

Ministério da Saúde
Secretaria de Vigilância em Saúde
Departamento de Vigilância de Doenças e Agravos não Transmissíveis e
Promoção da Saúde DANTPS

Produto Número 01

Relatório que descreve o resultado de um exercício de profile matching entre o
SIM (Sistema de Informação sobre Mortalidade) e o SINASC (Sistema de
Informações sobre Nascidos Vivos) visando identificar aqueles que foram ao
óbito com microcefalia entre as causas de morte mas onde a mesma condição não
foi assinalada no registro SINASC correspondente.

Antony Peter Stevens
Contrato OPAS
Número SCON2016–02180
Distrito Federal – Brasília

13 de junho de 2016

Sumário

1	Introdução	3
2	Rotinas.	6
3	O Pareamento.	11
4	Comentários sobre os dados.	16
5	Mapas.	19
6	As linhas de causas nas declarações de óbito.	28
7	Sobrevida.	34
8	Os próximos passos.	35
	Referências Bibliográficas	37
9	Copyright Notice	39

Lista de Figuras

5.1	Municípios onde encontramos registros de óbito onde consta Q02, independente do registro ser ou não pareado.	20
5.2	Clusters dos municípios onde encontramos registros de óbito onde consta Q02, independente do registro ser ou não pareado.	21
5.3	Municípios onde encontramos registros de nascimento onde consta Q02, independente do registro ser ou não pareado.	22
5.4	Clusters dos municípios onde encontramos registros de nascimento onde consta Q02, independente do registro ser ou não pareado. . .	23
5.5	Municípios onde encontramos registros de nascimento pareados através de <i>profile matching</i> com registros de óbito onde consta Q02 em ambos.	24
5.6	Clusters dos municípios onde encontramos registros de nascimento pareados através de <i>profile matching</i> com registros de óbito onde consta Q02 em ambos.	25
5.7	Municípios onde encontramos registros de óbito que não foram pareados através de <i>profile matching</i> com registros de nascimento onde consta Q02 em ambos.	26
5.8	Municípios onde encontramos registros de nascimento que não foram pareados através de <i>profile matching</i> com registros de óbito onde consta Q02 em ambos.	27

Capítulo 1

Introdução

Ao estudar uma referência sobre malformações congênitas [Jones et al., 2013] descobrimos que não existe um diagnóstico específico “microcefalia” visto que uma malformação é definida através de uma associação de anomalias das quais a microcefalia pode ou não fazer parte. Em certos casos ela pode fazer parte de um diagnóstico embora o diagnóstico pode prevalecer mesmo na sua ausência [Ventura et al., 2016].

Grande parte das malformações são consequências a nível de cromossomo no momento da fertilização ou logo depois. Observamos na Tabela 1.1 10 das 52 malformações onde a microcefalia pode ser um fator principal no diagnóstico. Em outros diagnósticos ele pode ser um fator secundário.

Acredita-se que a microcefalia que hoje recebe atenção seja consequência de um evento durante a gestação. Imaginamos que estudos clínicos irão identificar a janela durante a gestação quando uma infecção do vírus da Zika pode desencadear a sequência de anomalias porque o conhecimento da embriologia pode identificar os momentos onde os tecidos afetados começam a se desenvolver.

Porém em registros de nascimento e óbito encontramos códigos onde a microcefalia é registrada. No caso do registro de nascimento o código pode fazer parte de uma coleção de códigos usados para descrever anomalias observadas no recém-nascido. No caso do óbito a condição pode aparecer como causa básica ou nas outras linhas que juntas apoiam a definição da causa básica.

Neste relatório apresentamos resultados de estudo dos registros de 1996 – 2013 de nascimento e óbito. Estes registros estão disponíveis no domínio público.

Visamos responder uma pergunta simples:

“Se microcefalia foi identificada no óbito ela teria sido identificada também no nascimento?”

Mas o leitor pode perguntar:

“Se os dados foram anonimizados como podemos associar um óbito a um nascimento?”

A resposta é que quando os eventos são poucos as combinações das variáveis secundárias tendem a ser únicas. Suponhamos que o município de residência no óbito seja igual ao município de nascimento no registro de nascimento. Não teremos muitas combinações município \times data de nascimento e quando uma combinação só é encontrada uma vez em cada sistema podemos, numa primeira aproximação, acreditar que se trata da mesma pessoa sem saber o nome dela.

O resultado principal deste relatório mostra um desencontro entre os dois sistemas. São poucos os registros de nascimento que assinalam microcefalia onde o mesmo fato foi assinalado no óbito. E vice-versa.

Aproveitamos também para apresentar:

1. Mapas que mostram onde os registros são encontrados, diferenciando os pareados e os não-pareados.
2. Resultados do uso do Software SaTScan™[Kulldorff and Inc,] que pode ser usado para sugerir “clusters” de casos distribuídos geograficamente.

Angelman
Aniridia–Wilms Tumor Association
Autosomal Recessive Chondrodysplasia Punctata
Bloom
Börjeson–Forssman–Lehmann
...
Trisomy 13
Trisomy 18
Williams
X–Linked α – Thalassemia/Mental Retardation
Yunis–Varón

Tabela 1.1: Exemplos de malformações onde microcefalia é encontrada com frequência.

Capítulo 2

Rotinas.

Seleção dos óbitos.

```
def other_lines(a,b,c,d,ii):
    yes_no = '0'
    if a[:3] == "Q02":
        yes_no = '1'
    if b[:3] == "Q02":
        yes_no = '1'
    if c[:3] == "Q02":
        yes_no = '1'
    if d[:3] == "Q02":
        yes_no = '1'
    if ii[:3] == "Q02":
        yes_no = '1'
    return yes_no

#
# SIM
#
print "SIM"
stem = '/home/antony/Desktop/'
stem = stem + 'SIM_FROM_DATASUS_MAY_2015/'
#
f = open('../data/sim_Q02_dict', 'w' )
g = open('../data/sim_not_Q02_dict', 'w' )
#
cid10_uf = [ 'ro', 'ac', 'am', 'rr', 'pa', 'ap', 'to',
              'ma', 'pi', 'ce', 'rn', 'pb', 'pe', 'al', 'se', 'ba',
              'mg', 'es', 'rj', 'sp',
              'pr', 'sc', 'rs',
              'ms', 'mt', 'go', 'df' ]
#
for yr in cid10_years:
    print yr,
    for uf in cid10_uf:
        dbf1 = dbf.Dbfc( stem + yr + '/do' + uf + yr + '.dbf')
        for rlocal in range(min(10000000,len(dbf1))):
            rec = dbf1[rlocal]
```

```

cb = rec[ 'CAUSABAS' ][ :3 ]
a  = rec[ 'LINHAA' ]
b  = rec[ 'LINHAB' ]
c  = rec[ 'LINHAC' ]
d  = rec[ 'LINHAD' ]
ii = rec[ 'LINHAII' ]
#
dt_nasc = str( rec[ 'DINASC' ] ).replace( "\00", "-" )
dt_obit = str( rec[ 'DTOBITO' ] ).replace( "\00", "-" )
sx = str( rec[ 'SEXO' ] ).replace( "\00", "-" )
if sx != '1' and sx != '2':
    sx = 'X'
mres = calcula_dv_municipio( str( rec[ 'CODMUNRES' ] ).replace( "\00", "-" ) )
s = dt_nasc + ',' + dt_obit + ',' + sx + ',' + mres[0:6] + ','
s = s + cb + ',' + a + ',' + b + ',' + c + ','
s = s + d + ',' + ii + '\n'
#
iuse = False
if cb == "Q02":
    iuse = True
if other_lines(a,b,c,d,ii) == '1':
    iuse = True
if iuse:
    f.write(s)
else:
    g.write(s)
f.close()
g.close()

```

Seleção dos nascimentos.

```

#
# SINASC
#
cid10_years = [ '2000',
                '2001', '2002', '2003', '2004', '2005',
                '2006', '2007', '2008', '2009', '2010',
                '2011', '2012', '2013' ]

print "SINASC"
f = open( '../data/sinasc_Q02_dict', 'w' )
g = open( '../data/sinasc_not_Q02_dict', 'w' )
stem = '/home/antony/Desktop/'
stem = stem + 'SINASC_FROM_DATASUS_JULHO_2015/'
cid10_uf = [ 'RO', 'AC', 'AM', 'RR', 'PA', 'AP', 'TO',
            'MA', 'PI', 'CE', 'RN', 'PB', 'PE', 'AL', 'SE', 'BA',
            'MG', 'ES', 'RJ', 'SP',
            'PR', 'SC', 'RS',
            'MS', 'MT', 'GO', 'DF' ]

for yr in cid10_years:
    print yr,
    for uf in cid10_uf:
        dbf1 = dbf.Dbf( stem + yr + '/DN' + uf + yr + '.dbf' )
        for rlocal in range( min(10000000, len( dbf1 )) ):
            rec = dbf1[ rlocal ]
            #
            c = str( rec[ "CODANOMAL" ] )

```



```

#
dn = str(rec['DTNASC']).replace("\00","-")
sx = str(rec["SEXO"])
if sx != '1' and sx != '2':
    sx = 'X'
mres = calcula_dv_municipio(str(rec['CODMUNRES']).replace("\00","-"))
s = dn + ',' + sx + ',' + mres[0:6] + ',' + c + '\n'
#
if 'Q02' in c:
    f.write(s)
else:
    g.write(s)
f.close()
g.close()

```

Ano do óbito daqueles que nasceram após 1995.

```

sim_Q02 = pd.read_csv('../data/sim_Q02_dict',
                      names=['nasc','obit','sx','m','cb','a','b','c','d','ii'],
                      low_memory=False, dtype={'nasc':'S8','obit':'S8'})
sim_Q02.fillna('99999999', inplace=True)
sim_Q02['yearnasc'] = sim_Q02['nasc'].apply(lambda x: x[4:8])
sim_Q02['yearobit'] = sim_Q02['obit'].apply(lambda x: x[4:8])
temp = sim_Q02.loc[sim_Q02['yearnasc'] != '']
sim_Q02_nasc_after_1995 = temp.loc[temp['yearnasc'].apply(int) > 1995]
sim_Q02_nasc_after_1995.groupby(['yearobit']).size()

```

Ano do nascimento daqueles óbitos que nasceram após 1995.

```

sim_Q02_nasc_after_1995.groupby(['yearnasc']).size()

```

Preparação para SaTscan™ das frequências por município dos casos que encontram em cada sistema (SIM e SINASC).

```

x = sinasc_Q02.groupby(['m']).size().reset_index(name='count')
f = open('../SatScan/Q02AllSINASC/Q02_all_SINASC','w')
for idn, row in x.iterrows():
    if str(row['m'])[2:6] != '0000':
        if str(row['m'])[0:3] != '334':
            if str(row['m'])[0:6] != '330285' and str(row['m'])[0:6] != '320225':
                f.write(str(row['m']) + '\t' + str(row['count']) + '\n')
f.close()
x = sim_Q02_nasc_after_1995.groupby(['m']).size().reset_index(name='count')
f = open('../SatScan/Q02AllSIM/Q02_all_SIM','w')
for idn, row in x.iterrows():
    if str(row['m'])[2:6] != '0000':
        if str(row['m'])[0:3] != '334':
            f.write(str(row['m']) + '\t' + str(row['count']) + '\n')
f.close()

```

Preparação para SaTscan™ das frequências por município dos casos que se encontram em ambos dos sistemas (SIM e SINASC), os “pareados”.

```
#
# SIM and SINASC
#
both_Q02_dict = {}
temp_sinasc = sinasc_Q02
temp_sim = sim_Q02_nasc_after_1995
count = 0
for idn_sinasc, row_sinasc in temp_sinasc.iterrows():
    for idn_sim, row_sim in temp_sim.iterrows():
        if str(row_sinasc['m']) == str(row_sim['m']) and str(row_sinasc['nasc'])
== str(row_sim['nasc']):
            count += 1
            try:
                both_Q02_dict[row_sinasc['m']] += 1
            except:
                #print row_sinasc['m']
                both_Q02_dict[row_sinasc['m']] = 1
            #print row_sinasc['m'], row_sim['m'], row_sinasc['nasc'], row_sim['nasc'], count
print count
f = open('../SatScan/Q02Both/Q02_both', 'w')
for x in both_Q02_dict:
    #print x, both_Q02_dict[x]
    f.write( str( x ) + '\t' + str(both_Q02_dict[x]) + '\n' )
f.close()
```

Preparação para SaTscan™ das frequências por município dos casos que se encontram no SINASC mas não no SIM.

```
#
# SINASC but not SIM
#
count = 0
sinasc_not_sim_Q02_dict = {}
temp_sinasc = sinasc_Q02
temp_sim = sim_Q02_nasc_after_1995
count = 0
for idn_sinasc, row_sinasc in temp_sinasc.iterrows():
    found_match = 0
    for idn_sim, row_sim in temp_sim.iterrows():
        if str(row_sinasc['m']) == str(row_sim['m']) and str(row_sinasc['nasc'])
== str(row_sim['nasc']):
            found_match = 1
    if found_match == 0:
        count += 1
        #print row_sinasc['m'], row_sinasc['nasc'], count
        try:
            sinasc_not_sim_Q02_dict[row_sinasc['m']] += 1
        except:
            #print row_sinasc['m']
```

```

sinasc_not_sim_Q02_dict[row_sinasc['m']] = 1
print count
f = open('.../SatScan/Q02_SINASC_but_not_SIM/Q02SINASCbutnotSIM', 'w')
for x in sinasc_not_sim_Q02_dict:
    if x != 320225 and x != 330285 and x != 510325:
        #print x, sinasc_not_sim_Q02_dict[x]
        f.write( str( x ) + '\t' + str(sinasc_not_sim_Q02_dict[x]) + '\n' )
f.close()

```

Preparação para SaTscan™ das frequências por município dos casos que se encontram no SIM mas não no SINASC.

```

#
# SIM but no SINASC
#
count = 0
sim_not_sinasc_Q02_dict = {}
temp_sim = sim_Q02_nasc_after_1995
temp_sinasc = sinasc_Q02
count = 0
for idn_sinasc, row_sim in temp_sim.iterrows():
    found_match = 0
    for idn_sim, row_sinasc in temp_sinasc.iterrows():
        if str(row_sinasc['m']) == str(row_sim['m']) and str(row_sinasc['nasc'])
== str(row_sim['nasc']):
            found_match = 1
    if found_match == 0:
        count += 1
        #print row_sinasc['m'], row_sinasc['nasc'], count
        try:
            sim_not_sinasc_Q02_dict[row_sim['m']] += 1
        except:
            #print row_sinasc['m']
            sim_not_sinasc_Q02_dict[row_sim['m']] = 1
print count
f = open('.../SatScan/Q02_SIM_but_not_SINASC/Q02SIMbutnotSINASC', 'w')
for x in sim_not_sinasc_Q02_dict:
    if x != 320225 and x != 330285 and x != 510325:
        if str( x )[2:6] != '0000':
            if str( x )[0:3] != '334':
                if str( x )[0:6] != '330285' and str( x )[0:6] != '320225':
                    f.write( str( x ) + '\t' + str(sim_not_sinasc_Q02_dict[x]) + '\n' )
f.close()

```

Capítulo 3

O Pareamento.

Rotina do pareamento.

```
#
# No SIM born after 1999
#
count = 0
i_have_seen = {}
x = sim_Q02_nasc_after_1999
f = open('./test_counts/test','w')
for idn, row in x.iterrows():
    if str(row['m'])[2:6] != '0000':
        if str(row['m'])[0:3] != '334':
            count += 1
            k = str(row['m']) + '_' + str(row['nasc'])
            f.write( k + '\n' )
            try:
                i_have_seen[ k ] += 1
            except:
                i_have_seen[k] = 1
f.close()
print 'No_SIM_born_after_1999'
print count
print len( i_have_seen )
for x in i_have_seen:
    if i_have_seen[x] > 1:
        print x, i_have_seen[x]
#
# No SINASC born after 1995
#
count = 0
i_have_seen = {}
x = sinasc_Q02
f = open('./test_counts/test','w')
for idn, row in x.iterrows():
    if str(row['m'])[2:6] != '0000':
        if str(row['m'])[0:3] != '334':
```

```

        count += 1
        k = str(row['m']) + '_' + str(row['nasc'])
        f.write( k + '\n' )
    try:
        i_have_seen[ k ] += 1
    except:
        i_have_seen[k] = 1
f.close()
print 'No_SINASC_born_after_1995'
print count
print len( i_have_seen )
for x in i_have_seen:
    if i_have_seen[x] > 1:
        print x, i_have_seen[x]
#
# No SINASC born after 1995 paired with SIM born after 1995
#
count = 0
i_have_seen = {}
x = sinasc_Q02
f = open('./test_counts/test', 'w')
temp_sinasc = sinasc_Q02
temp_sim = sim_Q02_nasc_after_1999
count = 0
for idn_sinasc, row_sinasc in temp_sinasc.iterrows():
    for idn_sim, row_sim in temp_sim.iterrows():
        if str(row_sinasc['m']) == str(row_sim['m']) and str(row_sinasc['nasc'])
== str(row_sim['nasc']):
            count += 1
            k = str(row_sinasc['m']) + '_' + str(row_sinasc['nasc'])
            f.write( k + '\n' )
        try:
            i_have_seen[ k ] += 1
        except:
            i_have_seen[k] = 1
f.close()
print 'No_SINASC_born_after_1995_paired_with_SIM_born_after_1999'
print count
print len( i_have_seen )
for x in i_have_seen:
    if i_have_seen[x] > 1:
        print x, i_have_seen[x]

```

Resultado do pareamento.

```

No SIM born after 1999
1001
999
293270_12092001 2
421145_10032007 2
No SINASC born after 1995
2303

```

```
2294
330380_25102013 2
355030_28032013 2
330455_27122006 2
316550_15032005 2
290210_06032008 2
316720_26042008 2
500060_01032006 2
420260_27112001 2
355030_19022012 2
No SINASC born after 1995 paired with SIM born after 1999
208
205
500060_01032006 2
355030_19022012 2
421145_10032007 2
```

Fazemos a seguinte leitura do resultado:

1. –

```
No SIM born after 1999
1001
999
293270_12092001 2
421145_10032007 2
```

Encontramos 1001 combinações de data de nascimento/município entre os registros do SIM sendo que as combinações 293270×12092001 e 421145×10032007 foram cada uma encontrada duas vezes.

2. –

```
No SINASC born after 1995
2303
2294
330380_25102013 2
```

355030_28032013 2
330455_27122006 2
316550_15032005 2
290210_06032008 2
316720_26042008 2
500060_01032006 2
420260_27112001 2
355030_19022012 2

Encontramos 2303 combinações de data de nascimento/município entre os registros do SIM sendo que as combinações

330380 x 25102013
355030 x 28032013
330455 x 27122006
316550 x 15032005
290210 x 06032008
316720 x 26042008
500060 x 01032006
420260 x 27112001
355030 x 19022012

foram cada uma encontrada duas vezes.

3. –

No SINASC born after 1995 paired with SIM born after 1999
208
205
500060_01032006 2
355030_19022012 2
421145_10032007 2

Encontramos 208 combinações de data de nascimento/município entre os registros do SIM com as mesmas combinações entre os registros do SINASC sendo que as combinações

CAPÍTULO 3. O PAREAMENTO.

500060 x 01032006

355030 x 19022012

421145 x 10032007

foram cada uma encontrada duas vezes.

Capítulo 4

Comentários sobre os dados.

Anos em que os eventos se encontram no SIM.

Entre os “ óbitos Q02” que se encontram no SIM onde o ano de nascimento foi depois de 1995, quando em tese poderíamos encontrar referência ao mesmo código da Classificação Internacional de Doenças no SINASC, podemos fazer uma tabela de frequências. Ela se encontra na Tabela 4.1. Observamos uma queda nos anos mais recentes.

Ano de Nascimento do Óbito Q02 no SIM	Frequência
1996	84
1997	72
1998	77
1999	90
2000	99
2001	95
2002	83
2003	72
2004	69
2005	88
2006	88
2007	70
2008	84
2009	68
2010	49
2011	62
2012	50
2013	26

Tabela 4.1:

Anos em que os eventos se encontram no SINASC.

Não encontramos registros de nascimento com Q02 antes do ano 2000 o que sugere que embora o código estava disponível no sistema ele ainda não estava sendo usado.

CAPÍTULO 4. COMENTÁRIOS SOBRE OS DADOS.

Ano de nascimento de registros com Q02 no campo CODANOMAL	Frequência
2000	175
2001	170
2002	178
2003	172
2004	173
2005	164
2006	155
2007	135
2008	150
2009	141
2010	163
2011	154
2012	190
2013	183

Tabela 4.2:

Capítulo 5

Mapas.

A distribuição geográfica dos registros que contém o código para microcefalia (Q02) foi obtida usando o software SaTscan™[Kulldorff and Inc,]. Entre as várias opções oferecidas escolhemos as duas mais simples:

1. O modelo Poisson simples que assinala aqueles municípios onde encontramos mais casos do que esperado. Na prática isso é quase igual aos municípios onde encontramos casos visto que os municípios são muitos e os casos são poucos.
2. Aproveitamos a opção do software de formar “clusters” dos municípios encontrados no modelo Poisson, se eles existem.

Casos encontrados no SIM.

Podemos observar nas Figuras 5.1 e 5.2 que os registros do SIM com Q02 se encontram principalmente no Centroeste, Norte e no Sudoeste.

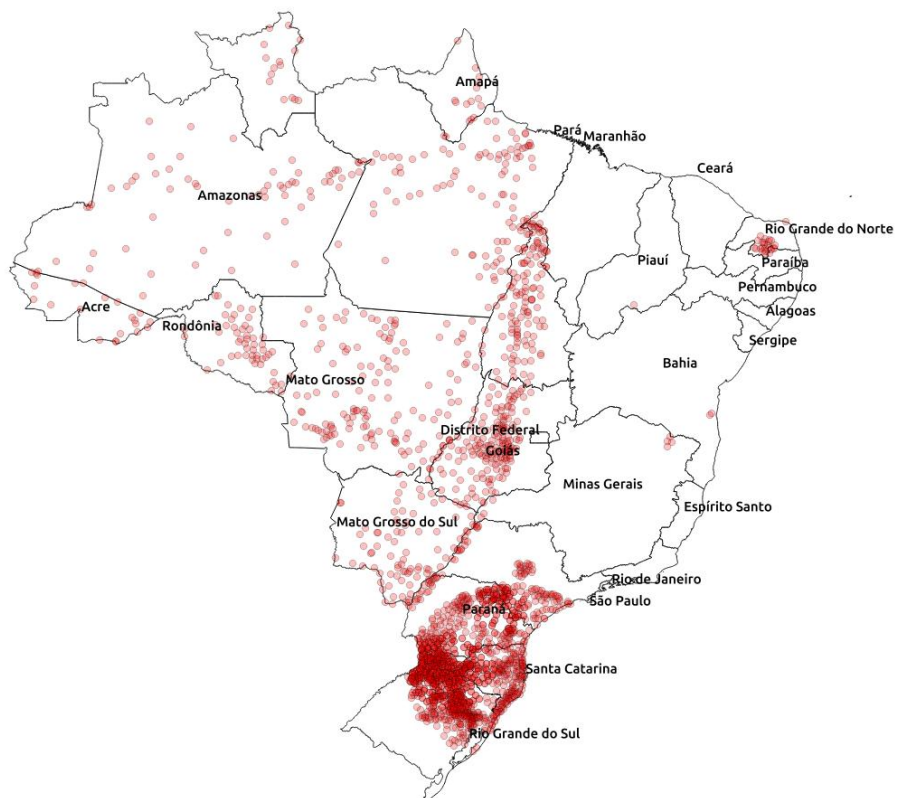


Figura 5.1: Municípios onde encontramos registros de óbito onde consta Q02, independente do registro ser ou não pareado.

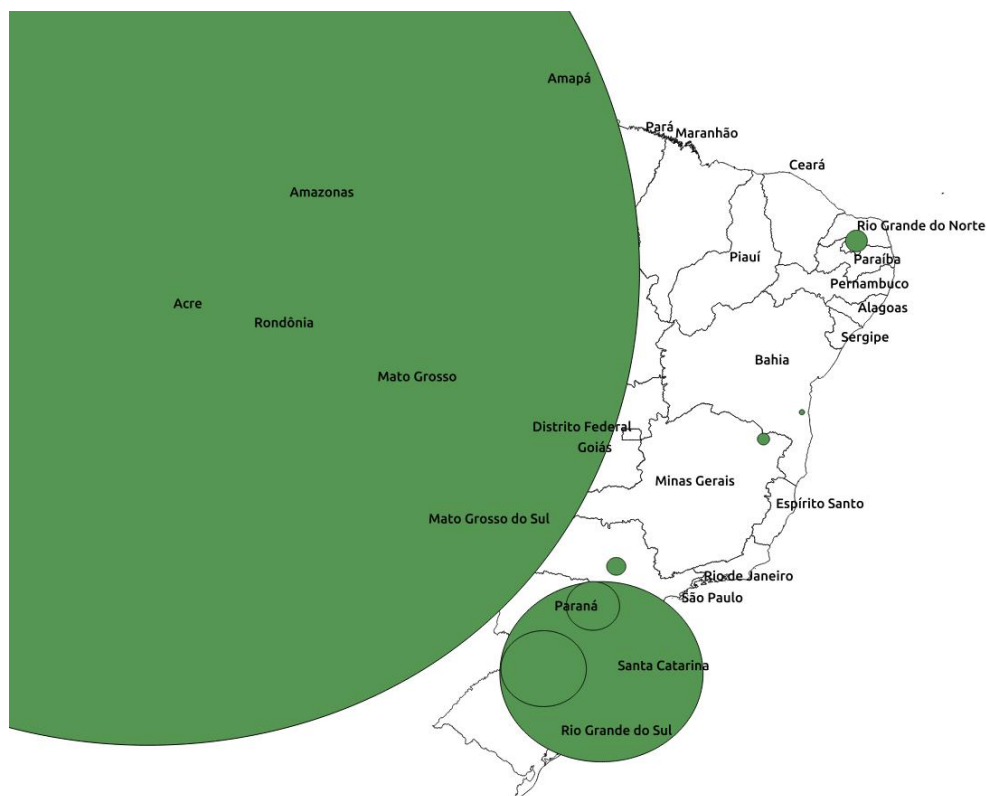


Figura 5.2: Clusters dos municípios onde encontramos registros de óbito onde consta Q02, independente do registro ser ou não pareado.

Casos encontrados no SINASC.

Podemos observar nas Figuras 5.3 e 5.4 que os registros do SINASC com Q02 se encontram principalmente no Norte e no Nordeste.

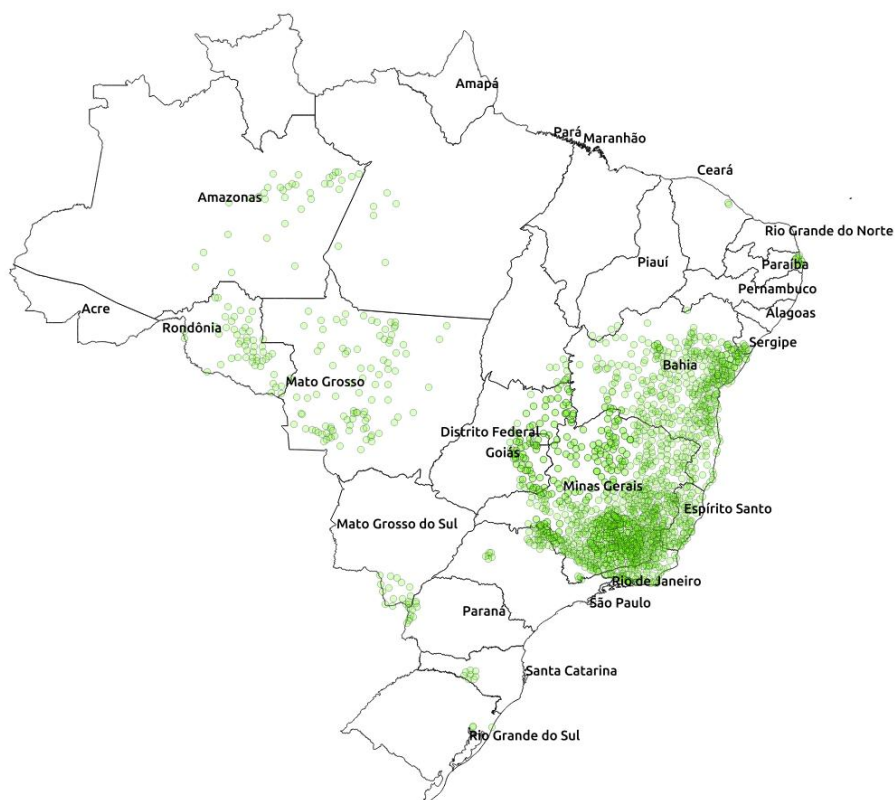


Figura 5.3: Municípios onde encontramos registros de nascimento onde consta Q02, independente do registro ser ou não pareado.

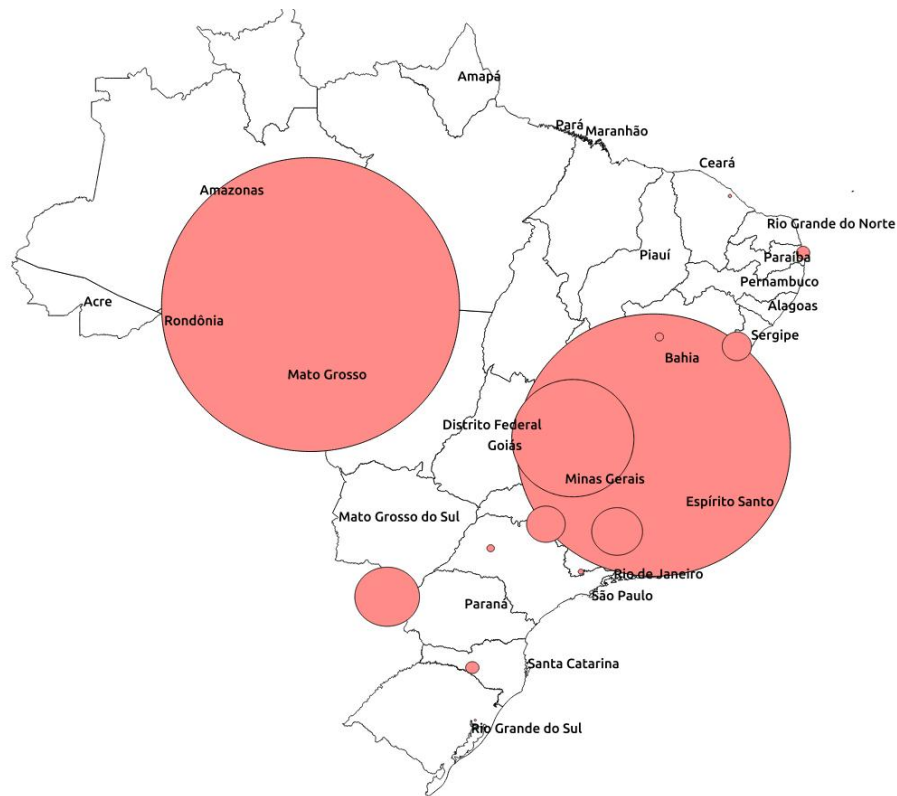


Figura 5.4: Clusters dos municípios onde encontramos registros de nascimento onde consta Q02, independente do registro ser ou não pareado.

Casos encontrados no SIM e no SINASC.

Podemos observar nas Figuras 5.5 e 5.6 que os registros do SINASC com Q02 pareados com registros do SIM com Q02 se encontram principalmente no Sudoeste com um pequeno cluster no Rio de Janeiro.



Figura 5.5: Municípios onde encontramos registros de nascimento pareados através de *profile matching* com registros de óbito onde consta Q02 em ambos.



Figura 5.6: Clusters dos municípios onde encontramos registros de nascimento pareados através de *profile matching* com registros de óbito onde consta Q02 em ambos.

Casos não pareados.

Embora o pareamento encontrou alguns pares na bacia amazonica existem muitos óbitos onde a condição não foi reconhecida no nascimento. No caso do nascimentos com Q02 encontramos um grupo grande em Minas Gerais que não foi pareado com óbitos com a mesma condição.

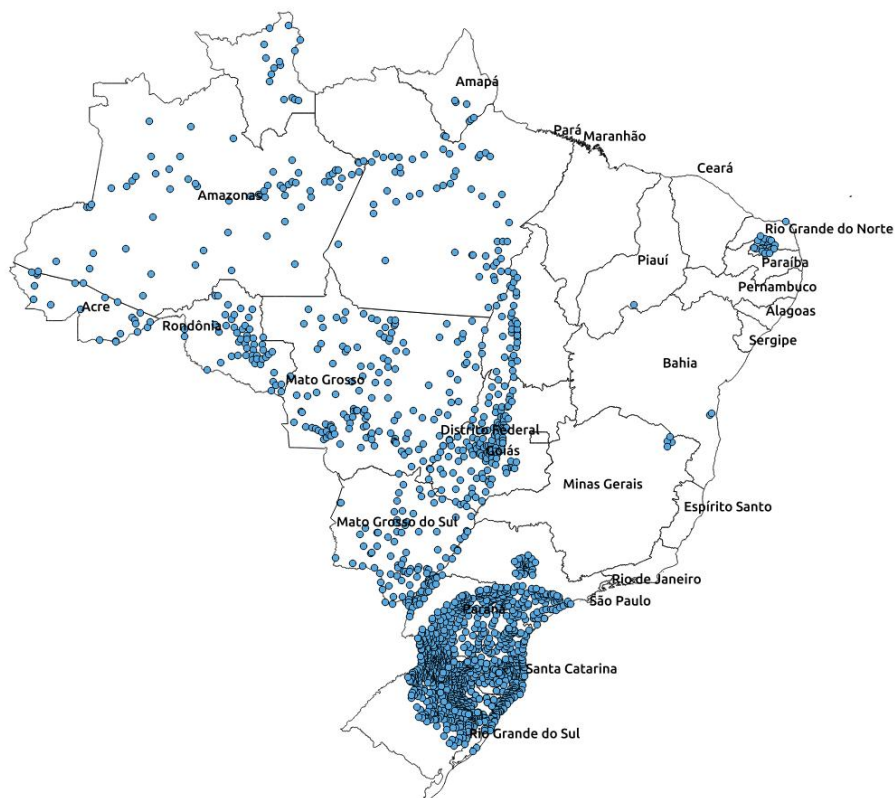


Figura 5.7: Municípios onde encontramos registros de óbito que não foram pareados através de *profile matching* com registros de nascimento onde consta Q02 em ambos.

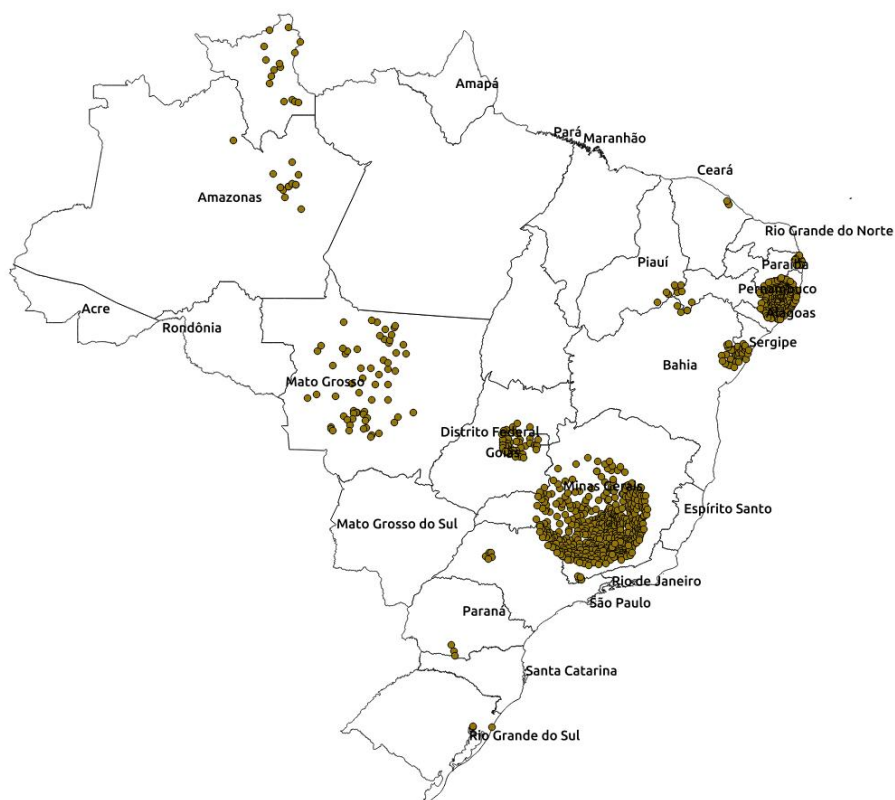


Figura 5.8: Municípios onde encontramos registros de nascimento que não foram pareados através de *profile matching* com registros de óbito onde consta Q02 em ambos.

Capítulo 6

As linhas de causas nas declarações de óbito.

Mostramos aqui os mais frequentes códigos encontrados nas diversas linhas da declaração de óbito usadas neste estudo.

Linha CAUSABAS.

Percebemos que só encontramos Q02 entre os registros escolhidos. Isso sugere a possibilidade da obrigatoriedade de usar o código nesta linha quando ele é mencionado em qualquer outra das linhas.

Frequência	
CAUSABAS	
Q02	1326

Tabela 6.1:

Outras Linhas.

LINHA A

Frequência	
Linha A	
...	...
J189	23
R99X	24
J180	32
P291	41
J960	64
A419	84
J969	91
R092	98
P285	135
Q02X	180
branco	203

Tabela 6.2:

CAPÍTULO 6. AS LINHAS DE CAUSAS NAS DECLARAÇÕES DE ÓBITO.

Linha B.

Frequência	
Linha B	
...	...
Q899	18
J690	21
P369	22
R568	23
G809	24
J180	29
P285	31
A419	57
J189	65
Q02X	323
branco	347

Tabela 6.3:

CAPÍTULO 6. AS LINHAS DE CAUSAS NAS DECLARAÇÕES DE ÓBITO.

Linha C.

Frequência	
Linha B	
...	...
Q897	11
P071	11
E46X	11
P073	14
A419	15
J180	17
G809	19
Q899	27
J189	34
Q02X	271
branco	653

Tabela 6.4:

CAPÍTULO 6. AS LINHAS DE CAUSAS NAS DECLARAÇÕES DE ÓBITO.

Linha D.

Frequência	
Linha B	
...	...
Q043	3
E46X	3
P219	5
G809	5
Q019	6
P073	7
Q897	10
Q899	12
J189	13
Q02X	167
branco	973

Tabela 6.5:

CAPÍTULO 6. AS LINHAS DE CAUSAS NAS DECLARAÇÕES DE ÓBITO.

Linha II.

	Frequência
Linha B	
...	...
R568	3
Q897	3
Q02X*N179	4
Q02X*Q019	4
Q02X*Q059	4
E46X	6
Q899	6
P073	7
Q02X	66
branco	986

Tabela 6.6:

Parecem que são sempre os mesmos códigos.

Capítulo 7

A sobrevida dos óbitos onde microcefalia é registrado.

Se comparamos o ano do óbito com o ano do nascimento daqueles registros onde consta Q02 observamos que a maior parte não chega ao segundo ano de vida. Observamos que entre os 95 casos que nasceram em 2001, 62 já tinham falecido em 2003.

yearobit	1996	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	All
yeamasc																			
1996	38	9	5	4	4	2	0	1	2	1	2	1	1	2	0	6	2	4	84
1997	0	29	14	4	3	2	2	2	1	1	0	4	2	2	2	1	2	1	72
1998	0	0	38	11	9	1	1	0	1	2	1	1	2	3	3	1	2	1	77
1999	0	0	0	44	12	6	2	2	2	3	1	3	2	4	3	2	3	1	90
2000	0	0	0	0	60	15	4	4	4	2	2	1	0	3	0	1	1	2	99
2001	0	0	0	0	0	50	12	2	9	3	1	5	1	2	2	2	1	5	95
2002	0	0	0	0	0	0	50	19	3	1	2	0	2	1	1	2	0	2	83
2003	0	0	0	0	0	0	0	32	23	5	1	1	0	4	1	2	1	2	72
2004	0	0	0	0	0	0	0	0	38	8	8	2	2	3	2	2	1	3	69
2005	0	0	0	0	0	0	0	0	0	41	15	9	6	5	3	0	6	3	88
2006	0	0	0	0	0	0	0	0	0	0	50	18	3	6	2	2	3	4	88
2007	0	0	0	0	0	0	0	0	0	0	0	43	15	4	2	2	1	3	70
2008	0	0	0	0	0	0	0	0	0	0	0	0	42	23	5	8	4	2	84
2009	0	0	0	0	0	0	0	0	0	0	0	0	0	44	10	7	5	2	68
2010	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	11	9	1	49
2011	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	40	11	11	62
2012	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	34	16	50
2013	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	26	26
All	38	38	57	63	88	76	71	62	83	67	83	88	78	106	64	89	86	89	1326

Tabela 7.1: As colunas representam o ano do óbito. As linhas aquele do nascimento.

Capítulo 8

Os próximos passos.

Os dados aqui apresentados precisam ser compartilhados com:

1. Aqueles que estudam malformações congênitas no Brasil.
2. Aqueles que são responsáveis pelo preenchimento dos registros vitais.

É possível que nessas comunidades encontraremos explicações para as tendências aqui descritas.

Referências Bibliográficas

- [Cech et al., 2007] Cech, I., Burau, K. D., and Walston, J. (2007). Spatial distribution of orofacial cleft defect births in Harris County, Texas, 1990 to 1994, and historical evidence for the presence of low-level radioactivity in tap water. *South. Med. J.*, 100(6):560–569.
- [Dipierri et al., 2014] Dipierri, J., Rodriguez-Larralde, A., Barrai, I., Camelo, J. L., Redomero, E. G., Rodriguez, C. A., Ramallo, V., Bronberg, R., and Alfaro, E. (2014). Random inbreeding, isonymy, and population isolates in Argentina. *J Community Genet*, 5(3):241–248.
- [Guerrini and Filippi, 2005] Guerrini, R. and Filippi, T. (2005). Neuronal migration disorders, genetics, and epileptogenesis. *J. Child Neurol.*, 20(4):287–299.
- [Guerrini and Marini, 2006] Guerrini, R. and Marini, C. (2006). Genetic malformations of cortical development. *Exp Brain Res*, 173(2):322–333.
- [Jones et al., 2013] Jones, K., Jones, M. C., and del Campo, M. (2013). *Smith’s Recognizable Patterns of Human Malformation*.
- [Jung et al., 2007] Jung, I., Kulldorff, M., and Klassen, A. C. (2007). A spatial scan statistic for ordinal data. *Stat Med*, 26(7):1594–1607.
- [Jung et al., 2010] Jung, I., Kulldorff, M., and Richard, O. J. (2010). A spatial scan statistic for multinomial data. *Stat Med*, 29(18):1910–1918.
- [Kulldorff et al., 2005] Kulldorff, M., Heffernan, R., Hartman, J., Assuncao, R., and Mostashari, F. (2005). A space-time permutation scan statistic for disease outbreak detection. *PLoS Med.*, 2(3):e59.
- [Kulldorff and Inc,] Kulldorff, M. and Inc, I. M. S. SaTscan™ Software for the spatial and space–time scan statistics. <http://www.satscan.org/>.

- [O’Leary et al., 2013] O’Leary, C. M., Elliott, E. J., Nassar, N., and Bower, C. (2013). Exploring the potential to use data linkage for investigating the relationship between birth defects and prenatal alcohol exposure. *Birth Defects Res. Part A Clin. Mol. Teratol.*, 97(7):497–504.
- [Rankin et al., 2010] Rankin, J., Cans, C., Garne, E., Colver, A., Dolk, H., Uldall, P., Amar, E., and Krageloh-Mann, I. (2010). Congenital anomalies in children with cerebral palsy: a population-based record linkage study. *Dev Med Child Neurol*, 52(4):345–351.
- [Rooney et al., 2015] Rooney, J., Vajda, A., Heverin, M., Elamin, M., Crampsie, A., McLaughlin, R., Staines, A., and Hardiman, O. (2015). Spatial cluster analysis of population amyotrophic lateral sclerosis risk in Ireland. *Neurology*, 84(15):1537–1544.
- [Ventura et al., 2016] Ventura, C., Maia, M., Dias, N., Ventura, L. O., and Belfort, R. J. (2016). Zika: neurological and ocular findings in infant without microcephaly. *The Lancet*.

Capítulo 9

Copyright Notice

Copyright (c) 2016, Antony Peter Stevens All rights reserved.

Redistribution and use in source and binary forms of the programs listed in this document, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.