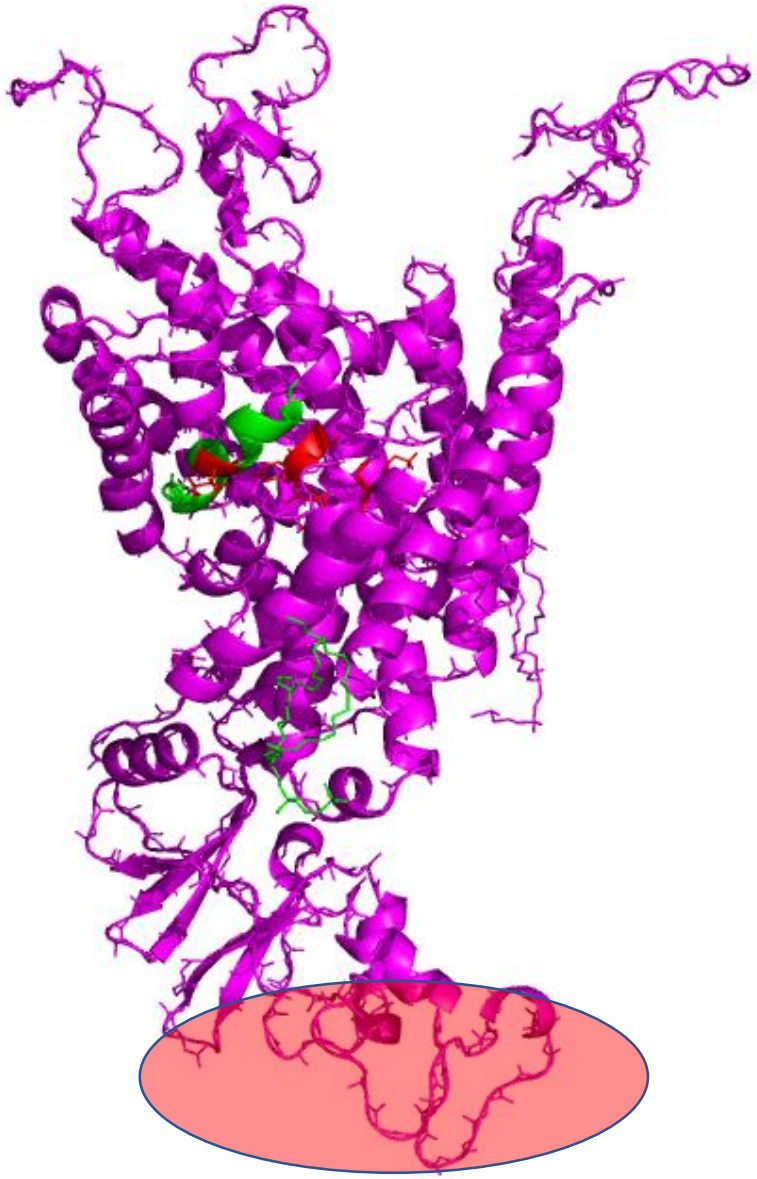
Data analysis challenge

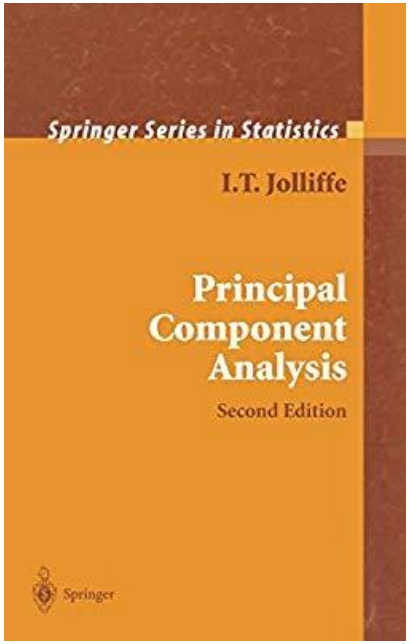
- Aim: to understand what is «wrong» with the mutated protein
- Why? We need a way of comparing movements among the different molecular simulations.
- How?
  - Because this analysis requires a lot of processing, you will have access to Power9 at BSC.
  - 3 main steps: generate simplified trajectories, perform analysis, and visualize the data.
  - Example scripts are provided for 2 analyses:
    - Principal Component Analysis (PCA) to see the most dominant motions (with most variance)
    - Autocorrelation Map to see correlated or anti-correlated motions
- Questions:
  - Is it possible to parallelize the calculations?
  - How can PCA be used to:
    - compare the movement of different chlorine and protonation states («healthy» or mutant protein)?
    - compare the movements of the «healthy» with the mutated protein?
  - Can you find a graphical way to easily compare 2 autocorrelation maps? e.g.
    - Comparison of the same chlorine state but a different protonation state for the «healthy» protein
    - Comparison of the same chlorine state but a different protonation state for the mutant protein
    - Comparison of the same chlorine and protonation state of «healthy» protein vs. mutant

Could PCA be improved in this analysis of simulations?

The loop which has larger contribution in the PCA is also located farthest from the center of coordinates (in the center of protein) and PCA analysis doesn't show other regions significantly contributing.

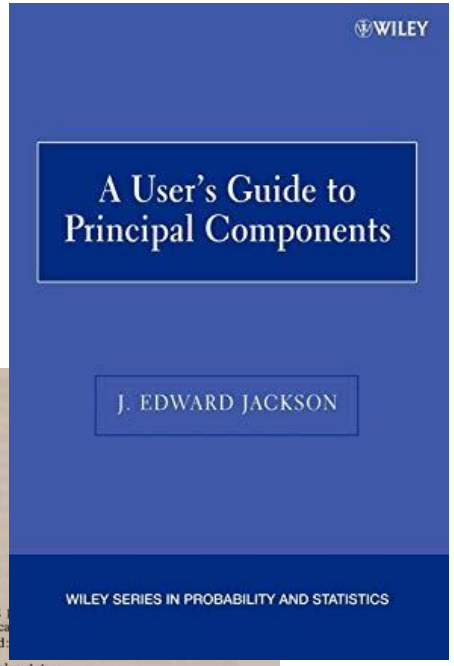


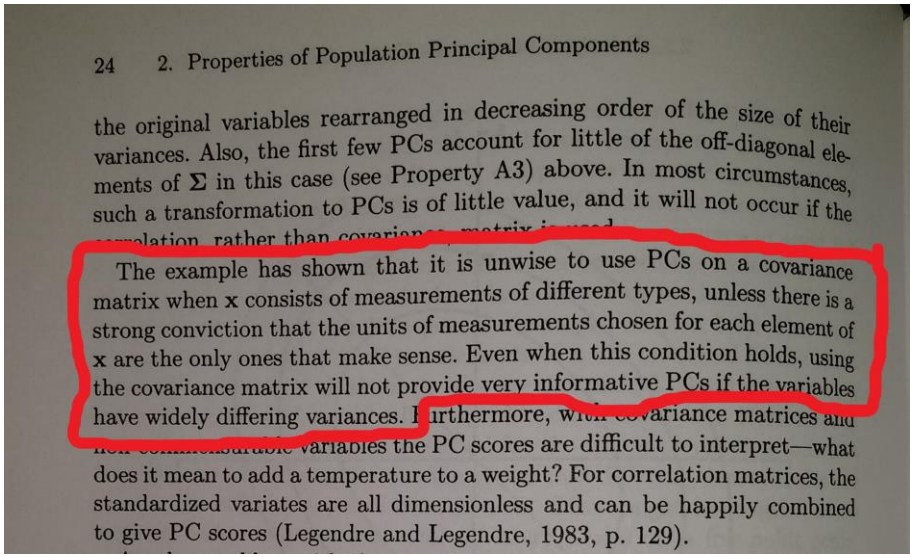
PCA textbooks recommend scaling & centering when variances are different between variables



For instance:

$$x_i' = \frac{(x_i - \bar{x})}{std.dev_i}$$

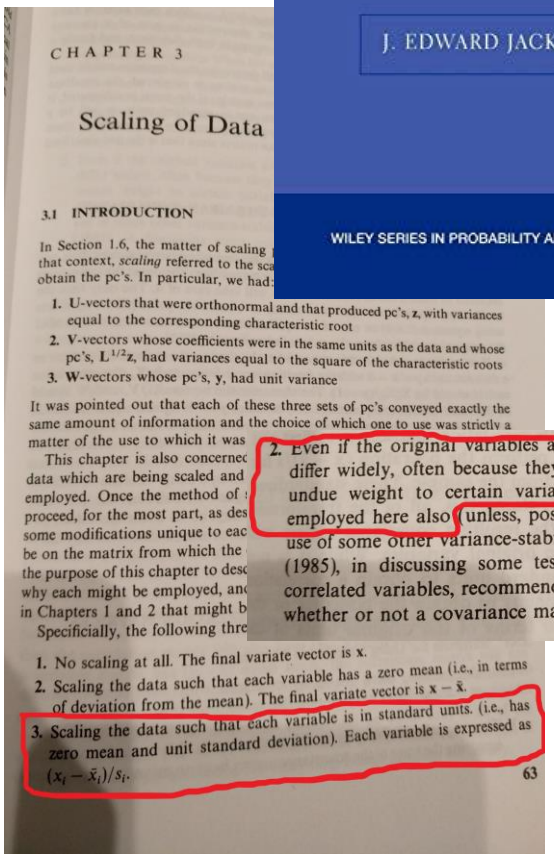




24 2. Properties of Population Principal Components

the original variables rearranged in decreasing order of the size of their variances. Also, the first few PCs account for little of the off-diagonal elements of Σ in this case (see Property A3) above. In most circumstances, such a transformation to PCs is of little value, and it will not occur if the correlation, rather than covariance, matrix is used.

The example has shown that it is unwise to use PCs on a covariance matrix when $\mathbf{x}$ consists of measurements of different types, unless there is a strong conviction that the units of measurements chosen for each element of $\mathbf{x}$ are the only ones that make sense. Even when this condition holds, using the covariance matrix will not provide very informative PCs if the variables have widely differing variances. Furthermore, with covariance matrices and non-commensurable variables the PC scores are difficult to interpret—what does it mean to add a temperature to a weight? For correlation matrices, the standardized variates are all dimensionless and can be happily combined to give PC scores (Legendre and Legendre, 1983, p. 129).



CHAPTER 3

Scaling of Data

3.1 INTRODUCTION

In Section 1.6, the matter of scaling ... that context, *scaling* referred to the sca... obtain the pc's. In particular, we had:

1. U-vectors that were orthonormal and that produced pc's, z, with variances equal to the corresponding characteristic root
2. V-vectors whose coefficients were in the same units as the data and whose pc's, $\mathbf{L}^{1/2}\mathbf{z}$, had variances equal to the square of the characteristic roots
3. W-vectors whose pc's, y, had unit variance

It was pointed out that each of these three sets of pc's conveyed exactly the same amount of information and the choice of which one to use was strictly a matter of the use to which it was ...

This chapter is also concerned ... data which are being scaled and ... employed. Once the method of ... proceed, for the most part, as des... some modifications unique to eac... be on the matrix from which the ... the purpose of this chapter to desc... why each might be employed, and ... in Chapters 1 and 2 that might b...

Specifically, the following thre...

1. No scaling at all. The final variate vector is x.
2. Scaling the data such that each variable has a zero mean (i.e., in terms of deviation from the mean). The final variate vector is $\mathbf{x} - \bar{\mathbf{x}}$.
3. Scaling the data such that each variable is in standard units. (i.e., has zero mean and unit standard deviation). Each variable is expressed as $(x_i - \bar{x}_i)/s_i$.

63

2. Even if the original variables are in the same units, the variances may differ widely, often because they are related to their means. If this gives undue weight to certain variables, the correlation matrix should be employed here also (unless, possibly, taking logs of the variables or the use of some other variance-stabilizing transformation will suffice). Harris (1985), in discussing some tests for the homogeneity of variance of correlated variables, recommended the use of these procedures to decide whether or not a covariance matrix may be employed.

When they are not centered and scaled, fewer PC dominate and mask the effect of other PCs

Dataset of cars (included by default in R package)

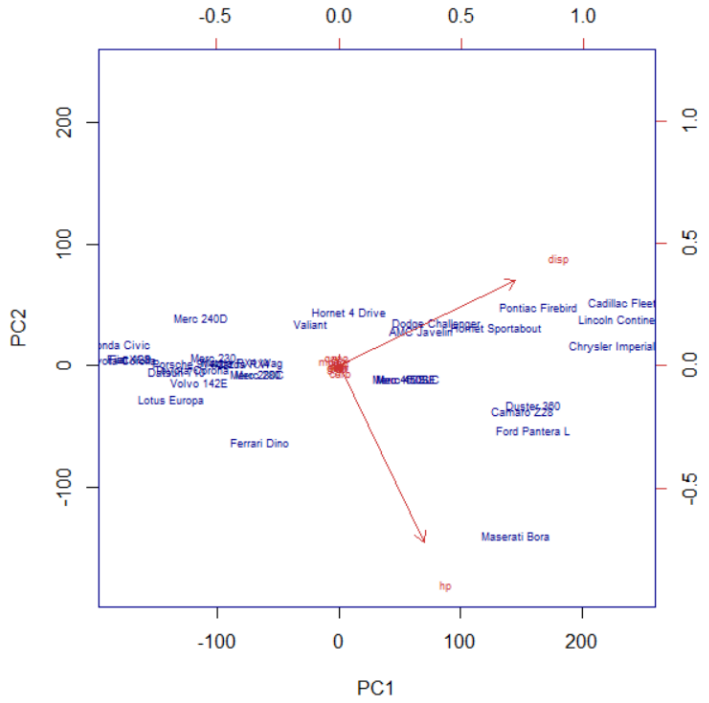
```
> summary(mtcars)
      mpg             cyl             disp             hp       
 Min.   :10.40   Min.   :4.000   Min.   : 71.1   Min.   : 52.0  
 1st Qu.:15.43   1st Qu.:4.000   1st Qu.:120.8   1st Qu.: 96.5  
 Median :19.20   Median :6.000   Median :196.3   Median :123.0  
 Mean   :20.09   Mean   :6.188   Mean   :230.7   Mean   :146.7  
 3rd Qu.:22.80   3rd Qu.:8.000   3rd Qu.:326.0   3rd Qu.:180.0  
 Max.   :33.90   Max.   :8.000   Max.   :472.0   Max.   :335.0  
      drat             wt             qsec             vs        
 Min.   :2.760   Min.   :1.513   Min.   :14.50   Min.   :0.0000  
 1st Qu.:3.080   1st Qu.:2.581   1st Qu.:16.89   1st Qu.:0.0000  
 Median :3.695   Median :3.325   Median :17.71   Median :0.0000  
 Mean   :3.597   Mean   :3.217   Mean   :17.85   Mean   :0.4375  
 3rd Qu.:3.920   3rd Qu.:3.610   3rd Qu.:18.90   3rd Qu.:1.0000  
 Max.   :4.930   Max.   :5.424   Max.   :22.90   Max.   :1.0000  
       am              gear            carb      
 Min.   :0.0000   Min.   :3.000   Min.   :1.000  
 1st Qu.:0.0000   1st Qu.:3.000   1st Qu.:2.000  
 Median :0.0000   Median :4.000   Median :2.000  
 Mean   :0.4062   Mean   :3.688   Mean   :2.812  
 3rd Qu.:1.0000   3rd Qu.:4.000   3rd Qu.:4.000  
 Max.   :1.0000   Max.   :5.000   Max.   :8.000  
```

mpg	Miles/(US) gallon
cyl	Number of cylinders
disp	Displacement (cu.in.)
hp	Gross horsepower
drat	Rear axle ratio
wt	Weight (1000 lbs)
qsec	1/4 mile time
vs	Engine (0 = V-shaped, 1 = straight)
am	Transmission (0 = automatic, 1 = manual)
gear	Number of forward gears
carb	Number of carburetors

Removed (binary/categoric data)

When they are not centered and scaled, fewer PC dominate and mask the effect of other PCs

```
> mtcarsCenteredScaled.pca <- prcomp(mtcars[,c(1:7,10,11)], center = TRUE,scale. = TRUE)
> summary(mtcarsCenteredScaled)
Error in summary(mtcarsCenteredScaled) :
  object 'mtcarsCenteredScaled' not found
> summary(mtcarsCenteredScaled.pca)
Importance of components:
                          PC1    PC2     PC3     PC4     PC5     PC6     PC7    PC8     PC9
Standard deviation     2.3782 1.4429 0.71008 0.51481 0.42797 0.35184 0.32413 0.2419 0.14896
Proportion of Variance 0.6284 0.2313 0.05602 0.02945 0.02035 0.01375 0.01167 0.0065 0.00247
Cumulative Proportion  0.6284 0.8598 0.91581 0.94525 0.96560 0.97936 0.99103 0.9975 1.00000
> mtcarsCenteredNOTScaled.pca <- prcomp(mtcars[,c(1:
> summary(mtcarsCenteredNOTScaled.pca)
Importance of components:
                           PC1      PC2     PC3
Standard deviation     136.532 38.14735 3.06642 1.27
Proportion of Variance   0.927  0.07237 0.00047 0.00
Cumulative Proportion    0.927  0.99938 0.99985 0.99
> summary(mtcarsNOTCenteredNOTScaled.pca)
Importance of components:
                           PC1      PC2      PC3
Standard deviation     310.1169 40.88309 15.83774 2.1
Proportion of Variance   0.9803  0.01704  0.00256 0.0
Cumulative Proportion    0.9803  0.99738  0.99993 0.9
```



Cpptraj can do it?

We couldn't find a way of transforming the coordinates like this:

$$x_i' = \frac{(x_i - \bar{x})}{std.dev_i}$$

Nor could find a way of calling diagmatrix in cpptraj so that it does it automatically. Neither we could find a way to do it using GROMACS

Our approach: DIY

We did export each “frame” of the simulation in text-readable format:

- 2500 PDB files (23 Mb), each with two proteins and many chlorine, sodium, lipids, Water molecules…

We removed chlorine, sodium, lipids, water, the protein “B”, and only kept Carbon-alpha of protein A.

Prepared a table where each cell’s value was centered & scaled

	x1	y1	z1	x2	y2	z2
T1						
T2						
…						
T2500						

→ PCA (using Python libraries) → Loaded into cpptraj

Our approach: DIY

step7.1a_production.dcd

individual PDB file for each frame

2500 PDB files

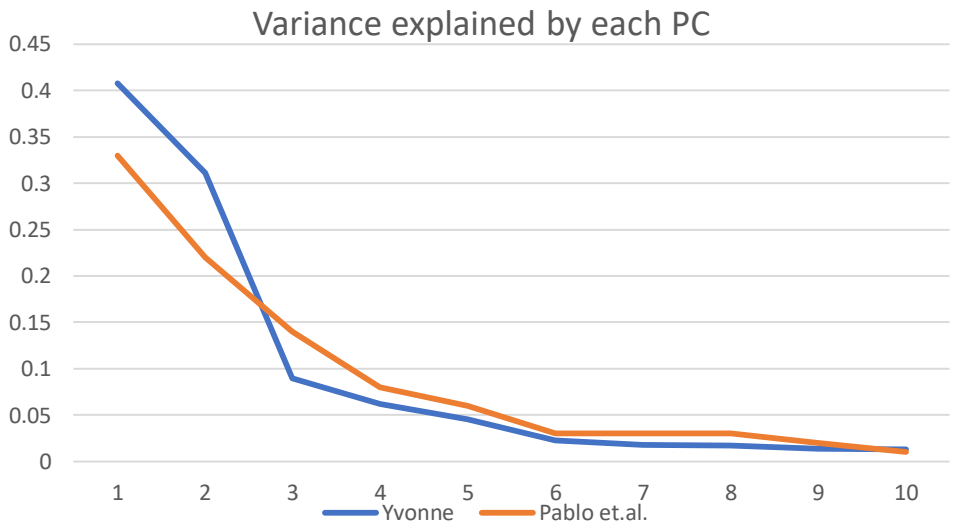
covariance matrix of C-alfa of protein “A”, using variables centered & scaled

PCA

Eigenvectors & eigenvalues

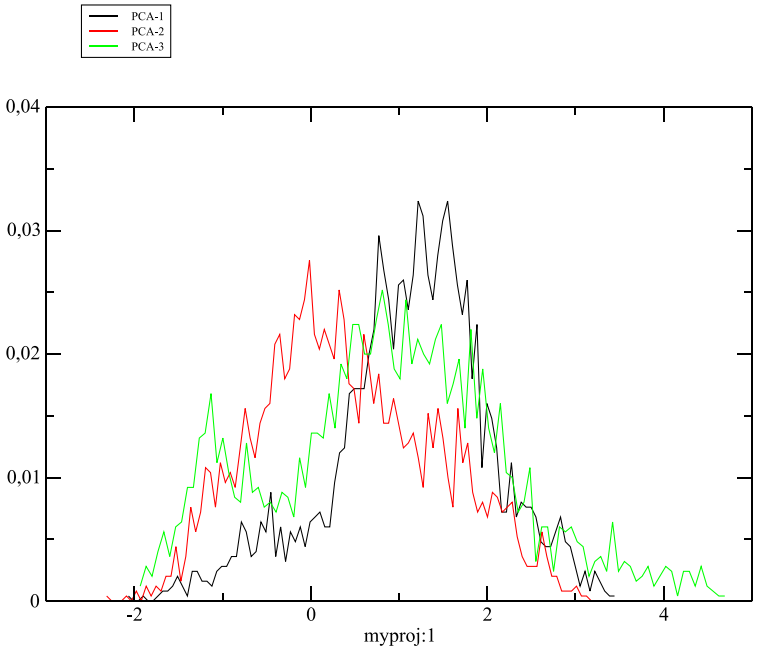
Convert eigenvector and eigenvalues into cpptraj format and load them into Cpptraj, and do the same analysis after PCA

With centering & scaling,
more principal components are needed to describe the C-alfa movements in the simulation (more information is captured by PCs)



Our results: histogram of projection using PCA eigenvectors

With our scaling



Original (without scaling)