

```
source('RECURSOS-INVESTIGACION/R/camel-getCamelIndNormalizada.R')
source('RECURSOS-INVESTIGACION/R/camel-get-limit-es-rangos.R')
source("RECURSOS-INVESTIGACION/R/camel-get-tabla-referencia-calificaciones.R")
source('RECURSOS-INVESTIGACION/R/camel-calificacion-indicadores.R')
source('RECURSOS-INVESTIGACION/R/camel-calificacion-entidad.R')
source('RECURSOS-INVESTIGACION/R/camel-getListStatsIndicadores.R')
source('RECURSOS-INVESTIGACION/R/camel-getListPlotsIndicadores.R')
source('RECURSOS-INVESTIGACION/R/camel-get-datIdsNamesCamelInds.R')
source("RECURSOS-INVESTIGACION/R/camel-render-tables-indicadoresStats.R")
source('RECURSOS-INVESTIGACION/R/camel-tendencia-stats-overview.R')
source('RECURSOS-INVESTIGACION/R/camel-plot-functions.R')
source("RECURSOS-INVESTIGACION/R/render-table-basic.R")
```

```
require(stringr)
```

```
## Loading required package: stringr
```

```
require(ggplot2)
```

```
## Loading required package: ggplot2
```

```
require(patchwork)
```

```
## Loading required package: patchwork
```

```
require(knitr)
```

```
## Loading required package: knitr
```

```
require(kableExtra)
```

```
## Loading required package: kableExtra
```

```
require(dplyr)
```

```
## Loading required package: dplyr
```

```
##
```

```
## Attaching package: 'dplyr'
```

```
## The following object is masked from 'package:kableExtra':
```

```
##
```

```
##   group_rows
```

```
## The following objects are masked from 'package:stats':
```

```
##
```

```
##   filter, lag
```

```
## The following objects are masked from 'package:base':
```

```
##
```

```
##   intersect, setdiff, setequal, union
```

```

if (!('dat' %in% ls())) {dat <- NULL}

datCamelIndNorm <- getDatCamelIndNormalizada(gestionInc=2014,by='ENTIDAD',dat=dat) %>% filter(TIPO_DE_E

## Loading required package: openxlsx

datCamelRangosLimites <- getDatCamelRangosLimites(datCamelIndNorm)

datCamelCalificacionIndicadores <-
  getDatCamelCalificacionIndicadores(datCamelIndNorm,datCamelRangosLimites)
datCamelCalificacionIndicadoresHard <-
  getDatCamelCalificacionIndicadores(datCamelIndNorm,datCamelRangosLimites,
                                     calificacionDuro = TRUE)

datCamelCalificacionEntidadSinPoderar <-
  getDatCamelCalificacionEntidad(datCamelCalificacionIndicadores, FALSE)
datCamelCalificacionEntidadSinPoderarHard <-
  getDatCamelCalificacionEntidad(datCamelCalificacionIndicadoresHard, FALSE)

datCamelCalificacionEntidad <-
  getDatCamelCalificacionEntidad(datCamelCalificacionIndicadores)
datCamelCalificacionEntidadHard <-
  getDatCamelCalificacionEntidad(datCamelCalificacionIndicadoresHard)

tableRefenceCamelCalificaciones <- getTableRefenceCamelCalificaciones()
datIdsNamesCamelInds <- getDatIdsNamesCamelIndicadores()

```

Resultados CAMEL del sistema financiero

Los sección de resultados CAMEL del sistema financiero de Bolivia presenta la información resultado del diagnostico de la solidez financiera de los sectores financieros contenidos en la misma sujeta a los indicadores, rangos y ponderaciones antes presentados.

Resutados CAMEL con rangos flexibles sin ponderar

Los resultados CAMEL con rangos flexibles sin ponderar agrupados por tipo de indicador se presentan a continuación:

```

ids <- c('C','A','M','E','L')
listResult <- getListStatsOverviewInd(ids,
                                     datCamelCalificacionEntidadSinPoderar,
                                     datIdsNamesCamelInds,
                                     TRUE)

```

```
## Loading required package: fpp2
```

```

## Registered S3 method overwritten by 'quantmod':
##   method      from
##   as.zoo.data.frame zoo

```

```
## -- Attaching packages ----- fpp2 2.5 --
```

```
## v forecast 8.21      v expsmooth 2.3
## v fma        2.5
```

```
##
```

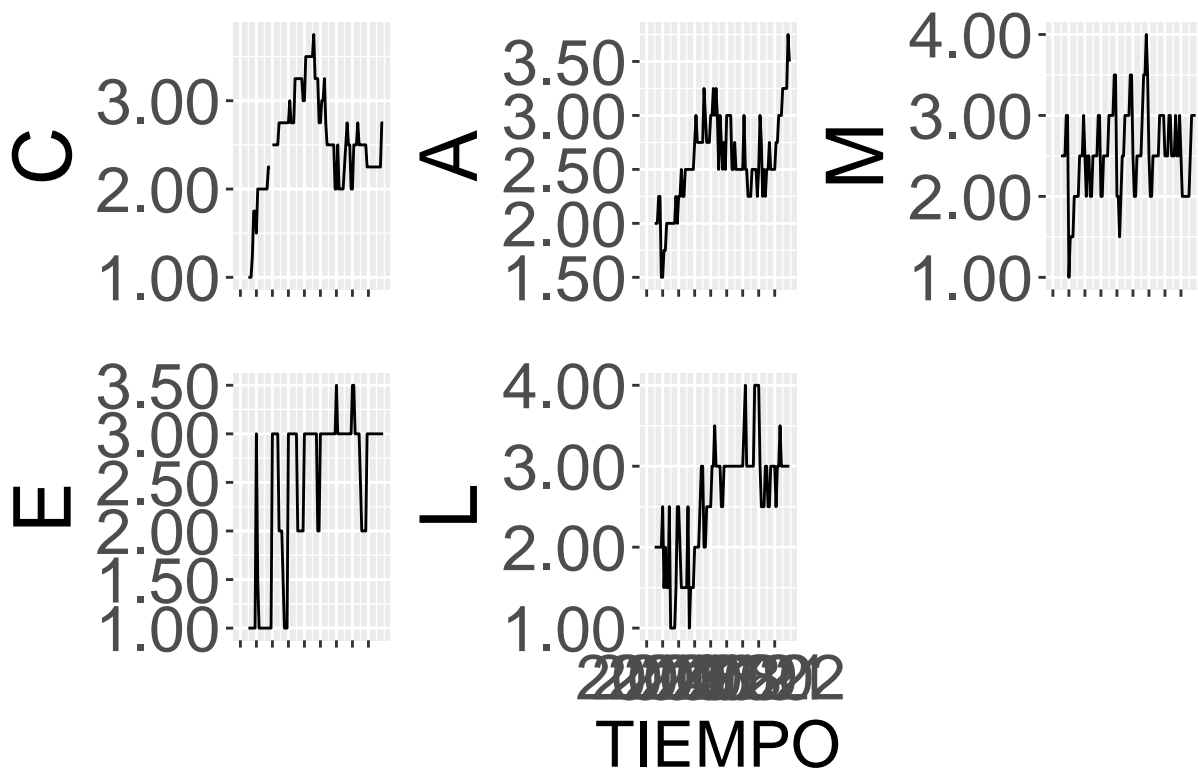
```
## Scale for x is already present.
## Adding another scale for x, which will replace the existing scale.
## Scale for x is already present.
## Adding another scale for x, which will replace the existing scale.
## Scale for x is already present.
## Adding another scale for x, which will replace the existing scale.
## Scale for x is already present.
## Adding another scale for x, which will replace the existing scale.
## Scale for x is already present.
## Adding another scale for x, which will replace the existing scale.
```

```
listPlots <- getListPlotsInd(listResult, scaleOne = FALSE, numDigScale = 0.01)
```

```
## Loading required package: scales
```

```
listTablesRender <- renderListTablesInd(listResult, 'Tendencia y estadísticas C-A-M-E-L con rangos flex
```

```
wrap_plots(listPlots)
```



```
listTablesRender %>%
  footnote(general_title = 'NOTA:',
           general = 'Los sectores en la tabla fueron ordenados en función del más a
```

Table 1: Tendencia y estadísticas C-A-M-E-L con rangos flexibles sin ponderar

TIPO DE ENTIDAD	TENDENCIA	PROMEDIO	DESVIACION	MINIMO	MAXIMO
C					
BFS	0.002868	3	0.555296	1	4
A					
BFS	0.008079	3	0.418297	2	4
M					
BFS	0.003998	3	0.519471	1	4
E					
BFS	0.017879	2	0.813937	1	4
L					
BFS	0.017277	3	0.716398	1	4

NOTA:

Los sectores en la tabla fueron ordenados en función del más al menos favorable dado su indicador promedio.

Resultados CAMEL con rangos estrictos sin ponderar

Los resultados CAMEL con rangos estrictos sin ponderar agrupados por tipo de indicador se presentan a continuación:

```
ids <- c('C','A','M','E','L')
listResult <- getListStatsOverviewInd(ids,
                                       datCamelCalificacionEntidadSinPoderarHard,
                                       datIdsNamesCamelInds,
                                       TRUE)
```

```
## Scale for x is already present.
## Adding another scale for x, which will replace the existing scale.
## Scale for x is already present.
## Adding another scale for x, which will replace the existing scale.
## Scale for x is already present.
## Adding another scale for x, which will replace the existing scale.
## Scale for x is already present.
## Adding another scale for x, which will replace the existing scale.
## Scale for x is already present.
## Adding another scale for x, which will replace the existing scale.
```

```
listPlots <- getListPlotsInd(listResult)
listTablesRender <- renderListTablesInd(listResult, 'Tendencia y estadísticas C-A-M-E-L con rangos estr
```

```
listTablesRender %>%
  footnote(general_title = 'NOTA:',
           general = 'Los sectores en la tabla fueron ordenados en función del más a
```

Table 2: Tendencia y estadísticas C-A-M-E-L con rangos estrictos sin ponderar

TIPO DE ENTIDAD	TENDENCIA	PROMEDIO	DESVIACION	MINIMO	MAXIMO
C					
BFS	0.004656	3	0.606922	2	4
A					
BFS	0.008557	4	0.432838	2	4
M					
BFS	0.006079	4	0.558376	2	5
E					
BFS	0.021119	3	0.951911	1	4
L					
BFS	0.019695	4	0.805244	2	5

NOTA:

Los sectores en la tabla fueron ordenados en función del más al menos favorable dado su indicador promedio.

Calificación CAMEL

La calificación CAMEL contiene dentro si las calificaciones ponderadas asignadas a los resultados de los diferentes indicadores, don 1 indica un sector solvente en todos aspectos y 5 serios problemas de solidez, así también una tendencia negativa se traduce en un incremento de la solidez financiera.

```
datAuxCAMEL <-  
  datCamelCalificacionEntidad %>%  
  select(ID, TIPO_DE_ENTIDAD, FECHA) %>%  
  mutate(CAMEL_FLEXIBLE=datCamelCalificacionEntidad$CAMEL,  
         CAMEL_ESTRICTO=datCamelCalificacionEntidadHard$CAMEL)
```

```
ids <- c('CAMEL_FLEXIBLE', 'CAMEL_ESTRICTO')  
listResult <- getListStatsOverviewInd(ids, datAuxCAMEL, datIdsNamesCamelInds, TRUE)
```

```
## Scale for x is already present.  
## Adding another scale for x, which will replace the existing scale.  
## Scale for x is already present.  
## Adding another scale for x, which will replace the existing scale.
```

```
listPlots <- getListPlotsInd(listResult)  
listTablesRender <- renderListTablesInd(listResult, 'Tendencia y estadísticas CAMEL', 1)
```

```
listTablesRender %>%  
  footnote(general_title = 'NOTA:',  
          general = 'Los sectores en la tabla fueron ordenados en función del más a
```

Table 3: Tendencia y estadísticas CAMEL

TIPO DE ENTIDAD	TENDENCIA	PROMEDIO	DESVIACION	MINIMO	MAXIMO
CAMEL CON RANGOS FLEXIBLES					
BFS	0.009400	3	0.424965	2	3
CAMEL CON RANGOS ESTRICTOS					
BFS	0.011207	3	0.499097	2	4

NOTA:

Los sectores en la tabla fueron ordenados en función del más al menos favorable dado su indicador promedio.

```
nameIndById <- listResult$CAMEL_FLEXIBLE$nameIndById  
nameEntBestPromedio <- listResult$CAMEL_FLEXIBLE$nameEntBestPromedio  
bestPromedio <- listResult$CAMEL_FLEXIBLE$bestPromedio
```

El promedio mas favorable en el CAMEL con rangos flexibles a través del tiempo esta dado en el sector de bfs con un promedio de 3.

```
nameIndById <- listResult$CAMEL_ESTRICTO$nameIndById  
nameEntBestPromedio <- listResult$CAMEL_ESTRICTO$nameEntBestPromedio  
bestPromedio <- listResult$CAMEL_ESTRICTO$bestPromedio
```

El promedio mas favorable en el CAMEL con rangos estrictos a través del tiempo esta dado en el sector de bfs con un promedio de 3.