

ANEXO 1

Código fonte do SAS© para extração e tratamento da
base de dados do DIOPS da despesa assistencial
acumulada em 12 meses

```

*****
* Extrai dados do DIOPS e trata;
* Rateia os dados de despesas e peona que não têm quebra por tipo de contratação, tipo de financiamento
* ou tipo de vigência (antigo/novo)
* Calcula valor do trimestre (DRE) e acumulado de 4 trimestres (12 meses);
*****;

* 1) Extrai dados do DIOPS;
***** !!!!!!! FILTRO !!!!!!! *****;

proc sql;
* Filtra contas contábeis relevantes e formata valores ;;
create table diops0 as
select CD_OPERADORA,
DT_INICIO_PERIODO,
CD_CONTA_CONTABIL,
(length(strip(CD_CONTA_CONTABIL))) AS N_CHAR,
DESCRICAO,
(sum(vl_saldo_inicial)) FORMAT=COMMAX20. AS VL_SALDO_INICIAL,
(SUM(VL_SALDO_FINAL)) FORMAT=COMMAX20. AS VL_SALDO_FINAL
from diope.vw_balancete_operadora_SAS
where (&DT_INI_t1 <= dt_inicio_periodo <= &DT_INI_t2)
and substr(cd_conta_contabil,1,5)='41111'
and cd_conta_analitica is missing
group by CD_OPERADORA,
DT_INICIO_PERIODO,
CD_CONTA_CONTABIL,
calculated N_CHAR,
DESCRICAO;

* exclui operadoras odontológicas e administradoras;;
create table al.DIOPS as
select t1.cd_operadora,
t2.NO_RAZAO,
t2.DS_CLASSIFICACAO,
t1.n_char,
t1.cd_conta_contabil,
year (DATEPART (DT_INICIO_PERIODO)) *100+month (DATEPART (DT_INICIO_PERIODO))+2 as dt_diops,
(sum(vl_saldo_inicial)) FORMAT=COMMAX20. AS VL_SALDO_INICIAL,
(SUM(VL_SALDO_FINAL)) FORMAT=COMMAX20. AS VL_SALDO_FINAL
from diops0 t1 inner join diope.vw_operadoras_public t2 ON t1.CD_OPERADORA=t2.CO_OPERADORA
where t2.DS_CLASSIFICACAO NOT IN ('Cooperativa odontológica','Odontologia de Grupo','Administradora','Administradora de Benefícios')
group by t1.cd_operadora,
t2.NO_RAZAO,
t2.DS_CLASSIFICACAO,
t1.n_char,
t1.cd_conta_contabil,
calculated dt_diops;
quit;
*/

*****;
* Identifica as despesas a serem rateadas: SUS e Outros pagamentos 2016;;
proc sql;
create table diops1 as
select
t1.cd_operadora,
t1.NO_RAZAO,
case
when t1.ds_classificacao = "Autogestão" then t1.ds_classificacao
else "Outras"
end as modalidade,
t1.dt_diops,
t1.n_char,
t1.cd_conta_contabil,

(CASE
WHEN substr(t1.cd_conta_contabil,1,6)='411111' and substr(t1.cd_conta_contabil,8,1)='1' THEN 'ind_a'
WHEN substr(t1.cd_conta_contabil,1,6)='411111' and substr(t1.cd_conta_contabil,8,1)='2' THEN 'ind_p'
WHEN substr(t1.cd_conta_contabil,1,6)='411111' and substr(t1.cd_conta_contabil,8,1)='3' THEN 'ade_a_pre'
WHEN substr(t1.cd_conta_contabil,1,6)='411111' and substr(t1.cd_conta_contabil,8,1)='4' THEN 'ade_p_pre'
WHEN substr(t1.cd_conta_contabil,1,6)='411111' and substr(t1.cd_conta_contabil,8,1)='5' THEN 'emp_a_pre'
WHEN substr(t1.cd_conta_contabil,1,6)='411111' and substr(t1.cd_conta_contabil,8,1)='6' THEN 'emp_p_pre'

when substr(t1.cd_conta_contabil,1,6)='411111' and substr(t1.cd_conta_contabil,8,1) in ('7','9')
THEN 'rateio_pre' /* sus e outros pagamentos 2016 */

WHEN substr(t1.cd_conta_contabil,1,6)='411112' and substr(t1.cd_conta_contabil,8,1)='3' THEN 'ade_a_pos'
WHEN substr(t1.cd_conta_contabil,1,6)='411112' and substr(t1.cd_conta_contabil,8,1)='4' THEN 'ade_p_pos'
WHEN substr(t1.cd_conta_contabil,1,6)='411112' and substr(t1.cd_conta_contabil,8,1)='5' THEN 'emp_a_pos'
WHEN substr(t1.cd_conta_contabil,1,6)='411112' and substr(t1.cd_conta_contabil,8,1)='6' THEN 'emp_p_pos'

WHEN substr(t1.cd_conta_contabil,1,6)='411112' and substr(t1.cd_conta_contabil,8,1) in ('1','2','7','9')
THEN 'rateio_pos' /* sus e outros pagamentos 2016 */

ELSE ""
END) AS Grupo_CC,

(sum(vl_saldo_inicial)) FORMAT=COMMAX20. AS VL_SALDO_INICIAL,
(SUM(VL_SALDO_FINAL)) FORMAT=COMMAX20. AS VL_SALDO_FINAL

from al.diops t1

```

```

GROUP BY t1.CD_OPERADORA, t1.NO_RAZAO, calculated modalidade,
DT_DIOPS,
n_char,
cd_conta_contabil,
calculated Grupo_CC
;quit;

*****!!!!!!!!!!!!!!!!!!!!!! FILTRO !!!!!!!!!!!!!!!!!!!!!!!*****;
* seleciona as contas contábeis de 9 dígitos ;;
* consolida os totais nas categorias de análise;;
proc sql;
create table diops2 as
select cd_operadora, NO_RAZAO, modalidade, dt_diops, Grupo_CC,
(sum(vl_saldo_inicial)) FORMAT=COMMAX20. AS VL_SALDO_INICIAL,
(SUM(VL_SALDO_FINAL)) FORMAT=COMMAX20. AS VL_SALDO_FINAL

from diops1
where n_char = 9
group by cd_operadora, NO_RAZAO, modalidade, dt_diops, Grupo_CC
;
quit;

*****!!!!!!!!!!!!!!!!!!!!!! FILTRO !!!!!!!!!!!!!!!!!!!!!!!*****;
* Exclui operadoras em início de operação;;
PROC SQL;
CREATE TABLE diops3 AS
SELECT t1.*
FROM diops2 t1 left join a1.registro t2 on t1.cd_operadora=t2.cd_operadora
where t1.dt_diops > (t2.dt_inicio)
;
QUIT;

*****!!!!!!!!!!!!!!!!!!!!!! FILTRO !!!!!!!!!!!!!!!!!!!!!!!*****;
* Exclui operadoras com registro cancelado;;
PROC SQL;
CREATE TABLE diops4 AS
SELECT t1.*
FROM diops3 t1 left join a1.CANCELAMENTOS t2 on t1.cd_operadora=t2.cd_operadora
where t2.cd_operadora is missing or t1.dt_diops < (t2.dt_cancelamento)
;
QUIT;

*****!!!!!!!!!!!!!!!!!!!!!! FILTRO !!!!!!!!!!!!!!!!!!!!!!!*****;
* exclui individual novo em autogestão;;
data diops5 xInd_Autogestao;
set diops4;
if modalidade = "Autogestão" and Grupo_CC = "ind_p" then output xInd_Autogestao;
else output diops5;
run;

*****!!!!!!!!!!!!!!!!!!!!!! FILTRO !!!!!!!!!!!!!!!!!!!!!!!*****;
*** Calcula o valor acumulado dos últimos 12 meses a cada trimestre;;

data eventos0;
set diops5;
format vl_trimestre commax20.;
where sum(vl_saldo_final,vl_saldo_inicial) ne 0;
chave = compress(cd_operadora)||" "||strip(grupo_cc);
vl_trimestre = vl_saldo_final - vl_saldo_inicial;
run;

proc sort data=eventos0; by chave dt_diops; run;

data eventos1;
* varrer as datas de cada chave;
do i = 1 by 1 until (last.chave);
set eventos0;
by chave;
format vl_12meses1 vl_12meses2 vl_12meses3 vl_12meses4 vl_12meses commax20.;
retain vl_12meses1 vl_12meses2 vl_12meses3 vl_12meses4 dt_diops0 loop;

* se for primeiro trimestre do tipo, inicializa as variáveis;
if first.chave then do;
vl_12meses4= . ;
vl_12meses3= . ;
vl_12meses2= . ;
vl_12meses1= vl_trimestre;
vl_12meses = vl_trimestre;
dt_diops0 = dt_diops;
loop = 1;
output;
end;

* se for o trimestre seguinte, carrega o valor anterior e calcula o acumulado em 12 meses;
else if ((dt_diops-dt_diops0 = 3) or (substr(put(dt_diops,6.),5,2)='03') and (dt_diops-dt_diops0 = 91))
then do;
vl_12meses4= vl_12meses3;

```

```

vl_12meses3= vl_12meses2;
vl_12meses2= vl_12meses1;
vl_12meses1= vl_trimestre;
vl_12meses = sum(vl_12meses1,vl_12meses2,vl_12meses3,vl_12meses4);
dt_diops0=dt_diops;
loop = loop+1;
output;
end;

* se não for o trimestre seguinte, reinicializa as variáveis;
else do;
vl_12meses4=.;
vl_12meses3=.;
vl_12meses2=.;
vl_12meses1=vl_trimestre;
vl_12meses=vl_trimestre;
dt_diops0=dt_diops;
loop=1;
output;
end;
end;
run;

*****!!!!!!!!!!!!!!!!!!!!!!!!!!!! FILTRO !!!!!!!!!!!!!!!!!!!!!!!!!!!!!*****;
*** se for 4o trimestre, assume o valor do saldo final como sendo o acumulado de 12 meses
*** se não for 4o trimestre, só aceita casos em que a operadora apresentar 4 trimestres seguidos com valores <> 0;;
*** e só aceita os casos em que o acumulado de 12 meses da despesa assistencial total for > 0;
data eventos2;
set eventos1;

where (loop >= 4 and vl_12meses > 0 and vl_12meses ne .)
or (substr(put(dt_diops,6.),5,2)="12" and vl_saldo_final > 0 and vl_saldo_final ne .);

if substr(put(dt_diops,6.),5,2)="12" then vl_12meses = vl_saldo_final;

drop i vl_12meses1 vl_12meses2 vl_12meses3 vl_12meses4 dt_diops0 chave vl_trimestre i loop;
run;

*****
*****
* >>>> RATEIOS;;

* Prepara base para cálculos de rateio;
* Joga grupos de Contas Contábeis em colunas;;
proc sort data=eventos2; by cd_operadora modalidade dt_diops grupo_cc; run;
proc transpose data=eventos2 out=eventos3 (drop=_name_) suffix=_0;
by cd_operadora modalidade dt_diops;
id Grupo_cc;
var vl_12meses;
run;
proc stdize out=eventos3 reponly missing=0; run; * troca vazios por zeros nas variáveis numéricas;

* Prepara tabelas de % por operadora e geral do mercado para rateio;;
PROC SQL;
* a) tabela % por operadora;
create table pct_tt_ops0 as
select CD_OPERADORA, DT_DIOPS, modalidade,
sum(ade_a_pre_0) as ade_a_pre,
SUM(ade_p_pre_0) AS ade_p_pre,
sum(emp_a_pre_0) AS emp_a_pre,
sum(emp_p_pre_0) AS emp_p_pre,
sum(ind_a_0) AS ind_a,
sum(ind_p_0) AS ind_p,
sum(ade_a_pos_0) AS ade_a_pos,
sum(ade_p_pos_0) AS ade_p_pos,
sum(emp_a_pos_0) AS emp_a_pos,
sum(emp_p_pos_0) AS emp_p_pos,
sum(ade_a_pos_0, ade_p_pos_0, emp_a_pos_0, emp_p_pos_0) AS pos_subtt,
sum(ade_a_pos_0, ade_p_pos_0, emp_a_pos_0, emp_p_pos_0, rateio_pos_0) AS pos_tt,
sum(ade_a_pre_0, ade_p_pre_0, emp_a_pre_0, emp_p_pre_0, ind_a_0, ind_p_0) as pre_subtt,
sum(ade_a_pre_0, ade_p_pre_0, emp_a_pre_0, emp_p_pre_0, ind_a_0, ind_p_0, rateio_pre_0) as pre_tt

from eventos3
group by CD_OPERADORA, DT_DIOPS, modalidade;

create table pct_tt_ops as
select CD_OPERADORA, DT_DIOPS, modalidade,
ade_a_pre/ pre_subtt as pct_ade_a_pre_ops format=percent20.1,
ade_p_pre/ pre_subtt as pct_ade_p_pre_ops format=percent20.1,
emp_a_pre/ pre_subtt as pct_emp_a_pre_ops format=percent20.1,
emp_p_pre/ pre_subtt as pct_emp_p_pre_ops format=percent20.1,
ind_a/ pre_subtt as pct_ind_a_ops format=percent20.1,
ind_p/ pre_subtt as pct_ind_p_ops format=percent20.1,
ade_a_pos/ pos_subtt as pct_ade_a_pos_ops format=percent20.1,
ade_p_pos/ pos_subtt as pct_ade_p_pos_ops format=percent20.1,
emp_a_pos/ pos_subtt as pct_emp_a_pos_ops format=percent20.1,
emp_p_pos/ pos_subtt as pct_emp_p_pos_ops format=percent20.1

from pct_tt_ops0
order by dt_diops, modalidade;

* b) tabela % do mercado;
create table pct_tt0 as

```

```

select DT_DIOPS, modalidade,
       sum(ade_a_pre) as ade_a_pre,
       sum(ade_p_pre) AS ade_p_pre,
       sum(emp_a_pre) AS emp_a_pre,
       sum(emp_p_pre) AS emp_p_pre,
       sum(ind_a) AS ind_a,
       sum(ind_p) AS ind_p,
       sum(ade_a_pos) AS ade_a_pos,
       sum(ade_p_pos) AS ade_p_pos,
       sum(emp_a_pos) AS emp_a_pos,
       sum(emp_p_pos) AS emp_p_pos,
       sum(pos_subtt) AS pos_subtt,
       sum(pre_subtt) AS pre_subtt
from pct_tt_ops0
group by DT_DIOPS, modalidade;

create table pct_tt as
select DT_DIOPS, modalidade,
       ade_a_pre/ pre_subtt as pct_ade_a_pre format=percent20.1,
       ade_p_pre/ pre_subtt as pct_ade_p_pre format=percent20.1,
       emp_a_pre/ pre_subtt as pct_emp_a_pre format=percent20.1,
       emp_p_pre/ pre_subtt as pct_emp_p_pre format=percent20.1,
       ind_a/ pre_subtt as pct_ind_a format=percent20.1,
       ind_p/ pre_subtt as pct_ind_p format=percent20.1,
       ade_a_pos/ pos_subtt as pct_ade_a_pos format=percent20.1,
       ade_p_pos/ pos_subtt as pct_ade_p_pos format=percent20.1,
       emp_a_pos/ pos_subtt as pct_emp_a_pos format=percent20.1,
       emp_p_pos/ pos_subtt as pct_emp_p_pos format=percent20.1
from pct_tt0
order by dt_diops, modalidade;

quit;

data pct_tt;
merge pct_tt_ops pct_tt;
by dt_diops modalidade;
run;

* EXECUTA O RATEIO;;
proc sort data=eventos3; by CD_OPERADORA dt_diops modalidade; run;
proc sort data=pct_tt; by CD_OPERADORA dt_diops modalidade; run;
data eventos4;
merge eventos3 (in=a) pct_tt (in=b);
by CD_OPERADORA dt_diops modalidade;

format emp_p_pre emp_a_pre ade_p_pre ade_a_pre ind_p ind_a
       emp_p_pos emp_a_pos ade_p_pos ade_a_pos commmax20.;

* Modalidade de financiamento PRÉ-ESTABELECIDO;
* se não tiver o que ratear, assume o valor como está;
if sum(rateio_pre_0) = 0 then do;
ind_a = ind_a_0;
ind_p = ind_p_0;
ade_a_pre = ade_a_pre_0;
ade_p_pre = ade_p_pre_0;
emp_a_pre = emp_a_pre_0;
emp_p_pre = emp_p_pre_0;
end;

* se tiver o que ratear...;
else do;
* rateia usando o % de todas as operadoras, se não tiver % da própria operadora;
if sum(ade_a_pre_0, ade_p_pre_0, emp_a_pre_0, emp_p_pre_0, ind_a_0, ind_p_0) = 0 then do;
ind_a = (pct_ind_a) * rateio_pre_0;
ind_p = (pct_ind_p) * rateio_pre_0;
ade_a_pre = (pct_ade_a_pre) * rateio_pre_0;
ade_p_pre = (pct_ade_p_pre) * rateio_pre_0;
emp_a_pre = (pct_emp_a_pre) * rateio_pre_0;
emp_p_pre = (pct_emp_p_pre) * rateio_pre_0;
end;

* ou rateia usando o % da própria operadora;
else do;
ind_a = (pct_ind_a_ops) * rateio_pre_0 + ind_a_0;
ind_p = (pct_ind_p_ops) * rateio_pre_0 + ind_p_0;
ade_a_pre = (pct_ade_a_pre_ops) * rateio_pre_0 + ade_a_pre_0;
ade_p_pre = (pct_ade_p_pre_ops) * rateio_pre_0 + ade_p_pre_0;
emp_a_pre = (pct_emp_a_pre_ops) * rateio_pre_0 + emp_a_pre_0;
emp_p_pre = (pct_emp_p_pre_ops) * rateio_pre_0 + emp_p_pre_0;
end;
end;

* Modalidade de financiamento PÓS-ESTABELECIDO;
* se não tiver o que ratear, assume o valor como está;
if sum(rateio_pos_0) = 0 then do;
ade_a_pos = ade_a_pos_0;
ade_p_pos = ade_p_pos_0;
emp_a_pos = emp_a_pos_0;
emp_p_pos = emp_p_pos_0;
end;

* se tiver o que ratear...;
else do;
* rateia usando o % de todas as operadoras, se não tiver % da própria operadora;

```

```

If sum(ade_a_pos_0, ade_p_pos_0, emp_a_pos_0, emp_p_pos_0) = 0 then do;
  ade_a_pos = (pct_ade_a_pos) * rateio_pos_0;
  ade_p_pos = (pct_ade_p_pos) * rateio_pos_0;
  emp_a_pos = (pct_emp_a_pos) * rateio_pos_0;
  emp_p_pos = (pct_emp_p_pos) * rateio_pos_0;
end;

* ou rateia usando o % da própria operadora;
else do;
  ade_a_pos = (pct_ade_a_pos_ops) * rateio_pos_0 + ade_a_pos_0;
  ade_p_pos = (pct_ade_p_pos_ops) * rateio_pos_0 + ade_p_pos_0;
  emp_a_pos = (pct_emp_a_pos_ops) * rateio_pos_0 + emp_a_pos_0;
  emp_p_pos = (pct_emp_p_pos_ops) * rateio_pos_0 + emp_p_pos_0;
end;

end;

keep cd_operadora dt_diops
emp_p_pre emp_a_pre ade_p_pre ade_a_pre ind_p ind_a emp_p_pos emp_a_pos ade_p_pos ade_a_pos;
run;

* volta os grupos de contas para linha;;
proc sort data=eventos4; by cd_operadora dt_diops; run;
proc transpose data=eventos4 out=eventos5 (rename=(coll=v1_12meses _name_=Grupo_CC));
  by cd_operadora dt_diops;
run;
data eventos5;
set eventos5;
  where v1_12meses ne 0 and v1_12meses ne .;
run;
* FIM DO RATEIO;;
*****;

* Após o rateio, agrupa as subcategorizações dos planos antigos entre pré e pós e cria campos de característica dos planos;;
proc sql;
create table eventos6 as

select t1.cd_operadora,
  case
    when t1.grupo_cc contains '_a_pre' then substr(t1.grupo_cc,1,5)
    when t1.grupo_cc contains '_a_pos' then substr(t1.grupo_cc,1,5)
    else t1.grupo_cc
  end as tipo,
  t1.dt_diops,
  sum(t1.v1_12meses) format=commmax20. as v1_12meses

from eventos5 t1
  group by t1.cd_operadora,
  calculated tipo,
  t1.dt_diops;
quit;

data a1.eventos_12m (rename=(dt_diops=trimestre));
set eventos6;
format de_tipo_contratacao $70. de_tipo_modalidade_finn $30.;

if tipo in ("ind_a", "ade_a", "emp_a") then TP_vigencia_plano = "A";
  else TP_vigencia_plano = "P";

if tipo in ("ind_a", "ind_p") then de_tipo_contratacao = "Individual ou familiar";
else if tipo in ("ade_a", "ade_p_pre", "ade_p_pos") then de_tipo_contratacao = "Coletivo por adesão";
else if tipo in ("emp_a", "emp_p_pre", "emp_p_pos") then de_tipo_contratacao = "Coletivo empresarial";
  else de_tipo_contratacao = "ERRO";

If tipo in ("ind_a", "ind_p", "ade_p_pre", "emp_p_pre") then de_tipo_modalidade_finn = "Pre-estabelecido";
else if tipo in ("ade_p_pos", "emp_p_pos") then de_tipo_modalidade_finn = "Pós-estabelecido";
else if tipo in ("ade_a", "emp_a") then de_tipo_modalidade_finn = "Plano antigo";
  else de_tipo_modalidade_finn = "ERRO";

run;

```