbero avere una numerosità simile. Disponendo anche di fattori riguardanti le prestazioni, può risultare interessante analizzare anche suddivisioni in un numero maggiore di cluster che, ad esempio, possono dividere i tennisti eccellenti al servizio/in risposta da quelli "solo" bravi. Però, dovendo suddividere 50 tennisti, non sarebbe molto sensato avere un numero elevato di cluster, che potrebbe portare a cluster

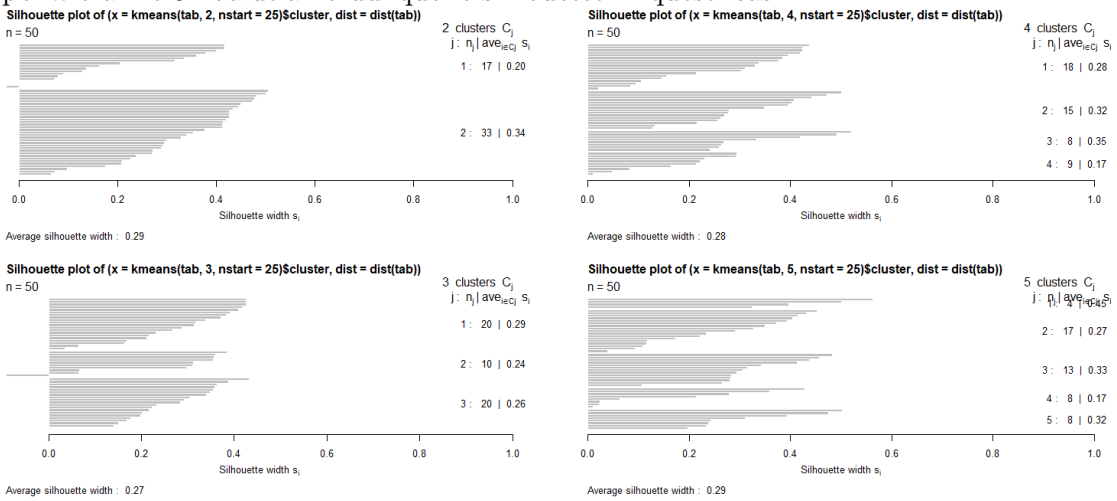
poco numerosi e quindi poco rappresentativi. Perciò, a meno di situazioni estremamente significative, escluderemo dall'analisi le suddivisioni in 6 o più cluster.
Il codice R completo utilizzato è disponibile in appendice 4.

2.1 Metodi per punti prototipo

2.1.1 k-means



Il grafico delle somme delle mutue distanze tra elementi di uno stesso cluster (sinistra) non presenta un chiaro "gomito", dunque non dà un'indicazione precisa sui valori del numero k ideale di cluster. Dal grafico della silhouette media (destra) si può osservare che non c'è una differenza significativa tra i vari valori di k, ma i valori maggiori si hanno per k tra 2 e 5. Valutiamo dunque le silhouette in questi casi.

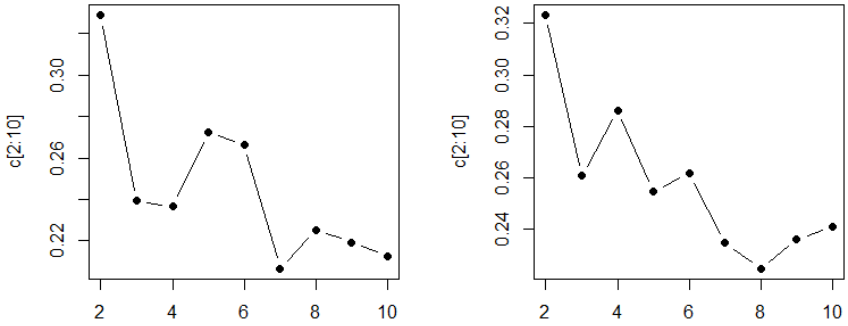


Per $k = 2$ osserviamo che i cluster non sono omogenei (un cluster ha quasi il doppio degli elementi dell'altro), c'è un valore di silhouette negativo e alcuni quasi nulli.
Per $k = 3$ c'è un valore di silhouette fortemente negativo e la silhouette media è la più bassa.
Per $k = 4$ osserviamo che i cluster sono piuttosto omogenei.
Per $k = 5$ c'è un cluster piccolo (4 elementi), ma la silhouette media è la più alta.

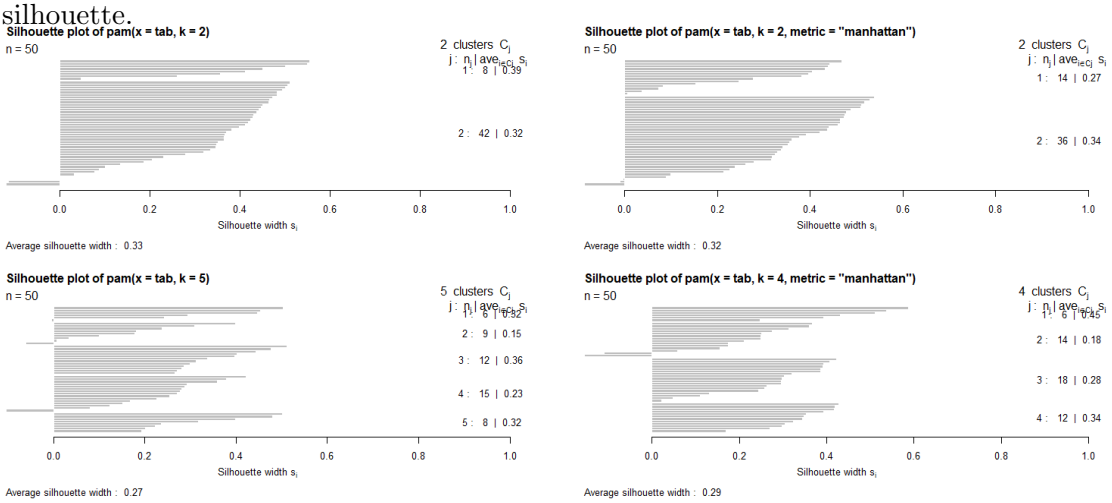
Allora teniamo in considerazione $k = 4, 5$ per l'analisi finale.

2.1.2 pam

Per questo metodo è importante anche la scelta della distanza da utilizzare. Noi useremo la distanza euclidea e la distanza L^1 (anche detta manhattan).



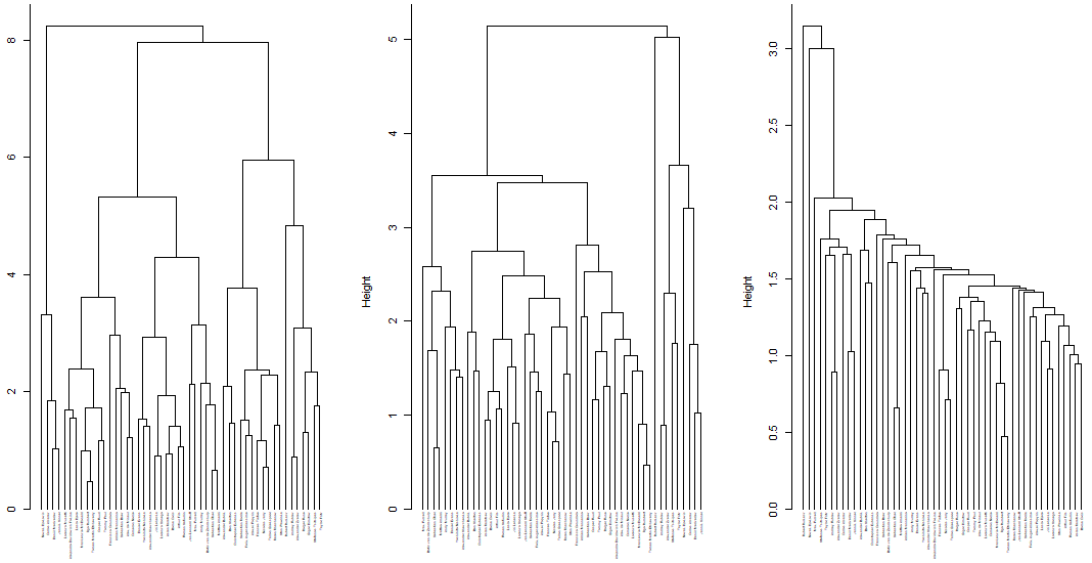
In questo caso la differenza di silhouette media è più marcata rispetto a quanto visto per k-means. Per quanto riguarda la distanza euclidea (sinistra) i valori migliori di silhouette media si hanno per $k = 2, 5$, mentre per quanto riguarda la distanza manhattan (destra) i valori migliori di silhouette media si hanno per $k = 2, 4$. Valutiamo dunque le rispettive silhouette.



Per quanto riguarda la distanza euclideea (sinistra), per $k = 2$ osserviamo che i due cluster sono fortemente sbilanciati, mentre per $k = 5$ ci sono due valori di silhouette negativi e uno quasi nullo ma i cluster sono più omogenei.
Per quanto riguarda la distanza manhattan (destra), per $k = 2$ si hanno dei cluster sbilanciati, mentre per $k = 4$ ci sono due valori di silhouette negativi ma c'è un buon valore di silhouette media.
Allora teniamo in considerazione $k = 5$ con distanza euclidea e $k = 4$ con distanza manhattan per l'analisi finale.

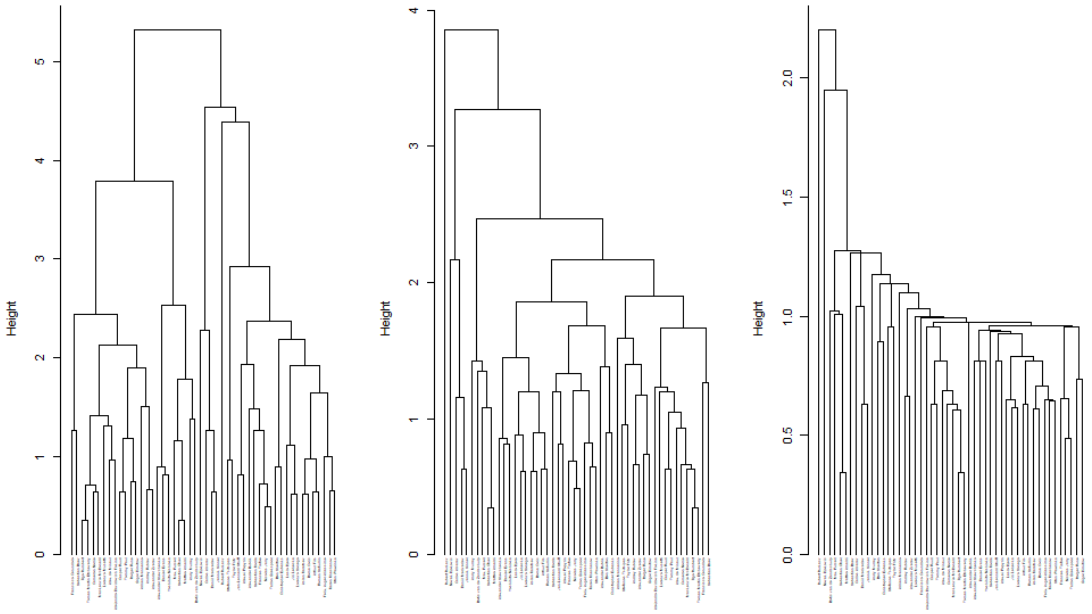
2.2 Metodi gerarchici

Iniziamo dai dendrogrammi ottenuti con la distanza euclidea.

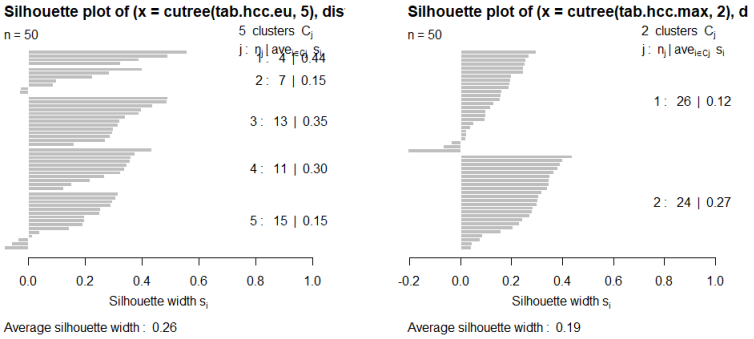


Osserviamo che per single linkage (destra) c'è sempre un cluster di un solo elemento, mentre per average linkage (centro) per $k = 2$ ci sono cluster fortemente sbilanciati e per $k \geq 3$ c'è un cluster con un singolo elemento; poichè non abbiamo mai riscontrato con i metodi precedenti la presenza di cluster con un singolo elemento, scartiamo questi metodi. Per complete linkage (sinistra) c'è sempre un cluster piccolo (4 elementi), ma questo è coerente con gli altri metodi per $k = 5$.

Alternativamente usiamo la distanza L^∞ (detta anche del massimo).



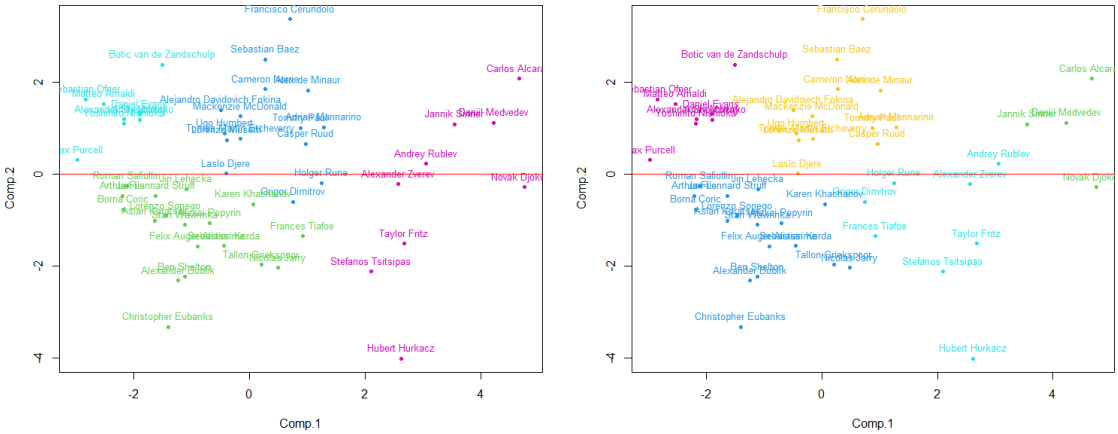
Osserviamo che per quanto riguarda average (centro) e single (destra) linkage vale quanto detto prima, invece per il complete linkage (sinistra) sembra esserci una distribuzione omogenea per $k = 2$ mentre per valori maggiori vi è sempre un cluster molto più piccolo degli altri (4 elementi per $k = 3$, un solo elemento per $k \geq 4$).



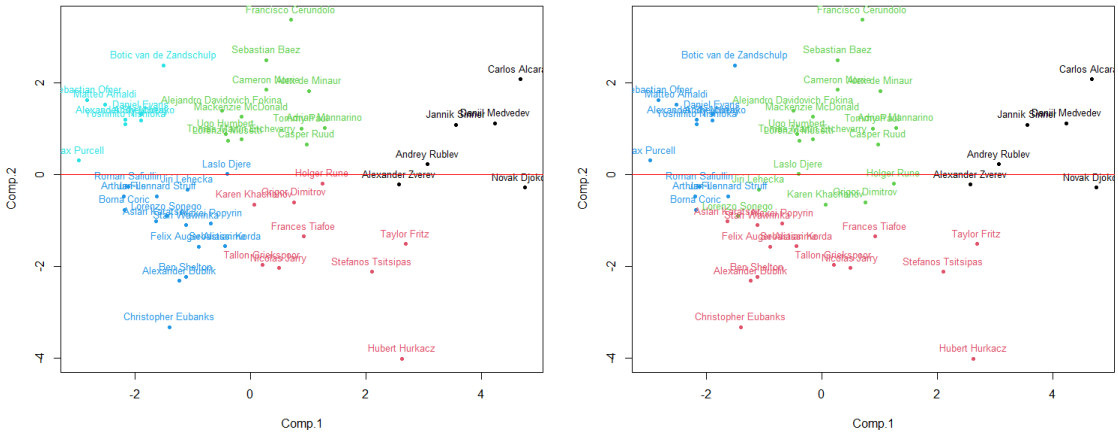
Per $k = 5$ con distanza euclidea e complete linkage (sinistra) abbiamo alcuni valori di silhouette leggermente negativi ma un buon valore di silhouette media. Per $k = 2$ con distanza del massimo e complete linkage (destra) possiamo osservare che il valore di silhouette media è molto basso e ci sono alcuni valori di silhouette fortemente negativi. Dunque teniamo in considerazione solo il primo caso per l'analisi finale.

2.3 Confronto grafico tra le suddivisioni più interessanti

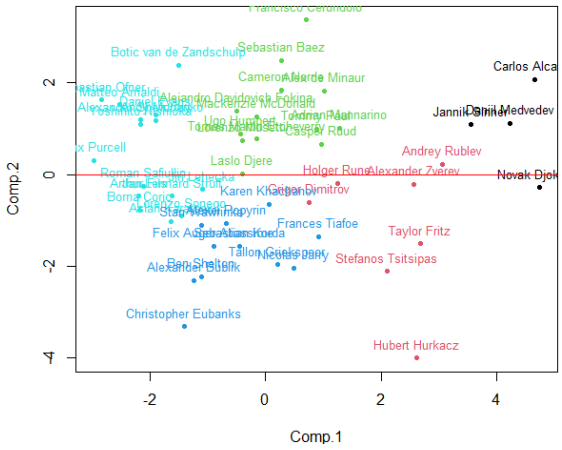
Nei grafici seguenti vedremo come si presentano i cluster scelti in precedenza nel piano principale. In particolare, ogni grafico è diviso in due parti dall'*indice di tipologia del giocatore*: nella metà superiore troviamo i giocatori abili in risposta e in quella inferiore quelli abili al servizio.



Iniziamo con i cluster ottenuti con k-means. Per $k = 4$ (sinistra) osserviamo la presenza di un cluster a destra con tennisti molto diversi per tipologia di gioco, dunque scartiamo questa suddivisione. Per $k = 5$ (destra) osserviamo che i cluster sono piuttosto coerenti con la richiesta (uniche eccezioni Andrej Rublev e Novak Djokovic) ma non troppo omogenei.



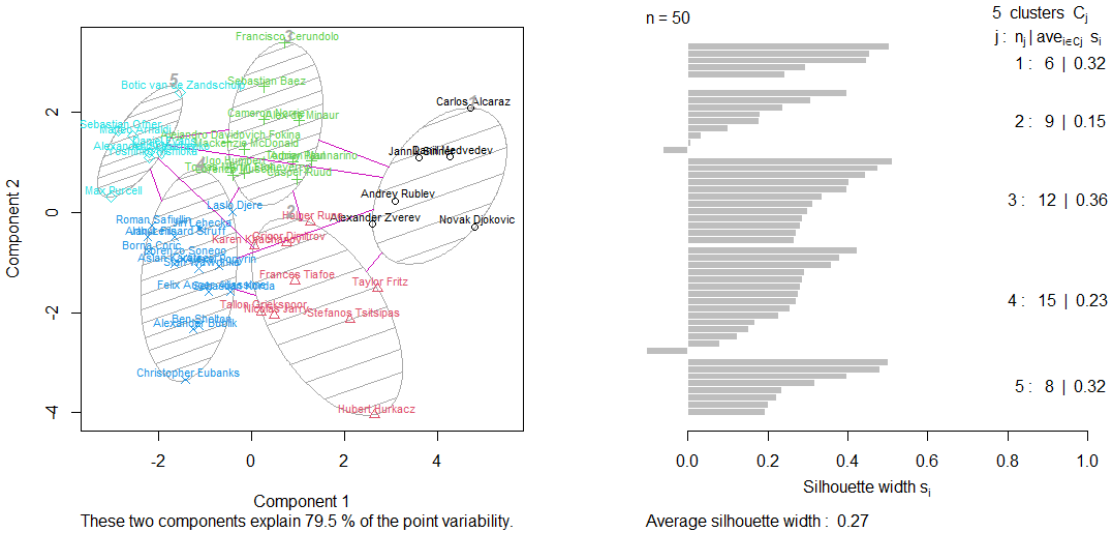
Passiamo adesso a pam. Per $k = 5$ (sinistra) osserviamo che i cluster sono piuttosto omogenei e coerenti con la richiesta ad eccezione di Laslo Djere, Alexander Zverev e Novak Djokovic. Per $k = 4$ (destra) invece non si ha una suddivisione coerente con la richiesta.



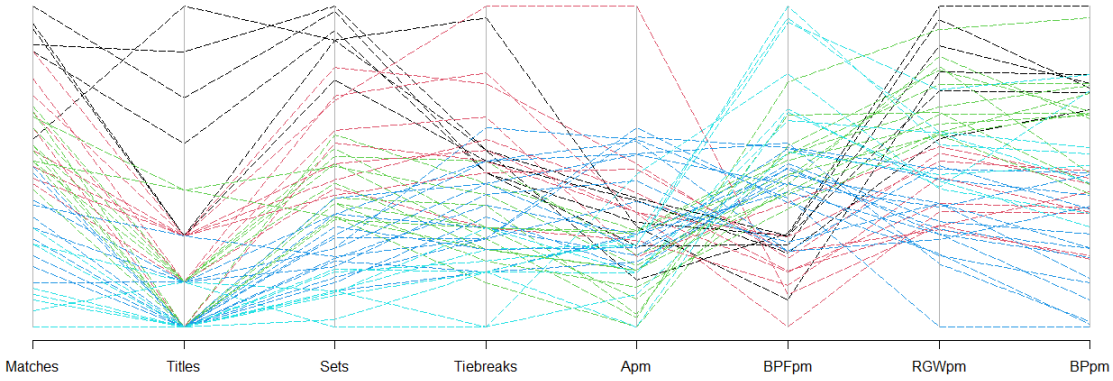
Infine, per $k = 5$ gerarchico con distanza euclidea e complete linkage non si ha una suddivisione coerente con la richiesta.
Scegliamo di proseguire con l'analisi usando la suddivisione in 5 cluster ottenuta con pam e distanza euclidea essendo il miglior compromesso tra coerenza e omogeneità.

3 Analisi e interpretazione dei cluster

Come abbiamo già osservato in precedenza, ci sono alcune osservazioni con silhouette negativa che sono: Alexander Zverev (cluster 1) rispetto al cluster 2; Karen Khachanov (cluster 2) rispetto al cluster 4; Laslo Djere (cluster 4) rispetto al cluster 3. In effetti, Zverev e Djere risulterebbero più coerenti con la richiesta se posti rispettivamente nei loro cluster più vicini, ma è anche vero che non si tratta di errrori grossolani (in senso interpretativo) essendo il loro *indice di tipologia del giocatore* molto vicino a 0.



Per dare un'interpretazione definitiva ai cluster vediamo come si comportano rispetto ai fattori d'ingresso.



Possiamo osservare che: il primo cluster corrisponde ai tennisti con prestazioni migliori, buona abilità nei game al servizio ed eccellente abilità nei game in risposta; il secondo cluster corrisponde ai tennisti con buone prestazioni ed eccellente abilità nei game al servizio; il terzo cluster corrisponde ai tennisti con buone prestazioni ed eccellente abilità nei game in risposta; il quarto cluster corrisponde ai tennisti con prestazioni sufficienti e buona abilità nei game al servizio; il quinto cluster corrisponde a tennisti con prestazioni sufficienti e buona abilità nei game in risposta.

4 Conclusioni

Alla luce dell'analisi precedente suggeriamo la seguente categorizzazione della tipologia di gioco dei tennisti: per i tennisti nel primo cluster *Elite Complete Player*; per i tennisti nel secondo cluster *Top Serve Player*; per i tennisti nel terzo cluster *Top Return Player*; per i tennisti nel quarto cluster *Serve Specialist Player*; per i tennisti nel quinto cluster *Return Specialist Player*.

Appendice

Script R

```
t=read.csv2("tabella.csv", row.names = 1)
tab=scale(t[,-9]) #standardizzo tabella
tab.pca=princomp(tab) #breve analisi delle componenti principali
summary(tab.pca)
loadings(tab.pca)
biplot(tab.pca,col=c("darkgray","red"),cex=0.65) #piano principale
    #metodi a prototipo
library(cluster)
  #kmeans
layout(t(1:2))
wss=rep(0,10) #valutazione somma mutue distanze in uno stesso cluster
for(k in 2:10){
  wss[k]=kmeans(tab,k,nstart=25)$tot.withinss
}
plot(2:10,wss[2:10],type="b",pch=16)
as=rep(0,10) #valutazione andamento silhouette kmeans
for(k in 2:10){
  cl=kmeans(tab,k,nstart=25)$cluster
  as[k]=mean(silhouette(cl,dist(tab))[,3])
}
plot(2:10,as[2:10],type="b",pch=16)
layout(matrix(c(1:4),2,2)) #confronto silhouette interessanti
plot(silhouette(kmeans(tab,2,nstart=25)$cluster,dist(tab)))
plot(silhouette(kmeans(tab,3,nstart=25)$cluster,dist(tab)))
plot(silhouette(kmeans(tab,4,nstart=25)$cluster,dist(tab)))
plot(silhouette(kmeans(tab,5,nstart=25)$cluster,dist(tab)))
#le silhouette più interessanti si hanno per k=4,5
  #pam
layout(t(1:2))
c=rep(0,10) #valutazione andamento silhouette pam euclidea
for(i in 2:10){
  c[i]=pam(tab,i)$silinfo$avg.width
}
plot(2:10,c[2:10],type="b",pch=19)
c=rep(0,10)#valutazione andamento silhouette pam manhattan
for(i in 2:10){
  c[i]=pam(tab,i,metric="manhattan")$silinfo$avg.width
}
plot(2:10,c[2:10],type="b",pch=19)
```

```
layout(matrix(1:4,2,2)) #confronto silhouette interessanti
plot(silhouette(pam(tab,2)))
plot(silhouette(pam(tab,5)))
plot(silhouette(pam(tab,2,metric="manhattan")))
plot(silhouette(pam(tab,4,metric="manhattan")))
#le silhouette più interessanti sono k=5 euclidea e k=4 manhattan
    #metodi gerarchici
  #distanza euclidea
layout(t(1:3))
tab.hcc.eu=hclust(dist(tab),"complete")
plot(tab.hcc.eu,hang=-1,cex=0.3) #sembra interessante k=5
tab.hca.eu=hclust(dist(tab),"average")
plot(tab.hca.eu,hang=-1,cex=0.3) #non interessante
tab.hcs.eu=hclust(dist(tab),"single")
plot(tab.hcs.eu,hang=-1,cex=0.3) #non interessante
  #distanza del massimo
layout(t(1:3))
tab.hcc.max=hclust(dist(tab,method="maximum"),"complete")
plot(tab.hcc.max,hang=-1,cex=0.3) #sembra interessante k=2
tab.hca.max=hclust(dist(tab,method="maximum"),"average")
plot(tab.hca.max,hang=-1,cex=0.3) #non interessante
tab.hcs.max=hclust(dist(tab,method="maximum"),"single")
plot(tab.hcs.max,hang=-1,cex=0.3) #non interessante
layout(t(1:2)) #confronto silhouette interessanti
plot(silhouette(cutree(tab.hcc.eu,5),dist(tab)))
plot(silhouette(cutree(tab.hcc.max,2),dist(tab,method="maximum")))
#l'unica silhouette interessante è k=5 con distanza euclidea e complete linkage
    #confronto grafico tra i cluster con silhouette più interessante
tab.km4=kmeans(tab,4,nstart=25)
tab.km5=kmeans(tab,5,nstart=25)
tab.pam5=pam(tab,5)
tab.pamman4=pam(tab,4,metric="manhattan")
tab.hcc.eu5=cutree(hclust(dist(tab),"complete"),5)
layout(t(1:2))
plot(tab.pca$scores,col=2+tab.km4$cluster,pch=20) #kmeans k=4
text(tab.pca$scores,labels=as.character(row.names(tab)),
     col=2+tab.km4$cluster,pos=3,cex=0.8)
abline(0,0,col="red")
plot(tab.pca$scores,col=2+tab.km5$cluster,pch=20) # kmeans k=5
text(tab.pca$scores,labels=as.character(row.names(tab)),
     col=2+tab.km5$cluster,pos=3,cex=0.8)
abline(0,0,col="red")
layout(t(1:2))
```

```
plot(tab.pca$scores,col=tab.pam5$clustering,pch=20) #pam k=5
text(tab.pca$scores,labels=as.character(row.names(tab)),
     col=tab.pam5$clustering,pos=3,cex=0.8)
abline(0,0,col="red")
plot(tab.pca$scores,col=tab.pamman4$clustering,pch=20) #pam manhattan k=4
text(tab.pca$scores,labels=as.character(row.names(tab)),
     col=tab.pamman4$clustering,pos=3,cex=0.8)
abline(0,0,col="red")
layout(1)
plot(tab.pca$scores,col=tab.hcc.eu5,pch=20) #gerarchico complete linkage k=5
text(tab.pca$scores,labels=as.character(row.names(tab)),
     col=tab.hcc.eu5,pos=3,cex=0.8)
abline(0,0,col="red")
#scegliamo pam k=5 per l'analisi finale
    #analisi
tab.pam=pam(tab,5)
layout(t(1:2))
clusplot(tab,tab.pam$clustering,stand=F,shade=T,labels=2,
         col.p=tab.pam$clustering,cex.txt=0.7,col.txt=tab.pam$clustering,
         col.clus="darkgrey")
plot(silhouette(tab.pam))
layout(1)
silhouette(tab.pam5) #chi sono i tennisti con silhouette negativa?
library(MASS)
parcoord(tab,col=as.numeric(tab.pam$cluster),lty=5)
```