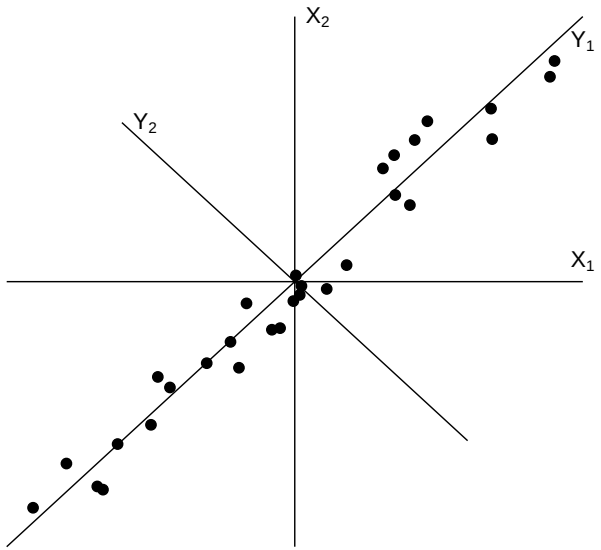
# WYKŁAD I: ANALIZA SKŁADOWYCH GŁÓWNYCH

Zaawansowane Metody Uczenia Maszynowego

# Analiza składowych głównych (PCA - principal components analysis)

$\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n \in R^p$ – dane (wartości $p$ atrybutów dla obiektów)
szukamy 'najbardziej interesujących' kierunków w danych.
'Interesujący' kierunek: ten, który najmniej zniekształca dane po zrzutowaniu na niego.
Najmniejsze zniekształcenie: suma kwadratów rzutów prostopadłych na kierunek minimalna.



Umieścimy układ współrzędnych O w środku ciężkości chmury punktów, czyli w

$$\bar{\mathbf{x}} = n^{-1} \sum_{i=1}^{n} \mathbf{x}_i$$

(odpowiada to przesunięciu danych o ich średnią $\mathbf{x}_i := \mathbf{x}_i - \bar{\mathbf{x}}$).

Szukamy jednowymiarowej reprezentacji punktów $P_i$ ($P_i'$-rzut $P_i$).

$$(OP_i)^2 = (OP_i')^2 + (P_iP_i')^2$$

$$\sum_{i=1}^{n} (OP_i)^2 = \sum_{i=1}^{n} (OP_i')^2 + \sum_{i=1}^{n} (P_iP_i')^2 \quad (*)$$

Minimalizacja $\sum_{i=1}^{n}(P_iP_i')^2$ równoważna maksymalizacji $\sum_{i=1}^{n}(OP_i')^2$.

$$\frac{1}{n-1}\sum_{i=1}^{n}(OP_i')^2$$

jest wariancją rzutów na wybraną prostą (wariancja próbkowa $= \frac{1}{n-1}\sum_{i=1}^{n}(x_i-\bar{x})^2$, ale odjęliśmy średnią, 'nowa' średnia $=0$).

Zadania:

(i) znaleźć analitycznie postać pierwszego interesującego kierunku;
(ii) znaleźć dalsze interesujące kierunki.

Oznaczenia:

$$\mathbf{x}_i{}' = (x_{i1}, x_{i2}, \ldots, x_{ip})$$

$x_{ij}$ – wartość $j$–tej cechy (atrybutu) $i$–tego obiektu.
($\mathbf{x}'$: transpozycja wektora kolumnowego $\mathbf{x}$)

Empiryczna macierz kowariancji punktów ($\mathbf{x}_i$):

$$\mathbf{S} = (s_{jk}) \quad 1 \leqslant j, k \leqslant p;$$

$$s_{jk} = \frac{1}{n-1} \sum_{i=1}^{n} (x_{ij} - \bar{x}_j)(x_{ik} - \bar{x}_k)$$

$$\bar{x}_j = \frac{1}{n} \sum_{i=1}^{n} x_{ij}$$

Mamy

$$\mathbf{S} = \frac{1}{n-1}\sum_{i=1}^{n}(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})',$$

gdzie

$$\bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i = (\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_p)'$$

**a** -wektor o długości 1. Rzut **x** na **a** ( w przestrzeni $sp(\mathbf{a})$) wynosi $\mathbf{a}'\mathbf{x}$.

Zauważmy, że dla $y_i = \mathbf{a}'\mathbf{x}_i \Longrightarrow \bar{y} = n^{-1}\sum_{i=1}^{n} y_i = \mathbf{a}'\bar{\mathbf{x}}$
$y_i - \bar{y} = \mathbf{a}'(\mathbf{x}_i - \bar{\mathbf{x}}) = (\mathbf{x}_i - \bar{\mathbf{x}})'\mathbf{a}$. Stąd
$(y_i - \bar{y})^2 = \mathbf{a}'(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})'\mathbf{a}$.

$$\frac{1}{n-1}\sum_{i=1}^{n}(y_i - \bar{y})^2 = \mathbf{a}'\Big(\frac{1}{n-1}\sum_{i=1}^{n}(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})'\Big)\mathbf{a} = \mathbf{a}'\mathbf{S}\mathbf{a}$$

Empiryczny odpowiednik faktu

$$\sigma^2_{\mathbf{a}'\mathbf{x}} = \mathbf{a}'\mathbf{\Sigma}_{\mathbf{X}}\mathbf{a}$$

(postać wariancji dla kombinacji liniowej wektora losowego).

Szukamy takiego $\mathbf{a} \in R^p$ o długości 1:

$$\mathbf{a}'\mathbf{a} = 1, \quad (**)$$

aby rzuty na kierunek $\mathbf{a}$:

$$\mathbf{a}'\mathbf{x}_1, \ldots, \mathbf{a}'\mathbf{x}_n$$

miały największą wariancję spośród wszystkich $\mathbf{a}$ spełniających $(**)$.
Dobre postawienie problemu wymaga założenia $(**)$, gdyż dla $\mathbf{b} = \lambda\mathbf{a}$

$$\text{wariancja } \{\mathbf{b}'\mathbf{x}_i\}_{i=1}^n = \lambda^2 * \text{wariancja } \{\mathbf{a}'\mathbf{x}_i\}_{i=1}^n$$

Poszukujemy maksimum $\mathbf{a}'\mathbf{S}\mathbf{a}$, przy warunku $\mathbf{a}'\mathbf{a} = 1$.
Stosujemy metodę mnożników Lagrange'a:

$$f(\lambda, \mathbf{a}) = \mathbf{a}'\mathbf{S}\mathbf{a} - \lambda(\mathbf{a}'\mathbf{a} - 1)$$

Rozwiązaniem jest

$$\mathbf{S}\mathbf{a} = \lambda\mathbf{a}$$

Wektor $\mathbf{a}$ musi być wektorem własnym macierzy $\mathbf{S}$,
$\mathbf{a}$ musi odpowiadać największej wartości własnej: $\mathbf{a}'\mathbf{S}\mathbf{a} = \mathbf{a}'\lambda\mathbf{a} = \lambda$.

## Zasada analizy składowych głównych

- Szukamy $\mathbf{a}_1$ – unormowanego wektora własnego odpowiadającego największej wartości własnej $\mathbf{S}$ równej $\lambda_1$;
- Następnie szukamy unormowanego $\mathbf{a}_2 \perp \mathbf{a}_1$ takiego, że rzuty na kierunek prostopadły do $\mathbf{a}_1$ mają największą wariancję;
  $\mathbf{a}_2 \perp \mathbf{a}_1$ – wektor własny odpowiadający $\lambda_2$,
  gdzie $\lambda_1 \geqslant \lambda_2 \cdots \geqslant \lambda_p$ oznacza wartości własne,
- $\mathbf{a}_3 : \mathbf{a}_3 \perp \mathbf{a}_1, \mathbf{a}_3 \perp \mathbf{a}_2$ – wektor własny odpowiadający $\lambda_3$,
- itd.

Otrzymujemy $\mathbf{a}_1, \ldots, \mathbf{a}_p$: wyznaczające nowy układ współrzędnych w przestrzeni $R^p$.

Metoda PCA dla $q$ pierwszych kierunków daje rozwiązanie następującego problemu optymalizacyjnego

$$\text{argmin}_{\mu, V_q, \gamma_1, \ldots, \gamma_q} \sum_{i=1}^{n} ||x_i - \mu - V_q \gamma_i||^2,$$

$V_q$- macierz rozmiaru $p \times q$.

$$V_q \gamma_i = \sum_{j=1}^{q} \gamma_{ij} \mathbf{v}_j,$$

$\mathbf{v}_j \, j = 1, \ldots q$: kolumny macierzy $V_q$.

# PCA z SVD

$$\mathbf{X}(n \times p) = \mathbf{U}\mathbf{D}\mathbf{V}',$$

$U(n \times p)$ kolumnowo ortonormalna, $D(p \times p)$-diagonalna, $V$-ortonormalna.
Założyliśmy $\bar{\mathbf{x}} = 0$.

$$\mathbf{S} = \mathbf{X}'\mathbf{X} = \mathbf{V}\mathbf{D}^2\mathbf{V}'$$

$\mathbf{v}_i$: $i$-ta kolumna macierzy $\mathbf{V}$.

$$\mathbf{S}\mathbf{v}_i = \lambda_i \mathbf{v}_i, \quad \lambda_i = d_i^2$$

$(\mathbf{V}'\mathbf{v}_i = (0, \ldots, 1 \ldots 0)')$

$j-$ta składowa główna: $y_j = \mathbf{a}_j' x, \quad x = (x_1, x_2, \ldots, x_p)$
$S_{y_j}^2 = \mathbf{a}_j' S \mathbf{a}_j = \lambda_j$

Podstawa analizy składowych głównych rozumianej jako metoda redukcji wymiaru danych:

jeśli od pewnego $i_0$, $\lambda_i$ „małe" dla $i \geqslant i_0$,
to składowe głowne dla $i \geqslant i_0$ pomijamy, jako nie wnoszące istotnej informacji o danych.

Nomenklatura.

$y_j = \mathbf{a}_j'\mathbf{x}$ $\quad$ $j-$ta składowa główna (nowa j-ta zmienna)

$\mathbf{a}_j = (a_{j1}, a_{j2}, \ldots, a_{jp})'$ – kierunek $j-$tej składowej

$a_{jk}$ – ładunek (loading) $k-$tej zmiennej dla $j-$tej składowej

$y_{ji} = \mathbf{a}_j'\mathbf{x}_i$ – wynik (score) $\mathbf{x}_i$ dla $j-$tej składowej

Problem: jak wybierać liczbę interesujących kierunków ?

Rozpatrzmy $\mathbf{a}_1, \ldots, \mathbf{a}_r$ – $r$ pierwszych kierunków głównych.

Fakt

(i) Wariancja długości rzutów wektorów $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ $(= \sum_{i=1}^{n} ||P\mathbf{x}_i||^2)$, na podprzestrzeń rozpiętą przez $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_r$ jest równa $\lambda_1 + \lambda_2 + \ldots, +\lambda_r$.

(ii) Podprzestrzeń rozpięta na pierwszych $k$ kierunkach głównych minimalizuje sumę kwadratów odległości punktów wśród podprzestrzeni $k$- wymiarowych.

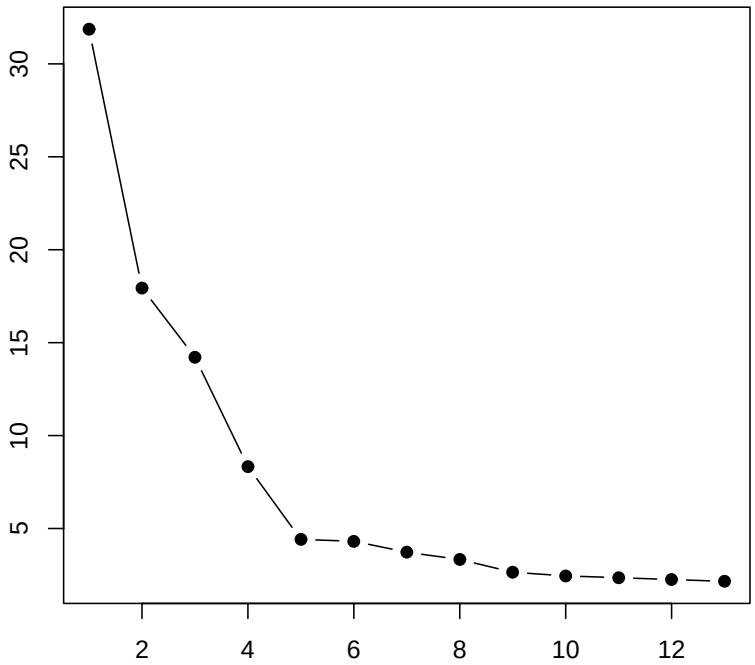
Pierwsza metoda wyboru liczby składowych $r$:

$$P_r = \frac{\lambda_1 + \lambda_2 + \cdots + \lambda_r}{\lambda_1 + \lambda_2 + \cdots + \lambda_p}$$

Szukamy najmniejszego $r$ takiego, że $P_r \geqslant a$, gdzie $a$ jest ustalonym progiem.

Druga metoda wyboru $r$:

Sporządzamy wykres $\lambda_i$ w funkcji $i$:




Płaska część wykresu odpowiada szumowi bez struktury, którego nie jesteśmy w stanie zinterpretować. Jako $r$ wybieramy minimalny indeks, po którym zaczyna się wypłaszczanie wykresu.

Uwaga. Metoda składowych głównych dla oryginalnych danych działa sensownie, gdy składowe $\mathbf{x}$ są mierzone w tych samych jednostkach i wariancje składowych są porównywalne, tzn. $\mathbf{a}'\mathbf{x}$ – interpretowalne w takich samych jednostkach.
Jeśli tak nie jest, kierunki początkowych składowych głównych będą odpowiadały współrzędnym o największej wariancji.

**Remedium: standaryzacja:**

$$x_{ij} \longrightarrow \frac{x_{ij} - \bar{x_j}}{(s_{jj})^{1/2}}$$

(zmienne po standaryzacji: średnia próbkowa 0, wariancja 1).
Metoda składowych głównych dla zmiennych standaryzowanych daje *inne* wyniki, niż dla oryginalnych zmiennych.

# Przykład analizy danych I: Hearing Loss

Dane HearingLoss zawierają pomiary 8 cech dotyczących utraty słuchu u pacjentów w wieku 39 lat.

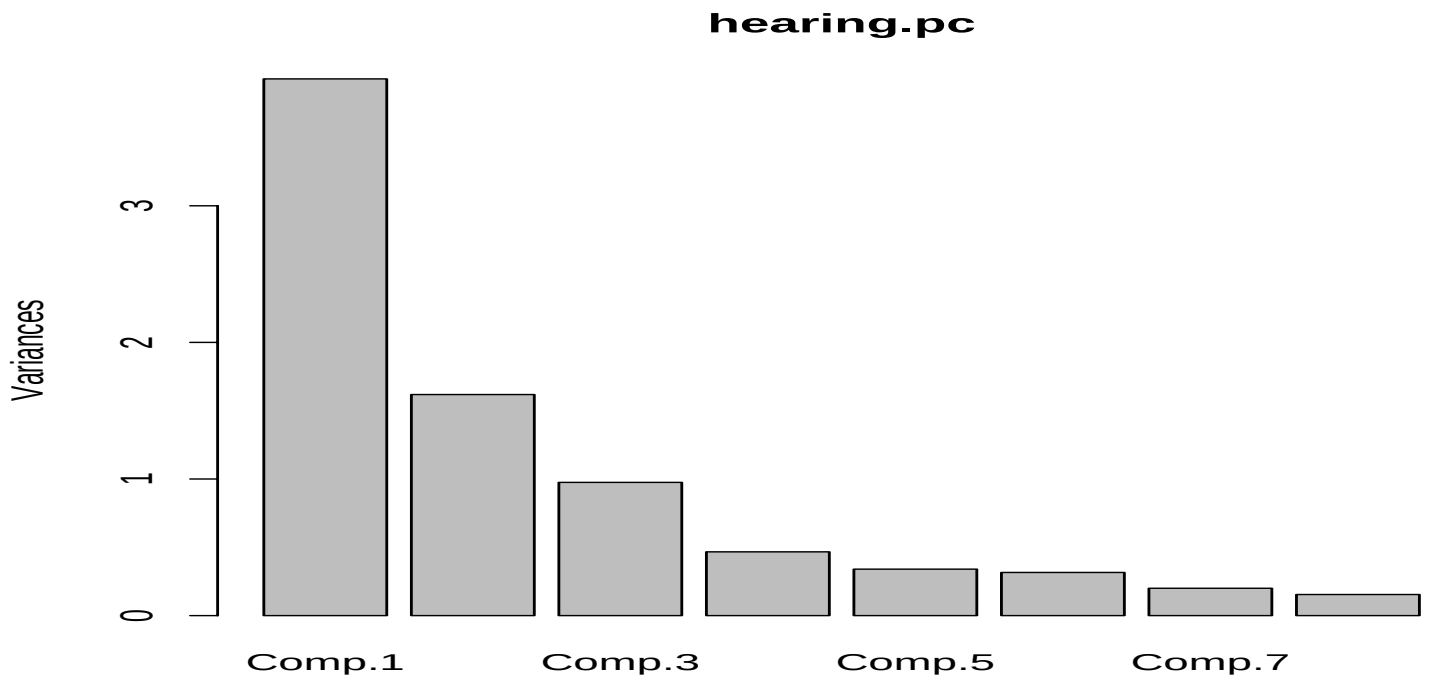
L500 - pomiar utraty słuchu dla lewego ucha dla częstotliwości 500 Hz
L1000 - pomiar utraty słuchu dla lewego ucha dla częstotliwości 1000 Hz
L2000 - pomiar utraty słuchu dla lewego ucha dla częstotliwości 2000 Hz
L4000 - pomiar utraty słuchu dla lewego ucha dla częstotliwości 4000 Hz
R500 - pomiar utraty słuchu dla prawego ucha dla częstotliwości 500 Hz
R1000 - pomiar utraty słuchu dla prawego ucha dla częstotliwości 1000 Hz
R2000 - pomiar utraty słuchu dla prawego ucha dla częstotliwości 2000 Hz
R4000 - pomiar utraty słuchu dla prawego ucha dla częstotliwosci 4000 Hz

```
Loadings:
      Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7 Comp.8
L500  -0.401  0.317 -0.158 -0.328         0.446 -0.329  0.546
L1000 -0.421  0.225        -0.482  0.379                -0.623
L2000 -0.366 -0.239  0.470 -0.282 -0.439         0.526  0.186
L4000 -0.281 -0.474 -0.430 -0.161 -0.350 -0.417 -0.427
R500  -0.343  0.386 -0.259  0.488 -0.498  0.195  0.159 -0.343
R1000 -0.411  0.232         0.372  0.351 -0.614         0.361
R2000 -0.312 -0.317  0.563  0.391  0.111  0.265 -0.478 -0.147
R4000 -0.254 -0.514 -0.426  0.159  0.396  0.366  0.414
```

Pierwsza składowa odpowiada średniej ważonej (z ujemnymi współczynnikami) wszystkich cech (sytuacja typowa, podobna do przykładu teoretycznego): ogólna utrata słuchu.
Druga składowa: przybliżony kontrast między niskimi i wysokimi częstotliwościami.
Czwarta składowa: przybliżony kontrast między lewym a prawym uchem.
( $\mathbf{b}'\mathbf{x}$ jest kontrastem jeśli $\sum_{i=1}^{p} b_i = 0$).

Skumulowane proporcje wyjaśnionej wariancji $P_r$ wynoszą

```
                          Comp.1    Comp.2    Comp.3     Comp.4     Comp.5
Standard deviation     1.9821719 1.2721328 0.9875853 0.68321459 0.58317235
Proportion of Variance 0.4911257 0.2022902 0.1219156 0.05834777 0.04251125
Cumulative Proportion  0.4911257 0.6934159 0.8153315 0.87367925 0.91619050
                          Comp.6     Comp.7     Comp.8
Standard deviation     0.5620420 0.44733783 0.39303135
Proportion of Variance 0.0394864 0.02501389 0.01930921
Cumulative Proportion  0.9556769 0.98069079 1.00000000
```


hearing.pc
Variances
0
1
2
3
Comp.1
Comp.3
Comp.5
Comp.7

# Przykład analizy danych II: dane miasta

Dane `miasta.txt` zawierają wartości trzech atrybutów dla 46 miast na świecie:
`Work` - ważona średnia liczby godzin pracy dla 12 zawodów,
`Price` - indeks kosztów utrzymania na podstawie cen 112 towarów i usług (wartość indeksu dla Zurichu równa się 100),
`Salary` - indeks płacy za godzinę w 12 zawodach po odjęciu podatku (wartość indeksu dla Zurichu równa się 100).
Macierz danych ma wymiar $46 \times 3$.

Wczytujemy dane i sprawdzamy ich postać:

```
> miasta <- read.table("Miasta.txt", header=TRUE)
```

```
> print(miasta)
```

```
             Work Price Salary
Amsterdam    1714  65.6   49.0
.............................
Toronto      1888  70.2   58.2
Vienna       1780  78.0   51.3
Zurich       1868 100.0  100.0
```

Ponieważ zmienne liczone są w różnych jednostkach, przeprowadzamy analizę składowych głównych (funkcja princomp) dla danych standaryzowanych (opcja cor=TRUE).

```
> miasta.pc <- princomp(~., cor=TRUE, data=miasta)

> print(summary(miasta.pc))

Importance of components:
                          Comp.1    Comp.2    Comp.3
Standard deviation     1.4536997 0.7935060 0.4380504
Proportion of Variance 0.7200679 0.2145480 0.0653841
Cumulative Proportion  0.7200679 0.9346159 1.0000000
```

Dwie pierwsze składowe tłumaczą 93.45 procent zmienności danych, tak więc w tym przypadku reprezentację dwuwymiarową można uznać za adekwatną.

Instrukcja

```
> plot(miasta.pc)
```

przedstawia graficznie wariancje składowych głównych.

graphics[height=2.5in,width=2.5in]fig/miastaPCA.pdf

Anliza struktury kierunków głównych (obiekt `miasta.pc$loadings`)

```
> miasta.pc$loadings

Loadings:
        Comp.1 Comp.2 Comp.3
Work     0.485  0.875
Price   -0.618  0.348 -0.705
Salary -0.619  0.338  0.709
```

Druga składowa jest średnią ważoną (z dodatnimi współczynnnikami) poszczególnych standaryzowanych cech.

Pierwsza jest w przybliżeniu różnicą `Work` razy współczynnik 0.485 i sumy `Salary` i `Price` razy współczynnik 0,62.

Trzecia jest w przybliżeniu tzw. kontrastem `Price` i `Salary`, czyli kombinacją liniową wartości cech o sumie wspólczynników równej 0.

Wartości składowych głownych dla naszych danych (obiekt `miasta.pc$scores`).

```
> miasta.pc$scores
```

| | Comp.1 | Comp.2 | Comp.3 |
|---|---|---|---|
| Amsterdam | -0.57401869 | -0.785036139 | 0.417490891 |
| . . . . . | | | |
| Luxembourg | -1.14157979 | -0.115818244 | 0.875981266 |
| . . . . . | | | |
| Manila | 2.86874742 | 0.983309709 | -0.011046941 |
| . . . . . | | | |
| Stockholm | -1.40511862 | 0.292341796 | -1.386304367 |
| . . . . . | | | |
| Vienna | -0.80877465 | -0.214758066 | 0.073180834 |
| Zurich | -2.43515004 | 1.265099775 | 0.752982364 |

Pierwsza kolumna zawiera wartości pierwszej składowej, to jest $\mathbf{a}_1'\mathbf{x}_i, i = 1, \ldots, 46$, druga - wartości drugiej składowej itd.

Znajdźmy miasto o największej wartości pierwszej składowej. Funkcja `which.max` zwraca wartość indeksu, dla którego przyjmowana jest maksymalna wartość wektora.

```
> which.max(miasta.pc$scores[,1])
Manila
25
```

Jest nim Manila.

Analiza standaryzowanych wartości

```
           Work     Price    Salary
Manila 2.226002 -1.407254 -1.435741
```

wyjaśnia dlaczego tak jest: wartość `Work` (po standaryzacji !) jest duża dodatnia, natomiast dwóch pozostałych duża ujemna, co daje dużą wartość pierwszej składowej dla tego miasta. Oznacza to relatywnie długie godziny pracy i relatywnie niskie zarobki i koszty utrzymania w tym mieście.

Luksemburg i Sztokholm mają podobne wartości pierwszej składowej, natomiast wartość trzeciej składowej jest w pierwszym przypadku dodatnia, a drugim ujemna. Stwierdzamy biorąc pod uwagę wartości ładunków dla trzeciego kierunku, że dzieje się tak na skutek relatywnie wyższych zarobków i niższych cen w Luksemburgu niż w Sztokholmie. Tak jest w istocie (standaryzowane rekordy wynoszą):

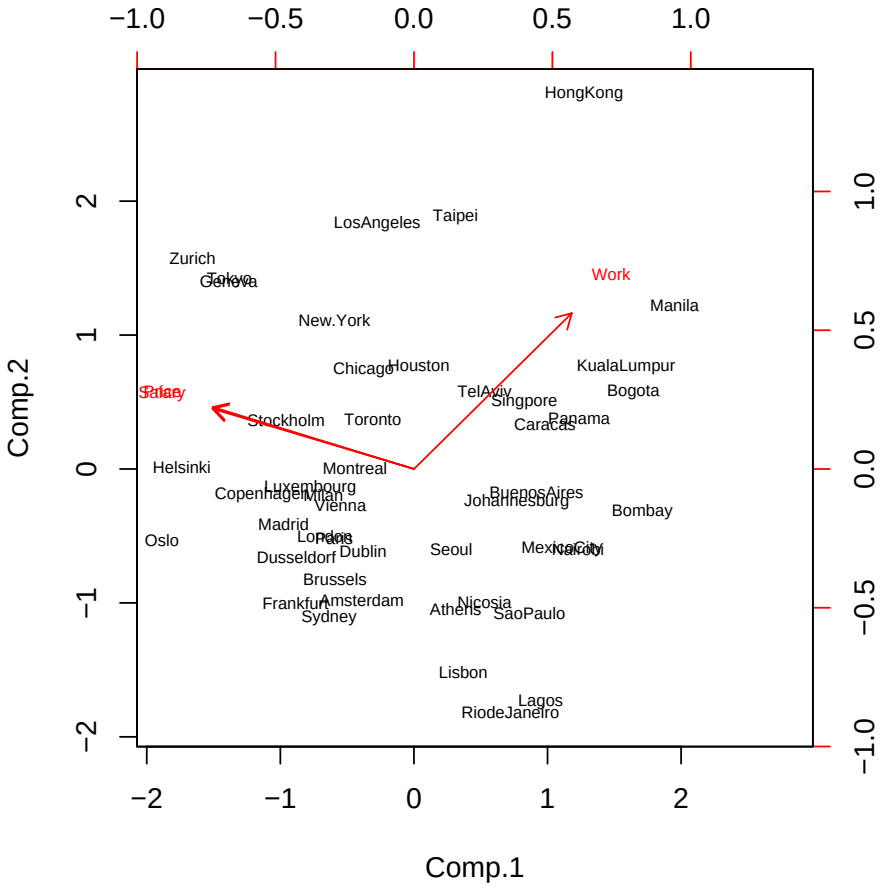
```
                Work    Price     Salary
Stockholm -0.4296888 1.926208 -0.0139614
                 Work      Price   Salary
Luxembourg -0.6419147 0.04675262 1.274526
```

# Dwuwykres (biplot)

Dwuwykres dla danych reprezentowanych przez dwie pierwsze składowe.

```
> biplot(miasta.pc, pc.biplot=TRUE)
```

Dwuwykres przedstawia $n + p = 46 + 3$ punkty odpowiadające $n$ obserwacjom i 3 zmiennym (punkty dla zmiennych reprezentowane są jako wektory). Punkty odpowiadające obserwacjom to dwie pierwsze składowe główne. Korelację między zmiennymi możemy ocenić licząc iloczyn skalarny odpowiednich wektorów. Długość rzutu wektora odpowiadającego obserwacji na wektor odpowiadający zmiennej pozwala ocenić wielkość odpowiedniej składowej obserwacji.




Widzimy, że kierunek wzrostu zmiennych `Price` i `Salary` niemal się pokrywają, co odpowiada dużej korelacji (wsp. kor. 0.8) między tymi zmiennymi. Widać również, że Manila związana jest z duża wartością zmiennej `Work`.

## Wizualizacja danych

Czy reprezentacja przy użyciu dwóch pierwszych składowych jest adekwatna ?

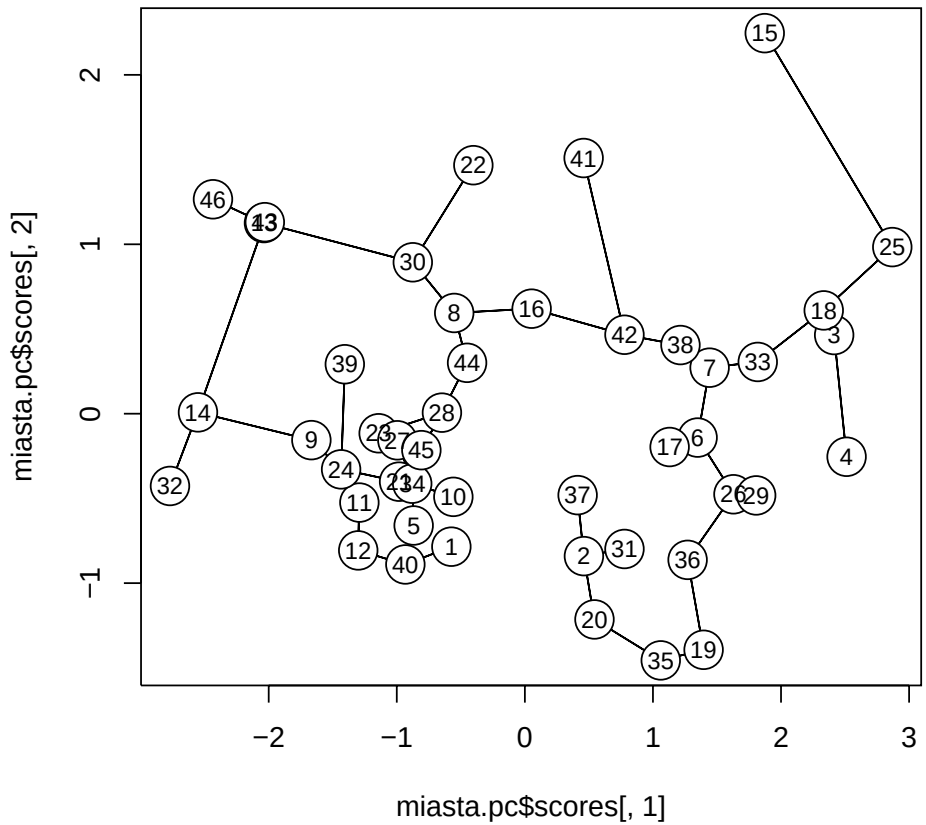
Na wykres dwóch pierwszych składowych możemy nanieść graf **minimalnego drzewa rozpinającego** (minimal spanning tree, MST).

MST dla chmury punktów w $R^p$ to taki graf, dla którego wierzchołkami są rozpatrywane punkty i który ma dwie własności:

(i) każde dwa punkty połączone są dokładnie jedną ścieżka

(ii) suma długości krawdzi (to jest odległości między połączonymi punktami) jest minimalna.

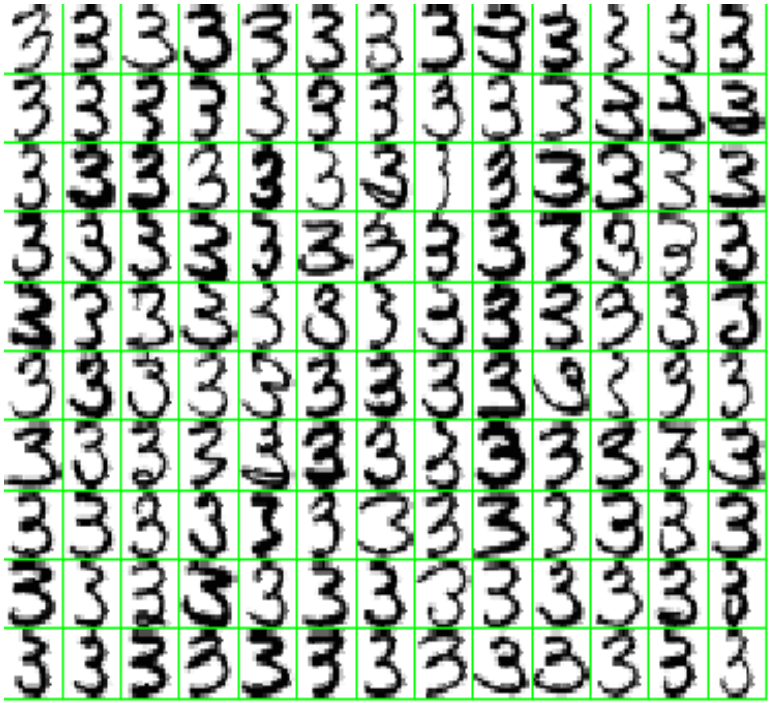
Dla adekwatnej reprezentacji w terminach dwóch pierwszych składowych, punkty połączone krawędziami winny być bliskie sobie na wykresie.




Większość miast połączonych krawędziami: blisko siebie na wykresie dwóch pierwszych składowych.
Wyjątki pary (15,25): Hongkong i Manila oraz (14,43): Helsinki i Tokio.

# Przykład analizy danych III: cyfry pisane ręcznie

130 cyfr 3 traktowanych jako wektory w $R^{256}$ ($256 = 16 \times 16$).



Wykres dwóch pierwszych składowych z naniesioną siatką kwantyli $5, 25, 50, 75, 95\%$ obu składowych.

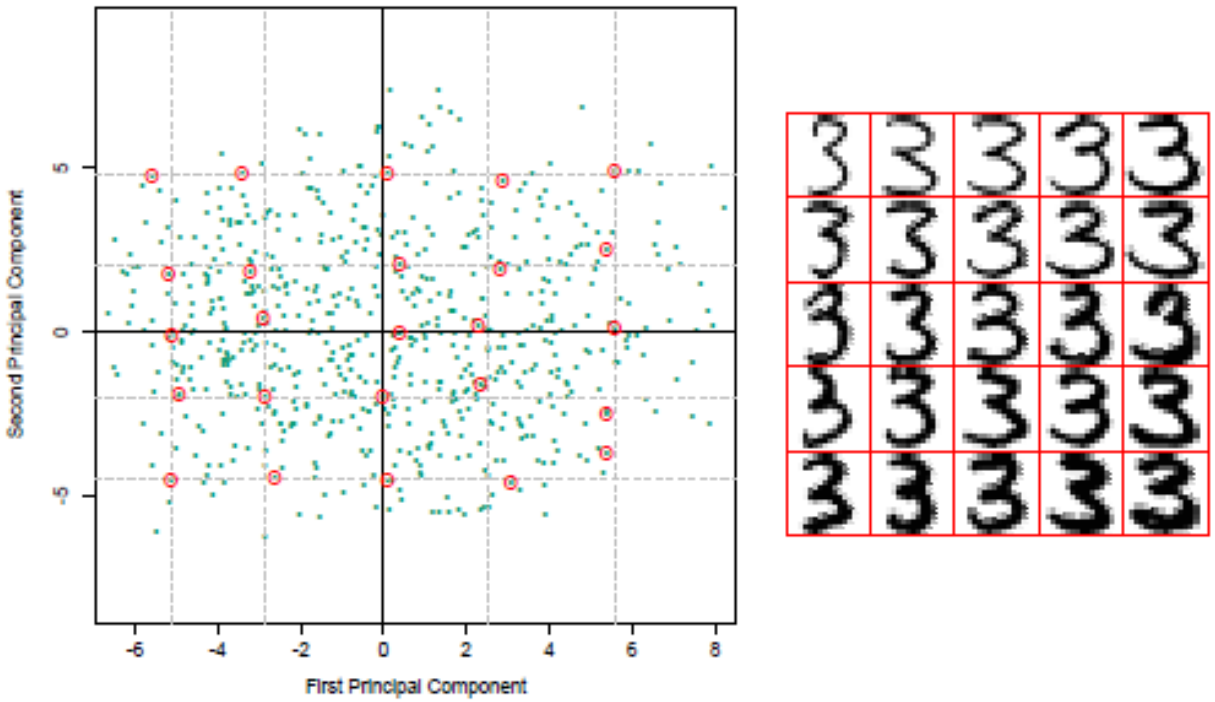



**FIGURE 14.23.** *(Left panel:) the first two principal components of the handwritten threes. The circled points are the closest projected images to the vertices of a grid, defined by the marginal quantiles of the principal components. (Right panel:) The images corresponding to the circled points. These show the nature of the first two principal components.*

# PCR - Principal Components Regression

PCA wykorzystywana do redukcji wymiaru przestrzeni predyktorów w analizie regresji: PCR - Principal Components Regression.
Problemy:

- dobór składowych głównych niezależny od ich wartości objaśniającej;
- z reguły trudna interpretowalność składowych, odróżnieniu od oryginalnych zmiennych

# PCR for `meatspec` data

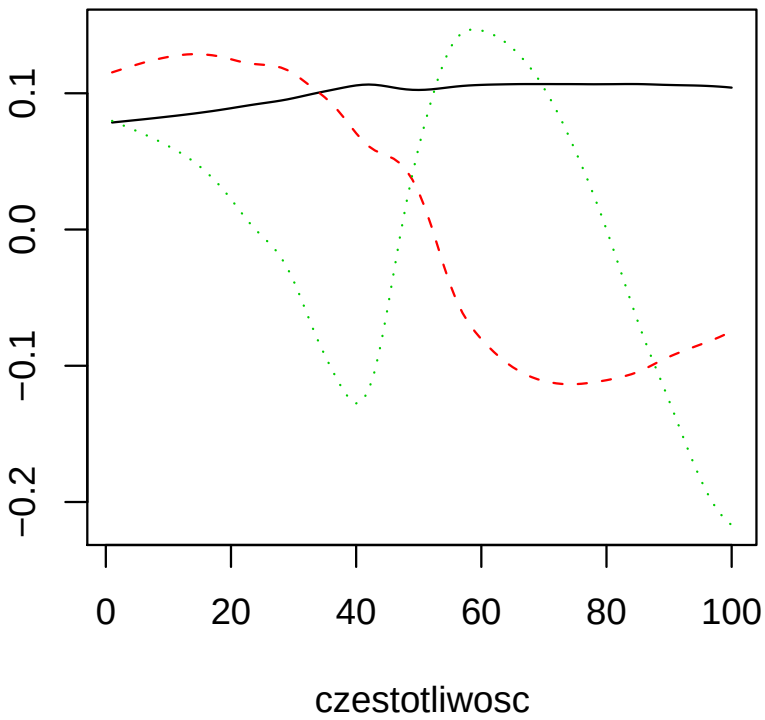
Dane `meatspec` dotyczą predykcji zawartości tłuszczu w drobno posiekanym mięsie na podstawie spektrum pochłaniania (100 zmiennych od niskich do wysokich częstotliwości). Próba 250 elementów podzielona na próbę treningową (172 elementy0 i testową (43 elementy). jakośc dopasowania mierzona przy użyciu Błędu Średniokwadratowego MSE:

$$\sqrt{\sum_{i=1}^{n}(\hat{Y}_i - Y_i)^2/n}$$

```
 model1<-lm(fat~.,meatspec[1:172,])
> summary(model$r.squared)
[1] 0.9970196
rmse <- function(x,y) sqrt(mean((x-y)^2))
> rmse(model1$fit,meatspec$fat[1:172])
[1] 0.6903167 # RMSE for training set
rmse(predict(model1,meatspec[173:215,]),meatspec$fat[173:215])
[1] 3.814000# RMSE  for  testing set
# dla danych testowych  RMSE  znacznie większe niż dla treningowych !
# eliminacja wsteczna oparta na  AIC
model2<-step(model1) #
# 28 zniennych pominiętych
> rmse(model2$fit,meatspec$fat[1:172])
[1] 0.7095069
> rmse(predict(model2,meatspec[173:215,]),meatspec$fat[173:215])
[1] 3.590245
# nieznaczne zmniejszenie  RMSE
```

```
 meatpca<-prcomp(meatspec[1:172,-101])
# prcomp - package stats
 round(meatpca$sdev,3)
  [1] 5.055 0.511 0.282 0.168 0.038 0.025 0.014 0.011 0.005 0.003 0.002
 [13] 0.001 0.001 0.001

matplot(1:100,meatpca$rot[,1:3],type="l",
xlab="czestotliwosc",ylab="")
```




as new variables. Pierwsza składowa: w przybliżeniu średnia, druga - kontrast między wysokimi i niskimi częstotliwościami. Wybieramy 4 pierwsze składowe jako nowe zmienne.

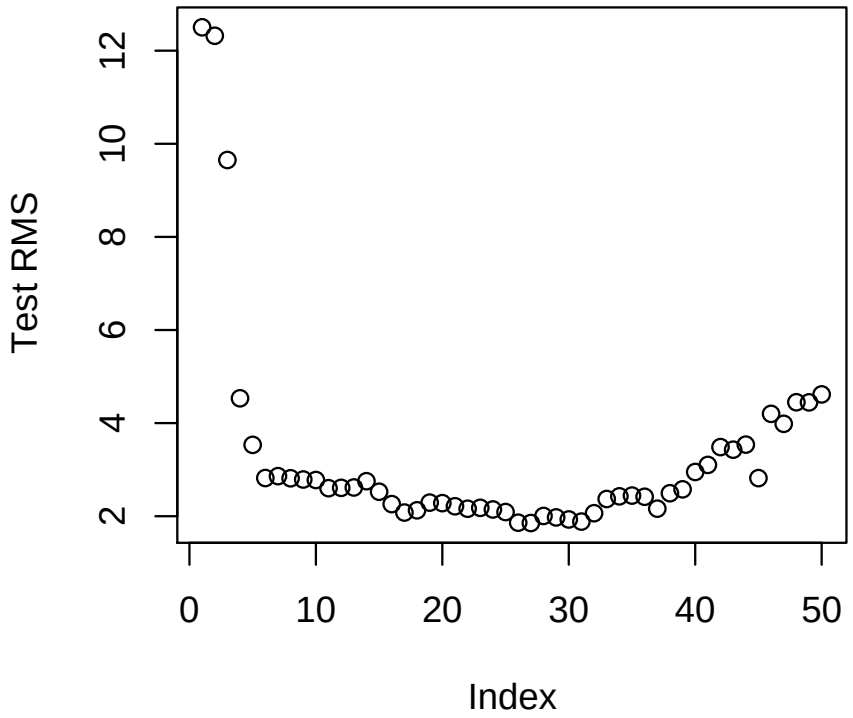
```
 model3<-lm(fat~meatpca$x[,1:4],meatspec[1:172,])
 rmse(model3$fit,meatspec$fat[1:172])
[1] 4.064745
#  RMSE duze, ale tylko 4 zmienne w modelu teraz
```

Chcemy wykonąc prognoze dla danych testowtych używając PCR wykonanej na danych treningowych. Musimy scentrować dane testowe średnimi z danych treningowych. *mogliśmy też wykonać PCA na całym zbiorze !*

```
mm = apply(meatspec[1:172, -101], 2, mean)
>
> tx = as.matrix(sweep(meatspec[173:215, -101], 2, FUN="-", mm))
> nx<-tx%*%meatpca$rot[,1:4]
# princ. components for testing set calculated
> pv<-cbind(1,nx)%*%model3$coef
> rmse(pv,meatspec$fat[173:215])
[1] 4.533982
#  złe dopasowanie. Spróbujmy  znaleźć liczbę składowych dających
# najlepszą prognozę.
```

```
rmsmeat = numeric(50)

for (i in 1:50) {
 nx <- tx %*% meatpca$rot[ ,1:i]
 model4 <- lm(fat ~ meatpca$x[ ,1:i], meatspec[1:172, ])
 pv <- cbind(1,nx) %*% model4$coef
 rmsmeat[i] <- rmse(pv, meatspec$fat[173:215])
}
 plot(rmsmeat, ylab="Test RMS")
 print(which.min(rmsmeat))
[1] 27
>
> print(min(rmsmeat))
[1] 1.854858
```


Test RMS
2
4
6
8
10
12
0
10
20
30
40
50
Index

# Partial Least Squares Regression, PLSR)

Original predictors $X_1, \ldots, X_p$ as in PCR are replaced by their linear combinations $T_1, \ldots, T_q$, but here $T_1, \ldots, T_q$ are chosen in a way corresponding to predicting $Y$ optimally.

$$\hat{Y} = \beta_1 T_1 + \cdots + \beta_q T_q.$$

Different algorithms depending on the way prediction optimality is understood, the most popular SIMPLS. Package `pls`, function `plsr`. For `meatspec` data does not yield better results than PCR.

Both methods are useful, especially for large $p$ (even for $p > n$ as lasso). If principal components can be interpreted, using PCR we may get simpler explanation of variability of $Y$. PLSR usually does not yield simpler interpretation.