KID
Function

Fabian Blasch

21.08.2021

Packages

```
# Packages
get.package <- function(package){

  lapply(package, \(x){
    # check if packages are installed and if not install them
    if(!require(x, character.only = T)){
       install.packages(x)
    }
    # call package
    library(x, character.only = T)
  })

}

# exec
get.package(c("png", "jpeg", "tabulizer", "pdftools", "raster", "rgdal", "sp",
              "cluster"))




# since I will use Map() / lapply() alot for plotting I will wrap them in invisible()
invis.Map <- function(f, ...) invisible(Map(f, ...))
invis.lapply <- function(x, f, ...) invisible(lapply(x, f, ...))
```

Actual SRRI

We can obtain the actual SRRI from the file name. Later this data will be utilized to evaluate the classification accuracy of the applied methods.

```
# set
setwd("C:/Users/blasc/OneDrive/Documents/GitHub/KID/KIDs")

# files
file_names <- list.files(pattern = ".pdf", recursive = T)

# create df
dat.valid.SRRI <- as.data.frame(cbind("KID" = file_names,
                         "SRRI" = sapply(strsplit(sapply(strsplit(file_names, "_", fixed = T),
```

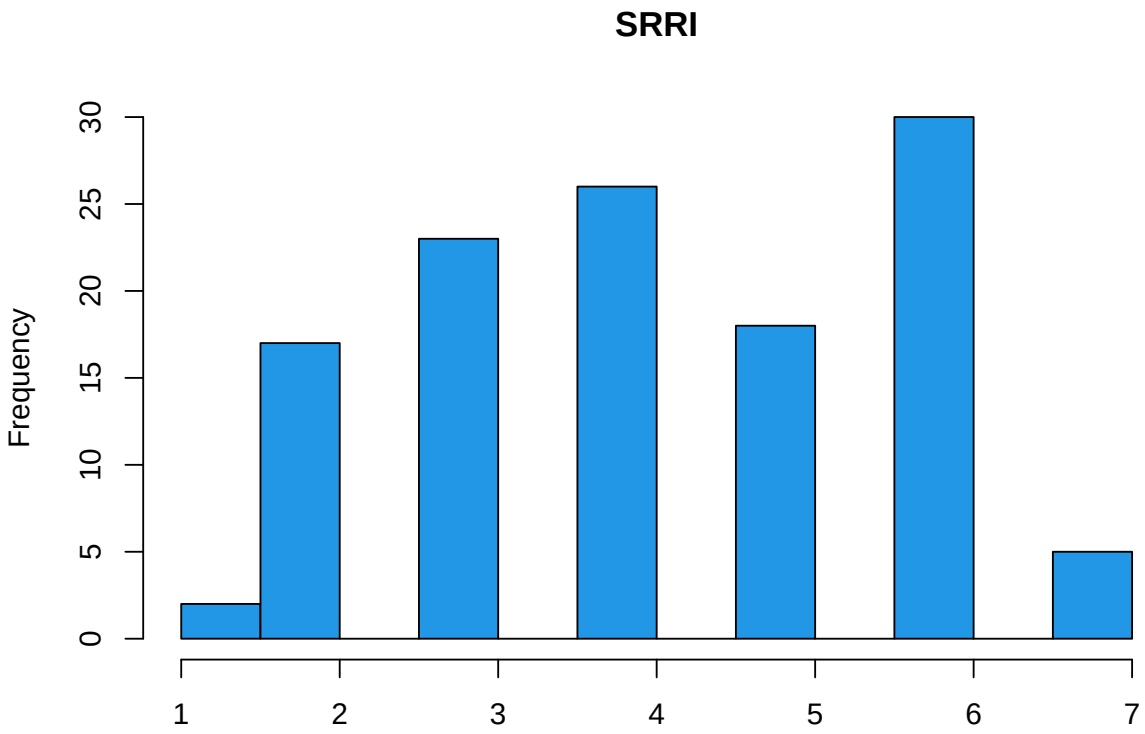
```
                                        function(x) x[length(x)]), ".", fixed = T), "[", 1)))

# split first col
dat.valid.SRRI[, "KAG"] <- sapply(strsplit(dat.valid.SRRI[, 1], "/"), "[", 1)
dat.valid.SRRI[, "KID"] <- sapply(strsplit(dat.valid.SRRI[, 1], "/"), "[", 2)

# order
dat.valid.SRRI <- dat.valid.SRRI[, c(3, 1, 2)]

# glimpse
head(dat.valid.SRRI, 7)
```

```
##       KAG             KID SRRI
## 1 Allianz ki-allakt_6.pdf    6
## 2 Allianz  ki-allap_6.pdf    6
## 3 Allianz ki-alleur_2.pdf    2
## 4 Allianz  ki-allna_6.pdf    6
## 5 Allianz ki-allnar_2.pdf    2
## 6 Allianz ki-allore_3.pdf    3
## 7 Allianz ki-allost_6.pdf    6
```

```
# dim
dim(dat.valid.SRRI)
```

```
## [1] 121   3
```

```
# Hist
hist(as.numeric(dat.valid.SRRI[, "SRRI"]), breaks = 10, main = "SRRI", col = 4, xlab = "")
```



Shade Color

To extract the SRRI the following colors are required and need to be converted to HEX.

```
# set
setwd("C:/Users/blasc/OneDrive/Documents/GitHub/KID/KIDs/Auxiliary")

# import
dat.col.KAG <- read.table(list.files(pattern = "RGB"),
                          col.names = c("KAG", "R", "G", "B"))

# add hex
sapply(as.data.frame(t(dat.col.KAG[, -1])),
       function(x) do.call(rgb, as.list(c(x, maxColorValue = 255)))) -> HEX

# bind
dat.col.KAG <- cbind(dat.col.KAG, "HEX" = HEX)

# display
dat.col.KAG
```

```
##               KAG   R   G   B     HEX
## V1     Raiffeisen   0  82 140 #00528C
## V2        Allianz 166 166 166 #A6A6A6
## V3         Amundi 204 210 219 #CCD2DB
## V4          Erste 166 166 166 #A6A6A6
```

```
## V5           IQAM 128 128 128 #808080
## V6         Kepler 204 204 204 #CCCCCC
## V7   Masterinvest  99 177 229 #63B1E5
## V8  Schoellerbank 217 217 217 #D9D9D9
## V9       Security 193 193 193 #C1C1C1
## V10         Union 196 197 199 #C4C5C7
```

SRRI Extraction Function

Given a KID document this function aims to extract the SRRI from the standard graph (usually) located on the first of two pages.

```
# source function
source("C:/Users/blasc/OneDrive/Documents/GitHub/KID/Code/Functions/SRRI_ext.R")
```

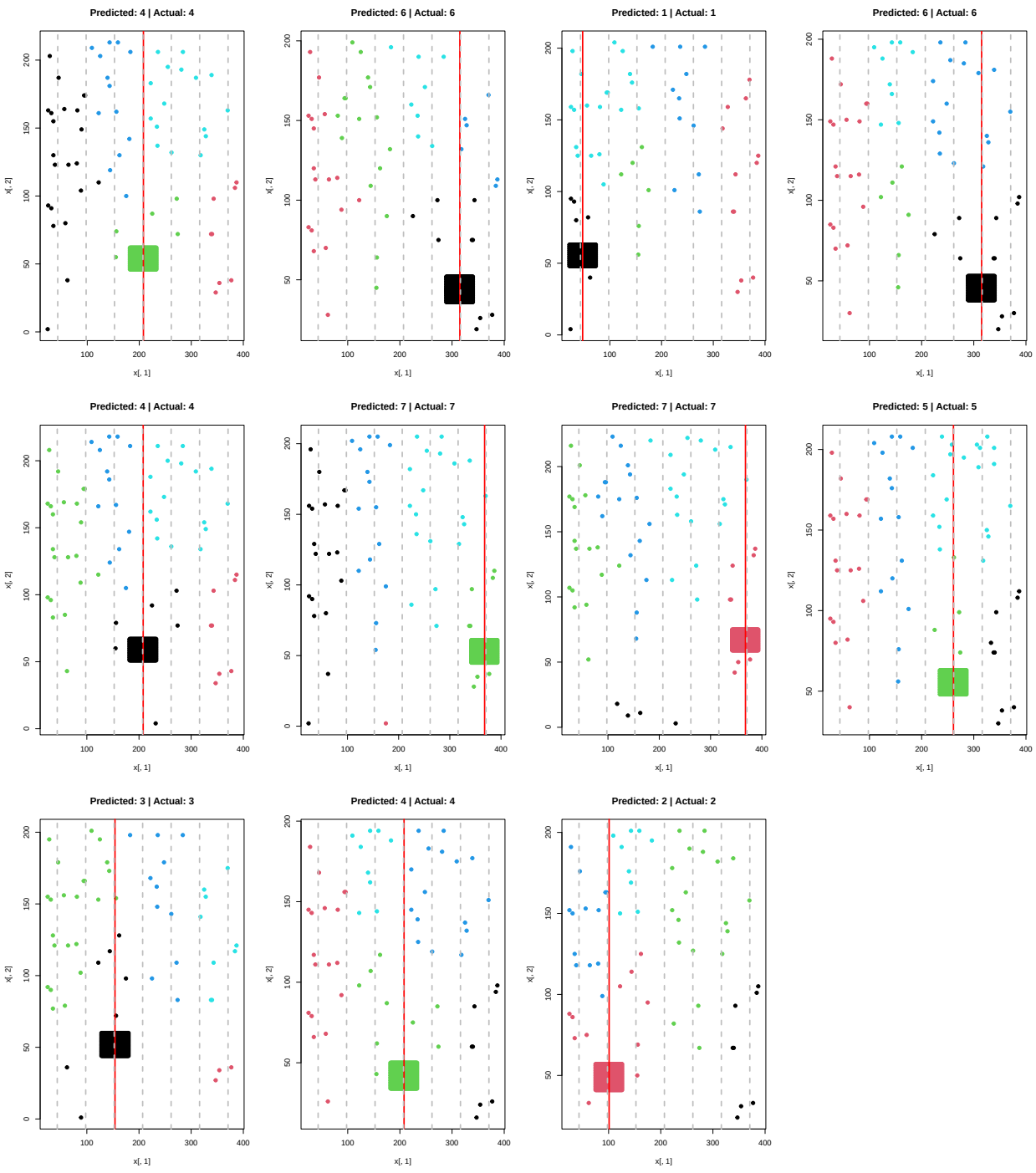
Tests

Starting with one KAG.

Erste

```
# set wd to file that contains
setwd("C:/Users/blasc/OneDrive/Documents/GitHub/KID/KIDs")

# safe dirs
dirs <- list.dirs()[-c(1, 4)] # remove hardcode later

# colors
col <- dat.col.KAG[order(dat.col.KAG[, "KAG"]), c("KAG", "HEX")]
col[5, 1] <- "Kepler Fonds"

# test Erste
Map(function(x, y){

  # set
  {setwd("C:/Users/blasc/OneDrive/Documents/GitHub/KID/KIDs")
   setwd(x)

  # ,pdfs
  file_nom <- list.files(pattern = ".pdf")}

   # FUN over all .pdfs
  lapply(file_nom, function(z){
    SRRI_ext(doc = z, col = y)
  })


}, dirs[3], col[3, 2]) -> erste.test


# extracted SRRI
cbind(dat.valid.SRRI[dat.valid.SRRI[, "KAG"] == "Erste", ],
      "Extracted" = sapply(erste.test[[1]], "[[", 2)) -> res
```

```r
par(mfrow = c(3, 4))

# plot
invis.Map(function(x, y, z, l, k){

  {plot(x[, 1], x[, 2], col = x[, ncol(x)], pch = 19, main = paste("Predicted:", z, "| Actual:", l))
  abline(v = y, col = "red", lwd = 2)
  lapply(k, function(x) abline(v = x, col = "grey", lwd = 2, lty = 2))}

}, lapply(erste.test[[1]], "[[", 3), sapply(erste.test[[1]], "[[", 4), res[, 4], res[, 3],
   lapply(erste.test[[1]], "[[", 5))
```



In the case of Erste the SRRI extraction works perfectly. Now the remaining KAGs will be examined.

```
invisible(

# store Errors
utils::capture.output(

# Map over dirs
Map(function(x, y){
```

```r
  # set
  {setwd("C:/Users/blasc/OneDrive/Documents/GitHub/KID/KIDs")
   setwd(x)

  # ,pdfs
  file_nom <- list.files(pattern = ".pdf")}

   # lapply over all .pdfs
  lapply(file_nom, function(z){

    # extract and error handle
    try(SRRI_ext(doc = z, col = y), silent = F)

  })

}, dirs, col[, 2]) -> test

, type = "message"))
```

```
## Error in SRRI_ext(doc = z, col = y) :
##   Error: No pixels of given color detected.
## Error in SRRI_ext(doc = z, col = y) :
##   Error: No pixels of given color detected.
## Error in SRRI_ext(doc = z, col = y) :
##   Error: No pixels of given color detected.
## Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.
## Error in SRRI_ext(doc = z, col = y) :
##   Error: No pixels of given color detected.
## Error in SRRI_ext(doc = z, col = y) :
##   Error: No pixels of given color detected.
## Error in pos.vec[page.SRRI] - off :
##   nicht-numerisches Argument für binären Operator
## Error in pos.vec[page.SRRI] - off :
##   nicht-numerisches Argument für binären Operator
## Error in pos.vec[page.SRRI] - off :
##   nicht-numerisches Argument für binären Operator
## Error in pos.vec[page.SRRI] - off :
##   nicht-numerisches Argument für binären Operator
## Error in pos.vec[page.SRRI] - off :
##   nicht-numerisches Argument für binären Operator
## Error in pos.vec[page.SRRI] - off :
##   nicht-numerisches Argument für binären Operator
## Error in pos.vec[page.SRRI] - off :
##   nicht-numerisches Argument für binären Operator
## Error in pos.vec[page.SRRI] - off :
##   nicht-numerisches Argument für binären Operator
## Error in pos.vec[page.SRRI] - off :
##   nicht-numerisches Argument für binären Operator
## Error in pos.vec[page.SRRI] - off :
##   nicht-numerisches Argument für binären Operator
## Error in pos.vec[page.SRRI] - off :
##   nicht-numerisches Argument für binären Operator
## Error in SRRI_ext(doc = z, col = y) :
```

```
##   Error: No pixels of given color detected.
## Error in SRRI_ext(doc = z, col = y) :
##   Error: No pixels of given color detected.
## Error in SRRI_ext(doc = z, col = y) :
##   Error: No pixels of given color detected.
## Error in SRRI_ext(doc = z, col = y) :
##   Error: Could not uniquely identify position of SRRI.
## Error in SRRI_ext(doc = z, col = y) :
##   Error: No pixels of given color detected.
## Error in SRRI_ext(doc = z, col = y) :
##   Error: No pixels of given color detected.
## Error in SRRI_ext(doc = z, col = y) :
##   Error: No pixels of given color detected.
## Error in SRRI_ext(doc = z, col = y) :
##   Error: No pixels of given color detected.
## Error in SRRI_ext(doc = z, col = y) :
##   Error: No pixels of given color detected.
## Error in SRRI_ext(doc = z, col = y) :
##   Error: No pixels of given color detected.
## Error in SRRI_ext(doc = z, col = y) :
##   Error: No pixels of given color detected.
## Error in SRRI_ext(doc = z, col = y) :
##   Error: No pixels of given color detected.
## Error in SRRI_ext(doc = z, col = y) :
##   Error: No pixels of given color detected.
## Error in SRRI_ext(doc = z, col = y) :
##   Error: No pixels of given color detected.
## Error in SRRI_ext(doc = z, col = y) :
##   Error: No pixels of given color detected.
## Error in SRRI_ext(doc = z, col = y) :
##   Error: No pixels of given color detected.
## Error in SRRI_ext(doc = z, col = y) :
##   Error: No pixels of given color detected.
## Error in SRRI_ext(doc = z, col = y) :
##   Error: No pixels of given color detected.
## Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.
## Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.
## Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.
## Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.
## Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.
## Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.
## Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.
## Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.
## Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.
## Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.
## Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.
## Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.
## Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.
```

```r
# error index
lapply(test, function(x){

  # error ind
  which(sapply(x, class) == "try-error")
```

```
}) -> err.tmp

# retrieve error throwing funds with ind
do.call(rbind, Map(function(x, y, z){

  if(length(y) > 0){

    {setwd("C:/Users/blasc/OneDrive/Documents/GitHub/KID/KIDs")
     setwd(z)

    # .pdfs
    file_nom <- list.files(pattern = ".pdf")}

    # subset
    cbind(rep(z, length(y)),
          file_nom[y],
          sapply(x[y], "[", 1))

  } else {
   cbind(NA, NA, "No errros.")
  }

}, test, err.tmp, dirs))
```

```
##       [,1]
##  [1,] NA
##  [2,] "./Amundi"
##  [3,] "./Amundi"
##  [4,] "./Amundi"
##  [5,] "./Amundi"
##  [6,] "./Amundi"
##  [7,] "./Amundi"
##  [8,] NA
##  [9,] NA
## [10,] "./Kepler Fonds"
## [11,] "./Kepler Fonds"
## [12,] "./Kepler Fonds"
## [13,] "./Kepler Fonds"
## [14,] "./Kepler Fonds"
## [15,] "./Kepler Fonds"
## [16,] "./Kepler Fonds"
## [17,] "./Kepler Fonds"
## [18,] "./Kepler Fonds"
## [19,] "./Kepler Fonds"
## [20,] "./Kepler Fonds"
## [21,] NA
## [22,] "./Raiffeisen"
## [23,] "./Raiffeisen"
## [24,] "./Raiffeisen"
## [25,] "./Raiffeisen"
## [26,] "./Raiffeisen"
## [27,] "./Raiffeisen"
## [28,] "./Raiffeisen"
```

```
## [29,] "./Raiffeisen"
## [30,] "./Raiffeisen"
## [31,] "./Raiffeisen"
## [32,] "./Raiffeisen"
## [33,] "./Raiffeisen"
## [34,] "./Raiffeisen"
## [35,] "./Raiffeisen"
## [36,] "./Raiffeisen"
## [37,] "./Raiffeisen"
## [38,] "./Raiffeisen"
## [39,] "./Raiffeisen"
## [40,] "./Schoellerbank"
## [41,] "./Schoellerbank"
## [42,] "./Schoellerbank"
## [43,] NA
## [44,] "./Union"
## [45,] "./Union"
## [46,] "./Union"
## [47,] "./Union"
## [48,] "./Union"
## [49,] "./Union"
## [50,] "./Union"
## [51,] "./Union"
## [52,] "./Union"
## [53,] "./Union"
##       [,2]
##  [1,] NA
##  [2,] "kiid-at0000619317-deu-aut_6.pdf"
##  [3,] "kiid-at0000674908-deu-aut_6.pdf"
##  [4,] "kiid-at0000767736-deu-aut_6.pdf"
##  [5,] "kiid-at0000822382-deu-aut_4.pdf"
##  [6,] "kiid-at0000857412-deu-aut_6.pdf"
##  [7,] "kiid-at0000903125-deu-aut_4.pdf"
##  [8,] NA
##  [9,] NA
## [10,] "report (10)_5.pdf"
## [11,] "report (11)_5.pdf"
## [12,] "report (2)_2.pdf"
## [13,] "report (3)_6.pdf"
## [14,] "report (4)_6.pdf"
## [15,] "report (5)_6.pdf"
## [16,] "report (6)_4.pdf"
## [17,] "report (7)_4.pdf"
## [18,] "report (8)_2.pdf"
## [19,] "report (9)_3.pdf"
## [20,] "short_invest_rentenfonds_2.pdf"
## [21,] NA
## [22,] "KID_DSTR_AT_mRG_de_20210531_2.pdf"
## [23,] "KID_ENA_AT_mRG_de_20210531_7.pdf"
## [24,] "KID_ERE_AT_mRG_de_20210531_3.pdf"
## [25,] "KID_ESC_AT_mRG_de_20210531_6.pdf"
## [26,] "KID_HCA_AT_mRG_de_20210601_5.pdf"
## [27,] "KID_OKR_AT_mRG_de_20210531_2.pdf"
## [28,] "KID_RAA_AT_mRG_de_20210531_5.pdf"
```

```
## [29,] "KID_RAC_AT_mRG_de_20210531_5.pdf"
## [30,] "KID_Raiffeisen-Euro-Rent_20210531_3.pdf"
## [31,] "KID_RFE_AT_mRG_de_20210531_4.pdf"
## [32,] "KID_RFS_AT_mRG_de_20210531_3.pdf"
## [33,] "KID_RFW_AT_mRG_de_20210618_4.pdf"
## [34,] "KID_RGASP_AT_mRG_de_20210531_7.pdf"
## [35,] "KID_RIA_AT_mRG_de_20210601_6.pdf"
## [36,] "KID_RIS_AT_mRG_de_20210531_3.pdf"
## [37,] "KID_RNM_AT_mRG_de_20210531_4.pdf"
## [38,] "KID_RNS_AT_mRG_de_20210531_3.pdf"
## [39,] "KID_RNW_AT_mRG_de_20210707_5.pdf"
## [40,] "Schoellerbank_Global_Balanced_KID_file_4.pdf"
## [41,] "Schoellerbank_Global_Balanced_Plus_KID_file_4.pdf"
## [42,] "Schoellerbank_Global_Income_KID_file_3.pdf"
## [43,] NA
## [44,] "KCD_Union_Nachhaltig_RENTEN_DE0005326524-20210212-1-AT_3.pdf"
## [45,] "LIGA_Pax_Rent_Union_DE0008491226-20210510-1-AT_3.pdf"
## [46,] "LIGA_Portfolio_Concept_A_LU1172417856-20210510-1-AT_3.pdf"
## [47,] "PrivatFonds__Konsequent_pro_LU0493584741-20210618-1-AT_4.pdf"
## [48,] "UniCommodities_LU0249045476-20200803-1-LU_6.pdf"
## [49,] "UniEuroKapital_Corporates_A_LU0168092178-20210212-1-AT_3.pdf"
## [50,] "UniStrategie__Dynamisch_DE0005314124-20210212-1-AT_5.pdf"
## [51,] "UniStrategie__Konservativ_DE0005314108-20210521-1-AT_3.pdf"
## [52,] "UniStruktur_LU1529950914-20210312-1-AT_4.pdf"
## [53,] "UniValueFonds__Global_A_LU0126315885-20210802-1-AT_5.pdf"
##       [,3]
##  [1,] "No errros."
##  [2,] "Error in SRRI_ext(doc = z, col = y) : \n  Error: No pixels of given color detected.\n"
##  [3,] "Error in SRRI_ext(doc = z, col = y) : \n  Error: No pixels of given color detected.\n"
##  [4,] "Error in SRRI_ext(doc = z, col = y) : \n  Error: No pixels of given color detected.\n"
##  [5,] "Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.\n"
##  [6,] "Error in SRRI_ext(doc = z, col = y) : \n  Error: No pixels of given color detected.\n"
##  [7,] "Error in SRRI_ext(doc = z, col = y) : \n  Error: No pixels of given color detected.\n"
##  [8,] "No errros."
##  [9,] "No errros."
## [10,] "Error in pos.vec[page.SRRI] - off : \n  nicht-numerisches Argument für binären Operator\n"
## [11,] "Error in pos.vec[page.SRRI] - off : \n  nicht-numerisches Argument für binären Operator\n"
## [12,] "Error in pos.vec[page.SRRI] - off : \n  nicht-numerisches Argument für binären Operator\n"
## [13,] "Error in pos.vec[page.SRRI] - off : \n  nicht-numerisches Argument für binären Operator\n"
## [14,] "Error in pos.vec[page.SRRI] - off : \n  nicht-numerisches Argument für binären Operator\n"
## [15,] "Error in pos.vec[page.SRRI] - off : \n  nicht-numerisches Argument für binären Operator\n"
## [16,] "Error in pos.vec[page.SRRI] - off : \n  nicht-numerisches Argument für binären Operator\n"
## [17,] "Error in pos.vec[page.SRRI] - off : \n  nicht-numerisches Argument für binären Operator\n"
## [18,] "Error in pos.vec[page.SRRI] - off : \n  nicht-numerisches Argument für binären Operator\n"
## [19,] "Error in pos.vec[page.SRRI] - off : \n  nicht-numerisches Argument für binären Operator\n"
## [20,] "Error in pos.vec[page.SRRI] - off : \n  nicht-numerisches Argument für binären Operator\n"
## [21,] "No errros."
## [22,] "Error in SRRI_ext(doc = z, col = y) : \n  Error: No pixels of given color detected.\n"
## [23,] "Error in SRRI_ext(doc = z, col = y) : \n  Error: No pixels of given color detected.\n"
## [24,] "Error in SRRI_ext(doc = z, col = y) : \n  Error: No pixels of given color detected.\n"
## [25,] "Error in SRRI_ext(doc = z, col = y) : \n  Error: Could not uniquely identify position of SRRI
## [26,] "Error in SRRI_ext(doc = z, col = y) : \n  Error: No pixels of given color detected.\n"
## [27,] "Error in SRRI_ext(doc = z, col = y) : \n  Error: No pixels of given color detected.\n"
## [28,] "Error in SRRI_ext(doc = z, col = y) : \n  Error: No pixels of given color detected.\n"
```

```
## [29,] "Error in SRRI_ext(doc = z, col = y) : \n  Error: No pixels of given color detected.\n"
## [30,] "Error in SRRI_ext(doc = z, col = y) : \n  Error: No pixels of given color detected.\n"
## [31,] "Error in SRRI_ext(doc = z, col = y) : \n  Error: No pixels of given color detected.\n"
## [32,] "Error in SRRI_ext(doc = z, col = y) : \n  Error: No pixels of given color detected.\n"
## [33,] "Error in SRRI_ext(doc = z, col = y) : \n  Error: No pixels of given color detected.\n"
## [34,] "Error in SRRI_ext(doc = z, col = y) : \n  Error: No pixels of given color detected.\n"
## [35,] "Error in SRRI_ext(doc = z, col = y) : \n  Error: No pixels of given color detected.\n"
## [36,] "Error in SRRI_ext(doc = z, col = y) : \n  Error: No pixels of given color detected.\n"
## [37,] "Error in SRRI_ext(doc = z, col = y) : \n  Error: No pixels of given color detected.\n"
## [38,] "Error in SRRI_ext(doc = z, col = y) : \n  Error: No pixels of given color detected.\n"
## [39,] "Error in SRRI_ext(doc = z, col = y) : \n  Error: No pixels of given color detected.\n"
## [40,] "Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.\n"
## [41,] "Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.\n"
## [42,] "Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.\n"
## [43,] "No errros."
## [44,] "Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.\n"
## [45,] "Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.\n"
## [46,] "Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.\n"
## [47,] "Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.\n"
## [48,] "Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.\n"
## [49,] "Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.\n"
## [50,] "Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.\n"
## [51,] "Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.\n"
## [52,] "Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.\n"
## [53,] "Error in SRRI_ext(doc = z, col = y) : Error: Could not detect SRRI.\n"
```

```
# Plot
          # Map(function(x, y){
          #
          #   # match
          #   cbind(dat.valid.SRRI[dat.valid.SRRI[, "KAG"] == x, ],
          #         "Extracted" = tryCatch({sapply(test, "[[", 2)}, function(e){}))
          #
          #
          # }, col[, 1], test)
          #
          #
          # par(mfrow = c(3, 4))
          #
          # # plot
          # invis.Map(function(x, y, z, l, k){
          #
          #   {plot(x[, 1], x[, 2], col = x[, ncol(x)], pch = 19, main = paste("Predicted:", z, "| Actu
          #   abline(v = y, col = "red", lty = 2, lwd = 2)
          #   lapply(k, function(x) abline(v = x, col = 4, lwd = 2))}
          #
          # }, lapply(erste.test[[1]], "[[", 3), sapply(erste.test[[1]], "[[", 4), res[, 4], res[, 3],
          #    lapply(erste.test[[1]], "[[", 5))
```