Appendix of the article: Effect of Post-Mortem Interval on the Biochemical Quality of Blood in Wild Boar (*Sus scrofa*).

Sylvain Larrat, Benoît Rannou, Léa Bonin, Karin Lemberger, Julie Tucoulet, Ariane Payne, Clément Calenge & Anouk Decors.

Contents

1 Introduction

This vignette corresponds to the appendix of the article of Larrat et al. (2024). The aim of this paper is to explore how the blood biochemistry of a wild boar varies with time after its death, based on a sample of 20 animals. As this is a small sample and we have a lot of parameters (15 parameters), there is a risk of identifying spurious patterns through univariate analyses. We therefore opt for an exploratory analysis to identify how the overall composition of the blood changes over time. We use multivariate analysis to derive an indicator of blood composition that summarizes the maximum variation in the composition. The study of the changes over time of this indicator provides insights into the dynamics of post-mortem changes in blood biochemical composition. This can provide clues to field workers regarding the optimal timing for blood collection to allow further investigations into the cause of death without being affected by these changes.

A companion package named `QBloodWB` contains the data and functions used for this paper, and is required to reproduce the calculations in this document. The present document is also available as a vignette of this package. To install this package, first install the package `devtools` and use the function `install_github` to install `wapat`:

```
## If devtools is not yet installed, type
install.packages("devtools")

## Install the package badgertub
devtools::install_github("ClementCalenge/QBloodWB", ref="main")
```

Remark: on Windows, it is required to also install the Rtools (https://cran.r-project.org/bin/windows/Rtools/) on your computer to have a working `devtools` package (see https://www.r-project.org/nosvn/pandoc/devtools.html).

Throughout this vignette, we suppose that the reader is familiar with the analysis carried out in the main paper.

2 R code used to fit the model

We now describe the R code used in the paper.

2.1 The data

We load the package containing the code and data:

```
library(QBloodWB)
```

And then, we load the dataset:

```
data(bloodWB)
str(bloodWB)

## 'data.frame': 20 obs. of  22 variables:
##  $ Hemolysis      : int  0 1 2 2 2 2 0 2 0 1 ...
##  $ Albumin        : num  24.1 23.3 36.5 36.9 21.5 ...
##  $ AlanineAT      : num  78.1 160.6 130.7 129.2 78.7 ...
##  $ AspartateAT    : num  287.79 -0.892 -0.529 -3.089 757.26 ...
##  $ CreatineKinase : num  2171 54724 50730 96498 26889 ...
```

```
##  $ Chloride       : num  102 109 100 103 104 ...
##  $ Creatinine     : num  48 79.7 111.4 125.8 103.2 ...
##  $ Fructosamine   : num  341 258 335 365 309 ...
##  $ Fe             : num  33.3 18.9 18.8 20.7 32.3 ...
##  $ Globulins      : num  39.7 26 28.9 43.9 34.6 ...
##  $ K              : num  9.81 12.54 12.67 12.42 12.23 ...
##  $ Na             : num  141 140 136 138 141 ...
##  $ Alk.Phosphatase: num  129.4 202 41.7 100.4 106.2 ...
##  $ TotalProtein   : num  63.7 49.3 65.4 80.8 56.2 ...
##  $ UreaNitrogen   : num  4.38 4.74 5.77 3.94 3.49 ...
##  $ age            : Factor w/ 3 levels "< 1 yo","1-3 yo",..: 1 1 3 2 1 3 2 2 3 3 ...
##  $ sex            : chr  "F" "M" "M" "F" ...
##  $ TimeColl       : num  11.1 11.3 11.3 11.4 11.5 ...
##  $ Temperature    : num  35.1 35 34.1 38.8 34.8 36.5 35.9 33.6 NA NA ...
##  $ Aspect         : chr  "correct" "correct" "presence of clots" "correct" ...
##  $ Hour           : chr  "11" "11" "11" "11" ...
##  $ TimeSinceColl  : num  1 1 1 1 1 1 2 2 2 3 ...
```

The dataset `bloodWB` is a data.frame containing data on the biochemistry of the blood of 20 wild boar. The help page of this dataset describes the different variables in this data.frame.

We first extract the first 15 variables, which pertain only to the biochemistry of the the blood of the 20 animals, as well as the time elapsed since the death of the animals:

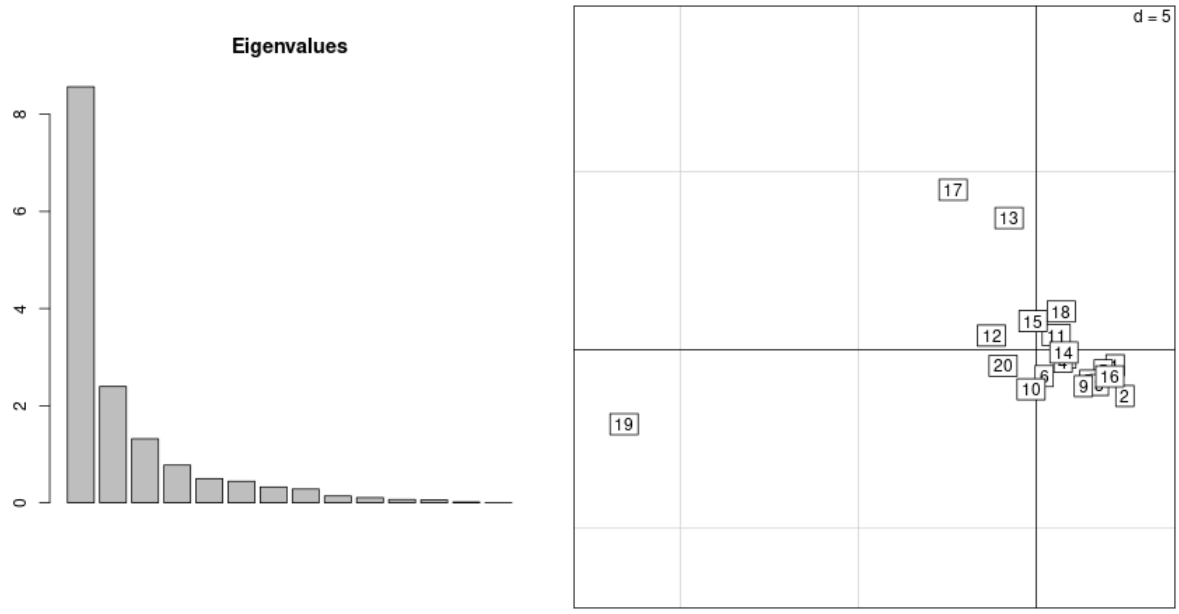
```
a <- bloodWB[,1:15]
time <- bloodWB$TimeSinceColl
```

2.2 Principal component analysis

We first carry out a principal component analysis (PCA) of the dataset. This analysis identifies linear combinations of the biochemical parameters (i.e., a sort of weighted average of these parameters), named principal components, each principal component attributing a "score" to each animal. The first principal components returned by the analysis have the property to capture the largest proportion of variance in the dataset. In other words, each principal component is a summary indicator of the blood composition that is highly correlated with the biochemical parameters of interest.

We use the function `dudi.pca` of the package **ade4** to carry out this analysis:

```
library(ade4)
pc0 <- dudi.pca(a, scannf=FALSE)
par(mfrow=c(1,2))
barplot(pc0$eig, main="Eigenvalues")
s.label(pc0$li)
```

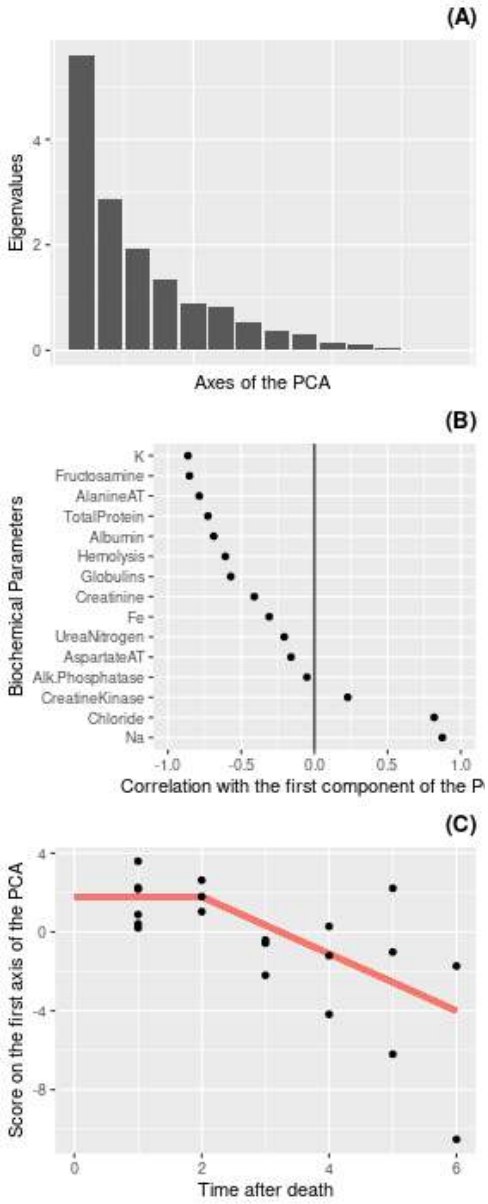


The left plot shows the screeplot of this analysis: it is proportional to the proportion of variance explained by the successive principal components of the analysis (eigenvalues). The first eigenvalue is much larger than the following ones, indicating that the first principal component summarizes most of the common structure in the dataset.

The right plot shows the scores of the wild boars on the first factorial plane of this analysis (x-axis = first principal component; y-axis = second component). Note that the wild boar number 19 strongly attracts the first principal component of the PCA. We therefore carry out again this analysis without this animal and project this animal as supplementary row in this analysis, to avoid this "leverage effect". The results corresponds to Fig. 1 of the paper:

```
## Remove WB #19
ab <- a[-19,]

## Carries out the PCA
pc <- dudi.pca(ab, scan=FALSE)

## adds WB 19 as supplementary individual, and insert it in the results:
pc$li <- rbind(pc$li[-19,],suprow(pc, a[19,])$lisup[1,1],
               pc$li[19,,drop=FALSE])

## Load the packages to allow the plot
library(ggplot2)
library(gridExtra)

## The segmented regression (threshold set at 2)
axis1 <- pc$li[,1]
re <- (time>2)*time
mod1 <- lm(axis1~re)

## First plot (screeplot)
df <- data.frame(noval=1:length(pc$eig),Eig=pc$eig)
gra <- ggplot(df)+geom_bar(aes(x=noval,y=Eig), stat="Identity")+
    xlab("Axes of the PCA")+ylab("Eigenvalues")+
    ggtitle("(A)")+
    theme(axis.text.x=element_blank(),
          axis.ticks.x=element_blank())+
    theme(plot.title = element_text(face="bold",hjust=1))
```

```
df2 <- pc$li  |> dplyr::mutate(Time=time)
df3 <- pc$co  |> tibble::rownames_to_column()  |>
dplyr::arrange(dplyr::desc(Comp1))
df3$rowname <- factor(df3$rowname, levels=df3$rowname)
grb <- ggplot(df3)+geom_point(aes(x=Comp1, y=rowname))+
    geom_vline(xintercept = 0)+xlab("Correlation with the first component of the PCA")+
    ylab("Biochemical Parameters")+xlim(-1,1)+ggtitle("(B)")+
    theme(plot.title = element_text(face="bold",hjust=1))
grc <- ggplot(df2)+geom_segment(aes(x = 0, y = coefficients(mod1)[1],
                                    xend = 2, yend = coefficients(mod1)[1]),
                                colour="#F8766D", size=1.5)+
    geom_segment(aes(x = 2, y = coefficients(mod1)[1],
                     xend = 6,
                     yend = coefficients(mod1)[1]+coefficients(mod1)[2]*6),
                 colour="#F8766D", linewidth=1.5, alpha=0.8)+
    geom_point(aes(Time, Axis1))+xlab("Time after death")+
    ylab("Score on the first axis of the PCA")+ggtitle("(C)")+
    theme(plot.title = element_text(face="bold",hjust=1))+
    NULL
grid.arrange(gra, grb, grc, ncol=1)
```



The first plot is the screeplot of this new PCA. The second plot gives the correlation of the biochemical parameters with the first principal components of this PCA. This plot allows to give a biological meaning to this summary indicator. Although caution is needed in interpreting these results (due to the small sample size), these correlations demonstrate that this indicator provides a reasonable summary of the biological process underlying post-mortem biochemical changes (see main text).

Finally, the last plot shows how the scores of the animals changes with the time elapsed since death. We added on this plot the segmented regression model with a threshold set at t = 2 hours. Note that the proportion of inertia expressed by the first axis is equal to:

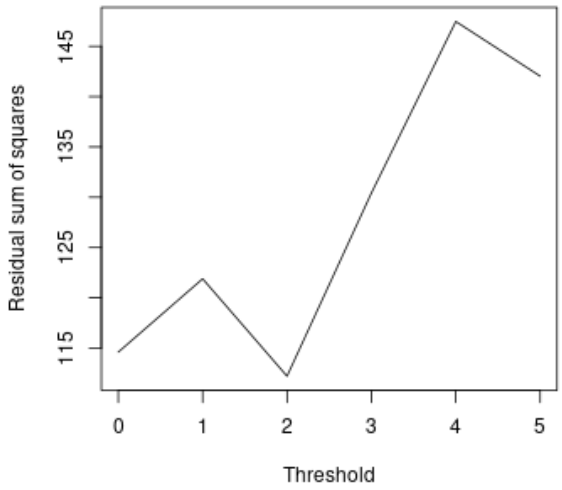
```
(su <- round(100*pc$eig[1]/sum(pc$eig)))

## [1] 37
```

37% of the inertia. The first component therefore explains an important pattern in the data.

2.3 Segmented regression

We have added the segmented regression describing how the scores of the wild boar on the first principal component of the PCA varies with time since death on the previous plot. We now demonstrate that a threshold at t = 2 hours is optimal. First, we can calculate the residual sum of squares associated to a segmented model characterized by different thresholds. We use the function `rssSRCuts` of the package for that:

```
ssr <- rssSRCuts(time, axis1, cutlim=c(0,5))
plot(c(0:5), ssr, ty="l", ylim=range(ssr),
     xlab="Threshold",
     ylab="Residual sum of squares")
```



The residual sum of square is minimal for t = 2 hours. Note that there is not a strong difference between the segmented regression with a threshold set at 2 hours and a classical linear regression:

```
## Segmented regression
segtime <- (time>2)*time
mod1 <- lm(axis1~segtime)
## Classical regression
mod2 <- lm(axis1~time)

summary(mod1)


##
## Call:
## lm(formula = axis1 ~ segtime)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.5216 -1.1609  0.4203  1.1026  5.2555
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.7901     0.8049   2.224  0.03919
## segtime      -0.9651     0.2416  -3.994  0.00085
##
```

```
## Residual standard error: 2.497 on 18 degrees of freedom
## Multiple R-squared:  0.4699,Adjusted R-squared:  0.4405
## F-statistic: 15.96 on 1 and 18 DF,  p-value: 0.0008502
```

```
summary(mod2)
```

```
##
## Call:
## lm(formula = axis1 ~ time)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.1798 -1.6224  0.1924  1.6519  5.2903
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.2900     1.1284   2.916  0.00923
## time         -1.2720     0.3258  -3.905  0.00104
##
## Residual standard error: 2.523 on 18 degrees of freedom
## Multiple R-squared:  0.4586,Adjusted R-squared:  0.4285
## F-statistic: 15.25 on 1 and 18 DF,  p-value: 0.001038
```

As indicated in the paper, the R-squared is equal to 0.47 for the segmented regression and 0.46 for the classical regression. The use of the function `anova` allows the calculation of the residual sum of squares (RSS) for the two model)

```
anova(mod2,mod1)
```

```
## Analysis of Variance Table
##
## Model 1: axis1 ~ time
## Model 2: axis1 ~ segtime
##   Res.Df    RSS Df Sum of Sq F Pr(>F)
## 1     18 114.61
## 2     18 112.22  0    2.3893
```

Finally, we use the boostrap approach described in the main paper and estimate the proportion of the boostrap samples for which a threshold of t = 2 hours is optimal. This boostrap approach is implemented in the function `bootSR` of the package (see the help page of this function. We

```
set.seed(777) ## for reproducibility

bootSR(time, axis1, cutlim=c(1,4), nBoot=1000)
```

```
##
##     1     2     3     4
## 0.071 0.834 0.018 0.077
```

A threshold set at two hours is optimal for 83% of the bootstrap samples.

References

Larrat, S. et al. 2024. Effect of Post-Mortem Interval on the Biochemical Quality of Blood in Wild Boars (Sus scrofa). – in prep.