

Supplement

Root-restricting layers in German agricultural soils

Part I: Extent and cause

Florian Schneider and Axel Don

This is a commented R-Script for calculating the area extent of compacted cropland.

```
rm(list=ls())
### Set working directory
# setwd("x/y/z")

library(tidyverse)
library(grid)
library(pBrackets)
```

Function to draw curly brackets (Source: <https://stackoverflow.com/a/35662327>):

```
bracketsGrob = function(...){
  l = list(...)
  e = new.env()
  e$l = l
  grid::recordGrob( {
    do.call(grid.brackets, l)
  }, e)
}
```

Load data

- *site*: Unique identifier for each sampling site
- *depth*: depth increment
- *use*: landuse (C=crop; G=grass)
- *PD0*: Measured packing density in g cm^{-3}
- *PD1*: Predicted packing density in g cm^{-3} without *man*
- *PD2*: Predicted packing density in g cm^{-3} with neither *man* nor *use*

```
data = read.delim("data.txt", header=T, sep="\t")
head(data,n=10)
```

##	site	depth	use	PD0	PD1	PD2
## 1	1	0-10cm	C	1.390200	1.416844	1.217591
## 2	1	10-30cm	C	1.762200	1.590747	1.574748
## 3	1	30-50cm	C	1.817500	1.702430	1.700176
## 4	1	50-70cm	C	1.848200	1.747727	1.747727
## 5	1	70-100cm	C	1.719000	1.783904	1.783904
## 6	2	0-10cm	C	1.347200	1.459575	1.242071
## 7	2	10-30cm	C	1.593400	1.604110	1.578901
## 8	2	30-50cm	C	1.719200	1.658199	1.655449
## 9	2	50-70cm	C	1.600200	1.660869	1.660869
## 10	2	70-100cm	C	1.741217	1.585096	1.585096

Reshape the data frame and calculate model residuals:

```
data.mod = as.data.frame(data %>%
  gather("model", "value", PD1:PD2) %>%
  unique() %>%
  mutate(model=ifelse(use=="G", "ref", as.character(model)),
    error=PD0-value))
```

Significant differences

Check for significant differences between model residuals (reference, PD1, PD2) per depth increment:

```
temp = as.data.frame(data.mod %>%
  group_by(depth, model) %>%
  summarise(n = sum(!is.na(error)),
    mean = round(mean(error, na.rm=T), 2),
    whisker = boxplot.stats(error)$stats[5],
    sd = round(sd(error, na.rm=TRUE), 2),
    se = round(sd / sqrt(n), 3)))

test=unique(as.character(temp[,c("depth")]))
temp2=data.frame(model=NA, test=NA, depth=NA)

for(i in 1:length(test)){
  layer = as.character(test[i])

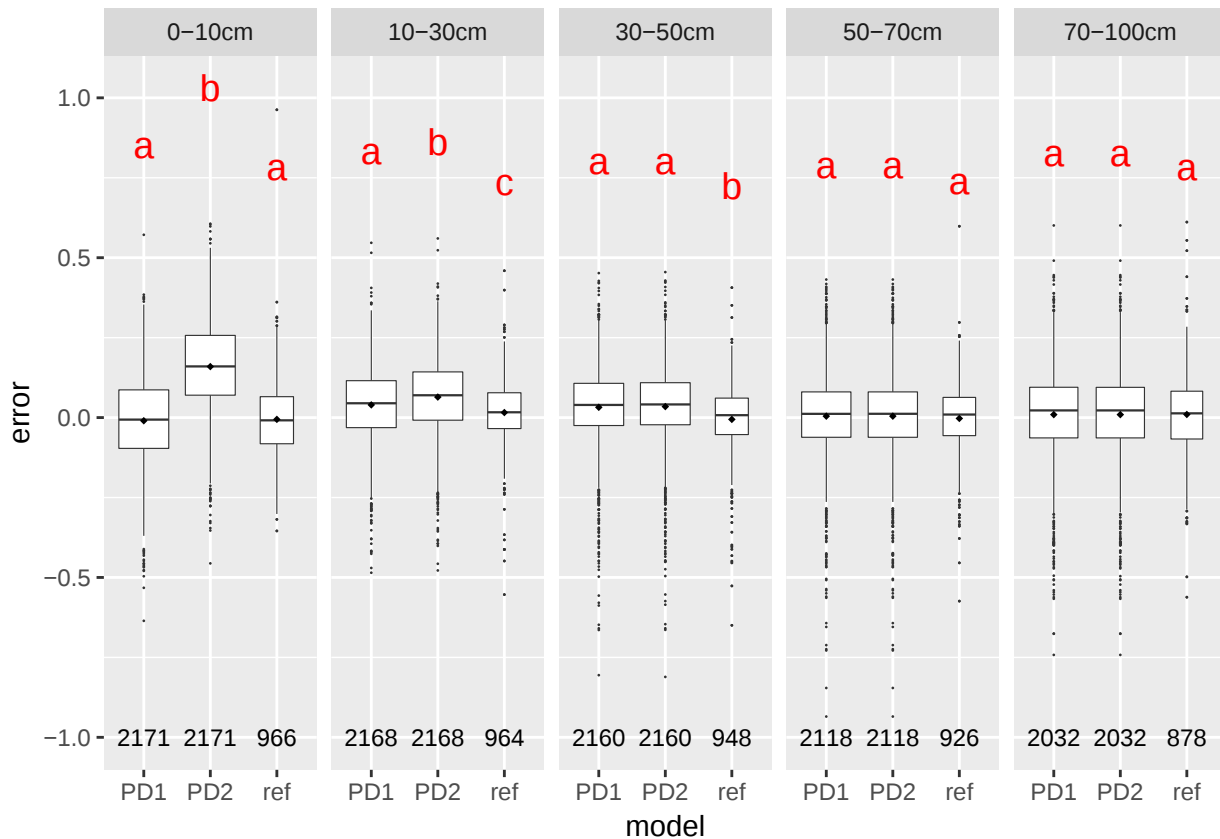
  temp0 = subset(data.mod, depth==layer)
  temp0 = droplevels(temp0)
  if (length(levels(temp0$model))==2) {
    temp1 = data.frame(stufe=levels(temp0$model),
      test= if(wilcox.test(error~model, temp0)$p.value<.05)
        {c("a", "b")}
      else
        {c("a", "a")},
      depth=layer)
    temp2 = rbind(temp2, temp1)
  } else {
    temp0 = pairwise.wilcox.test(temp0$error, temp0$model, p.adjust.method="bonferroni")
    temp0 = rcompanion::fullPTable(temp0$p.value)
    temp1 = multcompView::multcompLetters(temp0, compare="<",
      threshold=0.05,
      Letters=c(letters, LETTERS, "."),
      reversed = FALSE)

    temp1 = data.frame(as.vector(names(temp1$Letters)), as.vector(temp1$Letters))
    colnames(temp1) = c("model", "test")
    temp1$depth = layer

    temp2 = rbind(temp2, temp1)
  }
}

temp2 = na.omit(temp2)
temp = merge(temp, temp2, all.x=T)
temp = temp[order(temp$depth, temp$model),]
```

```
ggplot(data.mod,aes(model,error))+
  facet_grid(.~depth,scales="free") +
  geom_boxplot(outlier.shape = 16,
               outlier.size = .15,
               varwidth = T,
               size=.2) +
  stat_summary(fun.y=mean, geom="point", shape=18, size=1, color="black", fill="white") +
  geom_text(data = temp,aes(model,-1), label = temp$n, size=3) +
  geom_text(data = temp,aes(model,whisker+.5), label = temp$test, size=5, col = "red")
```



Calculate $\overline{use}$ - and $\overline{man}$ -effects

```
summary = data.mod %>%
  group_by(model,depth) %>%
  summarise(error=mean(error,na.rm=T)) %>%
  ungroup() %>%
  spread(model,error) %>%
  rename(error.PD1=PD1,
         error.PD2=PD2,
         error.ref=ref) %>%
  mutate(man.effect=ifelse(depth=="10-30cm" | depth=="30-50cm",
                           error.PD1-error.ref,
                           0),
         use.effect=ifelse(depth=="0-10cm" | depth=="10-30cm" | depth=="30-50cm",
                           error.PD2-error.PD1,
```

```
0))
```

Subtracting $\overline{use}$ - and $\overline{man}$ -effects from the measured $PD0$ gives the “natural” PD without anthropogenic disturbance $PD.nat$:

```
summary = data %>%
  filter(use=="C") %>%
  select(site,depth,PD0) %>%
  full_join(summary[,c("depth","man.effect","use.effect")]) %>%
  mutate(PD.nat=PD0-man.effect-use.effect)
```

Cause of compaction

Per site and depth

Threshold value used to separate compacted from non-compacted soil: 1.75 g cm^{-3}

Warning: This is a intermediate step. Evaluations per site and sample are not possible!

```
input=data.frame(threshold=1.75)
summary = summary %>%
  mutate(all = ifelse(PD0 > input$threshold,1,0),
         nat = ifelse(PD.nat > input$threshold,1,0),
         man = ifelse(PD.nat <= input$threshold &
                      PD.nat + man.effect > input$threshold,1,0),
         use = ifelse(PD.nat <= input$threshold &
                      man == 0 &
                      PD.nat + use.effect > input$threshold,1,0),
         use.man = ifelse(PD.nat <= input$threshold &
                          use == 0 &
                          man == 0 &
                          PD0 > input$threshold,1,0))
```

Per site

Choose cause from uppermost compacted layer

```
summary.Point = summary %>%
  mutate(nat = ifelse(nat==0,9,
                     ifelse(depth=="0-10cm",1,
                           ifelse(depth=="10-30cm",2,
                                   ifelse(depth=="30-50cm",3,
                                           ifelse(depth=="50-70cm",4,5))))),
         use = ifelse(use==0,9,
                     ifelse(depth=="0-10cm",1,
                           ifelse(depth=="10-30cm",2,
                                   ifelse(depth=="30-50cm",3,
                                           ifelse(depth=="50-70cm",4,5))))),
         man = ifelse(man==0,9,
                     ifelse(depth=="0-10cm",1,
                           ifelse(depth=="10-30cm",2,
                                   ifelse(depth=="30-50cm",3,
                                           ifelse(depth=="50-70cm",4,5))))),
         use.man = ifelse(use.man==0,9,
```

```

            ifelse(depth=="0-10cm",1,
                  ifelse(depth=="10-30cm",2,
                        ifelse(depth=="30-50cm",3,
                              ifelse(depth=="50-70cm",4,5)))))) %>%

group_by(site) %>%
summarise(nat = min(nat,na.rm=T),
          use = min(use,na.rm=T),
          man = min(man,na.rm=T),
          use.man = min(use.man,na.rm=T)) %>%
rowwise() %>%
mutate(all = min(nat,use,man,use.man)) %>%
mutate(nat = ifelse(all!=9 & nat==all,1,0),
       use = ifelse(all!=9 & use==all,1,0),
       man = ifelse(all!=9 & man==all,1,0),
       use.man = ifelse(all!=9 & use.man==all,1,0),
       depth = "Total",
       all = ifelse(all==9,0,1))

```

Aggregate depth increments

```

summary = bind_rows(summary,summary.Point)%>%
  group_by(depth) %>%
  summarise(n = sum(!is.na(all)),
            all = round(sum(all,na.rm=T)/n*100,1),
            fine = 100-all,
            nat = round(sum(nat,na.rm=T)/n*100,1),
            use = round(sum(use,na.rm=T)/n*100,1),
            use.man = round(sum(use.man,na.rm=T)/n*100,1),
            man = round(sum(man,na.rm=T)/n*100,1)) %>%
  arrange(depth)

```

Figure

```

temp0 = as.data.frame(t(summary)) %>%
  rownames_to_column(var = "name") %>%
  filter(name!="depth",name!="n",name!="all") %>%
  gather(Depth, VI, V1:V6) %>%
  mutate(name=as.factor(name),
         Depth=ifelse(Depth=="V1",10,
                      ifelse(Depth=="V2",30,
                            ifelse(Depth=="V3",50,
                                  ifelse(Depth=="V4",70,
                                        ifelse(Depth=="V5",110,150))))),
         VI=as.numeric(VI))

temp1 = data.frame(
  name=rep(levels(droplevels(temp0$name)),each=16),
  Depth=rep(0:15,length(levels(droplevels(temp0$name))))*10)

Kantig = temp0 %>%

```

```

full_join(temp1) %>%
  arrange(name,Depth) %>%
  fill(VI,.direction="up") %>%
  mutate(Depth=ifelse(Depth==0,5,
                      ifelse(Depth==150,155,Depth)),
         name=fct_relevel(name,"fine","nat","man","use","use.man"))

p=ggplot(Kantig,aes(x=Depth-5,y=VI,group=name,fill=name))+
  geom_area(position="fill",color="black")+
  coord_flip(ylim=c(0,1),
            xlim=c(0,150))+
  scale_x_continuous(trans="reverse",
                    breaks=c(0,10,30,50,70,40),
                    expand=c(0,0))+
  scale_y_continuous(expand=c(0,0),
                    breaks=c(0,.2,.4,.6,.8,1),
                    labels=c("0","20","40","60","80","100"))+
  scale_fill_viridis_d(name=NULL,
                      labels=c("Not compacted",
                                "Pedology + geology",
                                "Management (traffic)",
                                "Landuse (SOC loss)",
                                "Landuse + management"))+
  labs(y="Extent and cause of soil compaction [% of cropland]")+
  ggpubr::theme_pubr(18)+
  theme(legend.position="right",
        legend.justification="bottom",
        axis.ticks.y=element_blank(),
        axis.ticks.x=element_line(size=.5,color="black"),
        axis.line=element_blank(),
        axis.text.y=element_blank(),
        axis.text.x=element_text(size=13),
        axis.title.y=element_blank(),
        plot.margin=unit(c(40,50,5.5,110),"points"))+
  annotate("rect",xmin=100,xmax=120,ymin=-1,ymax=1.1,fill="white",col="white")+
  annotate("segment",x=100,xend=100,y=-Inf,yend=Inf,col="black")+
  annotate("segment",x=120,xend=120,y=-Inf,yend=Inf,col="black")+
  annotation_custom(bracketsGrob(0,1-2/3,0,1-.7*2/3,h=0.05,lwd=2,col="black"))+
  annotation_custom(bracketsGrob(0,1-.7*2/3,0,1-.5*2/3,h=0.05,lwd=2,col="black"))+
  annotation_custom(bracketsGrob(0,1-.5*2/3,0,1-.3*2/3,h=0.05,lwd=2,col="black"))+
  annotation_custom(bracketsGrob(0,1-.3*2/3,0,1-.1*2/3,h=0.05,lwd=2,col="black"))+
  annotation_custom(bracketsGrob(0,1-.1*2/3,0,1.0,h=0.05,lwd=2,col="black"))+
  annotation_custom(bracketsGrob(0,0,0,1-1.2*2/3,h=0.05,lwd=2,col="black"))+
  annotate("text",x=-14,y=-0.06,label="Depth",parse=T,size=6,hjust=1)+
  annotate("text",x=-5,y=-0.06,label="cm",parse=T,size=4,hjust=1)+
  annotate("text",x=c(5,20,40,60,85,135),y=rep(-0.06,6),
         label=c("0-10","10-30","30-50","50-70","70-100","Total"),size=5,hjust=1)

gt <- ggplot_gtable(ggplot_build(p))
gt$layout$clip[gt$layout$name == "panel"] <- "off"
grid.draw(gt)

```

