Diretrizes analíticas:
Como criar identificadores de deficiência usando a sintaxe SAS do Conjunto Resumido de Perguntas sobre Funcionalidade do Grupo de Washington (WG-SS)

Introdução

A deficiência pode ser melhor compreendida como um espectro contínuo (*continuum)*. Em termos de dificuldades de funcionalidade, uma "dificuldade" pode ser operacionalizada por meio uma gama de diversos descritores dentro de uma faixa que varia de nenhuma dificuldade a alguma dificuldade, muita dificuldade e incapacidade total de executar a ação em questão (definida como "não consegue de modo algum"). Cada um desses descritores representa um ponto de corte ou limiar para a determinação de um identificador final de deficiência para, por exemplo, caracterizar pessoas com e sem deficiência. Esses níveis de funcionalidade também estão representados nas categorias de respostas ao Conjunto Resumido de Perguntas sobre Funcionalidade do Grupo de Washington (WG-SS).

A prevalência de deficiências não pode ser definida com base em uma única estatística, mas pode ser calculada em diversos limiares, dependendo dos objetivos da coleta de dados e dos relatórios resultantes. Por exemplo, se o objetivo for garantir acesso equitativo a espaços públicos, o nível de inclusão para um identificador de deficiência pode ser o de *alguma dificuldade*, uma vez que pessoas com níveis mais baixos de dificuldade em termos de funcionalidade provavelmente seriam beneficiadas por adaptações concebidas para eliminar barreiras e facilitar esse acesso. A instalação de escadas rolantes em substituição a escadas comuns, por exemplo, é um elemento universal comum de projeto que beneficia pessoas com uma ampla gama de dificuldades em termos de mobilidade. Por outro lado, se o objetivo for garantir subsídios ou auxílios, o nível de inclusão para um identificador de deficiência pode ser o de *não consegue de modo algum*, uma vez que somente pessoas com limitações funcionais mais severas satisfariam critérios mais rigorosos de elegibilidade.

Os **Documentos de Implementação do Grupo de Washington** abrangem as ferramentas desenvolvidas pelo Grupo de Washington para Estatísticas sobre Pessoas com Deficiência (WG) para coletar dados sobre deficiências comparáveis internacionalmente em censos e pesquisas. Os documentos abordam as melhores práticas a serem adotadas na implementação do Conjunto Resumido, Conjunto Ampliado e Conjunto Resumido – Estendido, dos Módulos sobre Funcionalidade da Criança do WG/UNICEF para crianças nas faixas etárias de 2 a 4 e de 5 a 17 anos e do Módulo de Deficiência do WG/OIT LFS, bem como de outras ferramentas do WG. Os tópicos incluem tradução, especificações de perguntas, diretrizes analíticas, código de programação para análises, o uso das ferramentas para fins de desagregação e muitos outros.

Para localizar outros documentos de implementação do WG e para obter mais informações, visite o site do Grupo de Washington: http://www.washingtongroup-disability.com/.

A sintaxe SAS descrita neste documento permite o cálculo de quatro identificadores de deficiência em quatro limiares. A população de pessoas *com deficiência* calculada com base nesses quatro diferentes limiares produz os quatro identificadores de deficiência apresentados a seguir:

- **DEFICIÊNCIA1**: o nível de inclusão se refere a pelo menos um domínio/pergunta codificada como ALGUMA DIFICULDADE ou MUITA DIFICULDADE ou NÃO CONSEGUE DE MODO ALGUM.

- **DEFICIÊNCIA2**: o nível de inclusão se refere a pelo menos dois domínios/perguntas codificadas como ALGUMA DIFICULDADE ou qualquer um domínio/pergunta codificada como MUITA DIFICULDADE ou NÃO CONSEGUE DE MODO ALGUM.

- **DEFICIÊNCIA3**: o nível de inclusão se refere a qualquer domínio/pergunta codificada como MUITA DIFICULDADE ou NÃO CONSEGUE DE MODO ALGUM.

OBS.: **DEFICIÊNCIA3** É O PONTO DE CORTE RECOMENDADO PELO WG.

- **DEFICIÊNCIA4**: o nível de inclusão se refere a qualquer domínio codificado como NÃO CONSEGUE DE MODO ALGUM(4).

OBS.: A sintaxe SAS SPSS baseia-se nos *rótulos de variáveis e rótulos de valores* indicados nas tabelas abaixo. Não deixe de usar os mesmos *rótulos de variáveis e rótulos de valor* OU revise a sintaxe SAS para que ela reflita os *rótulos* usados no seu banco de dados.

O WG-SS é administrado como parte da Pesquisa Nacional de Saúde dos Estados Unidos (NHIS). Os dados usados na elaboração destas diretrizes foram extraídos da NHIS realizada em 2013.

Observação para usuários da NHIS: os nomes das variáveis contidos no arquivo de dados e na documentação da NHIS podem ser diferentes dos usados neste documento; por exemplo, a variável do domínio de cuidados pessoais referenciada como SC-SS neste documento tem o nome de UB_SS no arquivo de dados e na documentação da NHIS.

O código SAS usado para produzir os resultados deste documento foi incluído na íntegra no Apêndice.

Perguntas/Domínios do Conjunto Resumido do WG	Rótulo de variável
1. Você tem dificuldade para enxergar mesmo quando usa óculos?	VIS_SS
2. Você tem dificuldade para ouvir mesmo quando usa um aparelho auditivo?	HEAR_SS
3. Você tem dificuldade para andar ou subir degraus?	MOB_SS
4. Você tem dificuldade para se lembrar de coisas ou para se concentrar?	COG_SS
5. Você tem dificuldade (para realizar cuidados pessoais, como) para lavar o corpo ou se vestir?	SC_SS
6. Usando sua linguagem habitual, você tem dificuldade para se comunicar (por exemplo, para compreender ou ser compreendido(a) por outras pessoas)?	COM_SS

Os rótulos de valores usados para cada uma das perguntas do conjunto WG-SS são os seguintes:

1. Nenhuma dificuldade
2. Sim, alguma dificuldade
3. Sim, muita dificuldade
4. Não consigo de modo algum
7. Não quis responder
8. Não comprovado
9. Não sabe

SAS WG Short Set Syntax Annotated with Output Tables

Actual SAS syntax is indented and are in **Bold** text.

 NOTE: For data analysis, use your standard weighting and estimation techniques.

The syntax below produces frequency distributions on each the six domains. Codes 7 (REFUSED), 8 (NOT ASCERTAINED) and 9 (DON'T KNOW) are RECODED as **MISSING**.

Step 1: Generate frequency distributions on each of the six domain variables.

VIS_SS is the WG-SS Vision question.

> **If** VIS_SS in (**1**, **2**, **3**, **4**) **then** Vision=VIS_SS;
> **Else If** VIS_SS in (**7**, **8**, **9**) **then** Vision=**.**;
>
> **Proc Freq Data**=SS.Funcdisb13;
> **Tables** Vision;
> **Run**;

NOTE: *SS.Funcdisb13* is the name of the SAS file used for these analyses. When preparing your SAS code, replace this SAS file with the name of your SAS file.

Vision: Degree of difficulty seeing

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	No difficulty	13690	79.0	81.6	81.6
	Some difficulty	2708	15.6	16.2	97.8
	A lot of difficulty	**333**	**1.9**	**2.0**	**99.8**
	Cannot do at all	**36**	**.2**	**.2**	**100.0**
	Total	16767	96.8	100.0	
Missing		559	3.2		
Total		17326	100.0		

HEAR_SS is the WG-SS Hearing question.

> **If** HEAR_SS in (**1**, **2**, **3**, **4**) **then** Hearing =HEAR_SS;
> **Else If** HEAR_SS in (**7**, **8**, **9**   ) **then** Hearing =**.**;
>
> **Proc Freq Data**=SS.Funcdisb13;
> **Tables** Hearing;
> **Run**;

Hearing: Degree of difficulty hearing

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	No difficulty	13680	79.0	81.6	81.6
	Some difficulty	2753	15.9	16.4	98.0
	A lot of difficulty	**310**	**1.8**	**1.8**	**99.9**
	Cannot do at all	**23**	**.1**	**.1**	**100.0**
	Total	16766	96.8	100.0	
Missing		560	3.2		
Total		17326	100.0		

MOB_SS is the WG-SS Mobility question.

If MOB_SS in (**1**, **2**, **3**, **4**) **then** Mobility=MOB_SS;
Else If MOB_SS in (**7**, **8**, **9**   ) **then** Mobility=**.**;

Proc Freq Data=SS.Funcdisb13;
Tables Mobility;
Run;

Mobility: Degree of difficulty walking or climbing steps

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	No difficulty	13424	77.5	80.1	80.1
	Some difficulty	2165	12.5	12.9	93.0
	A lot of difficulty	**792**	**4.6**	**4.7**	**97.7**
	Cannot do at all	**380**	**2.2**	**2.3**	**100.0**
	Total	16761	96.7	100.0	
Missing		565	3.3		
Total		17326	100.0		

COM_SS is the WG-SS Communication question.

If COM_SS in (**1**, **2**, **3**,**4**) **then** Communication =COM_SS;
Else If COM_SS in (**7**, **8**, **9**) **then** Communication =**.**;

Proc Freq Data=SS.Funcdisb13;
Tables Communication;
Run;

Communication: Degree of difficulty communicating using usual language

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	No difficulty	15874	91.6	94.7	94.7
	Some difficulty	745	4.3	4.4	99.2
	A lot of difficulty	**94**	**.5**	**.6**	**99.7**
	Cannot do at all	**43**	**.2**	**.3**	**100.0**
	Total	16756	96.7	100.0	
Missing		570	3.3		
Total		17326	100.0		

SC_SS is the WG-SS Self-care question.

> **If** SC_SS in (**1**, **2**, **3**, **4**) **then** Self_Care=SC_SS;
> **Else If** SC_SS in (**7**, **8**, **9**) **then** Self_Care=**.**;
>
> **Proc Freq Data**=SS.Funcdisb13;
> **Tables** Self_Care;
> **Run**;

Self_Care Degree of difficulty with self-care

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	No difficulty	16029	92.5	95.7	95.7
	Some difficulty	544	3.1	3.2	98.9
	A lot of difficulty	**114**	**.7**	**.7**	**99.6**
	Cannot do at all	**68**	**.4**	**.4**	**100.0**
	Total	16755	96.7	100.0	
Missing		571	3.3		
Total		17326	100.0		

COG_SS is the WG-SS Cognition question.

> **If** COG_SS in (**1**, **2**, **3**, **4**) **then** Cognition=COG_SS;
> **Else If** COG_SS in (**7**, **8**, **9**   ) **then** Cognition=**.**;
>
> **Proc Freq Data**=SS.Funcdisb13;
> **Tables** Cognition;
> **Run**;

Cognition: Degree of difficulty remembering or concentrating

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	No difficulty	13719	79.2	81.9	81.9
	Some difficulty	2632	15.2	15.7	97.6
	A lot of difficulty	**382**	**2.2**	**2.3**	**99.9**
	Cannot do at all	**20**	**.1**	**.1**	**100.0**
	Total	16753	96.7	100.0	
Missing		573	3.3		
Total		17326	100.0		

Step 2: Calculate a variable, SUM_234

SUM_234 summates the number of domains coded SOME DIFFICULTY (2) or A LOT OF DIFFICULTY (3) or CANNOT DO AT ALL (4) for each person. This new variable is used in the determination of disability identifiers: DISABILITY1 and DISABILITY2.

The syntax below **counts** the number of domains/questions a person has that are coded SOME DIFFICULTY (2) or A LOT OF DIFFICULTY (3) or CANNOT DO AT ALL (4).

Possible range 0: no difficulties in any domain, to 6: all six domains coded SOME DIFFICULTY (2) or A LOT OF DIFFICULTY (3) or CANNOT DO AT ALL (4).
MISSING (9) are those who have coded 7, 8 or 9 on all six domains.

> **If** missing(Vision) and missing(Hearing) and missing(Mobility) and missing(Cognition) and missing(Self_Care) and missing(Communication) **then** SumPoints=**.**;
> **Else If** (Vision =**1**) and (Hearing =**1**) and (Mobility =**1**) and (Cognition =**1**) and (Self_Care =**1**) and (Communication =**1**) **then** SumPoints=**0**;
> **Else** SumPoints=SUM((Vision in(**2,3,4**)),(Hearing in(**2,3,4**)),(Mobility in(**2,3,4**)),(Cognition in(**2,3,4**)),(Self_Care in(**2,3,4**)),(Communication in(**2,3,4**)));
>
> **If** SumPoints =**.**   **then** SUM_234 =**.**;
> **Else If** SumPoints = **1 then** SUM_234=**1**;
> **Else If** SumPoints = **2 then** SUM_234=**2**;
> **Else If** SumPoints = **3 then** SUM_234=**3**;
> **Else If** SumPoints = **4 then** SUM_234=**4**;
> **Else If** SumPoints = **5 then** SUM_234=**5**;
> **Else If** SumPoints = **6 then** SUM_234=**6**;
> **Else If** SumPoints = **0 then** SUM_234=**0**;
>
> **Proc Freq Data**=SS.Funcdisb13;
> **Tables** SUM_234;
> **Run**;

SUM_234

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	.00	9266	53.5	55.2	55.2
	1.00	3839	22.2	22.9	78.1
	2.00	1892	10.9	11.3	89.4
	3.00	989	5.7	5.9	95.3
	4.00	481	2.8	2.9	98.2
	5.00	232	1.3	1.4	99.5
	6.00	78	.5	.5	100.0
	Total	16777	96.8	100.0	
Missing		549	3.2		
Total		17326	100.0		

Step 3: Calculate a variable, SUM_34

SUM_34 summates the number of domains coded A LOT OF DIFFICULTY (3) or CANNOT DO AT ALL (4) for each person. This new variable is used in the determination of disability identifier: DISABILITY2.

The syntax below counts the number of domains/questions a person has that are coded A LOT OF DIFFICULTY (3) or CANNOT DO AT ALL (4)
Possible range 0: no difficulties coded A LOT OF DIFFICULTY (3) or CANNOT DO AT ALL (4) in any domain, to 6: all six domains coded A LOT OF DIFFICULTY (3) or CANNOT DO AT ALL (4). MISSING (9) are those who have coded 7, 8 or 9 on all six domains.

If missing(Vision) and missing(Hearing) and missing(Mobility) and missing(Cognition) and missing(Self_Care) and missing(Communication) **then** SumPoints2=**.**;
Else If (Vision in(**1**,**2**)) and (Hearing in(**1**,**2**)) and (Mobility in(**1**,**2**)) and (Cognition in(**1**,**2**)) and (Self_Care in(**1**,**2**)) and (Communication in(**1**,**2**)) **then** SumPoints2=**0**;
Else SumPoints2=SUM((Vision in(**3**,**4**)),(Hearing in(**3**,**4**)),(Mobility in(**3**,**4**)),(Cognition in(**3**,**4**)),(Self_Care in(**3**,**4**)),(Communication in(**3**,**4**)));

If SumPoints2 =**.**   **then** SUM_34 =**.**;
Else If SumPoints2 = **1** **then** SUM_34=**1**;
Else If SumPoints2 = **2** **then** SUM_34=**2**;
Else If SumPoints2 = **3** **then** SUM_34=**3**;
Else If SumPoints2 = **4** **then** SUM_34=**4**;
Else If SumPoints2 = **5** **then** SUM_34=**5**;
Else If SumPoints2 = **6** **then** SUM_34=**6**;
Else If SumPoints2 = **0** **then** SUM_34=**0**;

Proc Freq Data=SS.Funcdisb13;
Tables SUM_34;
Run;

SUM_34

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	.00	14905	86.0	88.8	88.8
	1.00	1367	7.9	8.1	97.0
	2.00	345	2.0	2.1	99.0
	3.00	117	.7	.7	99.7
	4.00	31	.2	.2	99.9
	5.00	9	.1	.1	100.0
	6.00	3	.0	.0	100.0
	Total	16777	96.8	100.0	
Missing		549	3.2		
Total		17326	100.0		

Step 4: Calculate Disability Identifier: **DISABILITY1**

The syntax below calculates the first disability identifier: **DISABILITY1** where the level of inclusion is at least one domain/question is coded SOME DIFFICULTY or A LOT OF DIFFICULTY or CANNOT DO AT ALL.
MISSING (.) are those who have coded 7, 8 or 9 on all six domains.

> **If** missing(Vision) and missing(Hearing) and missing(Mobility) and missing(Cognition) and missing(Self_Care) and missing(Communication) **then** Disability1=**.**;
> **Else If** SUM_234 >=**1 then** Disability1=**1**;
> **Else** Disability1=**2**;

NOTE: SUM_234 >= 1 means that at least one of the six domains is coded at least SOME DIFFICULTY (2).

> **Proc Freq Data**=SS.Funcdisb13;
> **Tables** Disability1;
> **Run**;

DISABILITY1

		Frequency	Percent	Valid Percent	Cumulative Percent	Weighted Percent*
Valid	without disability	9266	53.5	55.2	55.2	58.1
	with disability	7511	43.4	44.8	100.0	41.9
	Total	16777	96.8	100.0		100.0
Missing		549	3.2			
Total		17326	100.0			

*Weighted estimate provided – but is not part of the SAS syntax.

Step 5: Calculate Disability Identifier: DISABILITY2

The syntax below calculates the second disability identifier: **DISABILITY2** where the level of inclusion is: at least 2 domains/questions are coded SOME DIFFICULTY or any 1 domain/question is coded A LOT OF DIFFICULTY or CANNOT DO AT ALL.
MISSING (9) are those who have coded 7, 8 or 9 on all six domains.

> **If** missing(Vision) and missing(Hearing) and missing(Mobility) and missing(Cognition) and missing(Self_Care) and missing(Communication) **then** Disability2=**.**;
> **Else If** (SUM_234 >=**2** OR SUM_34=**1**) **then** Disability2=**1**;
> **Else** Disability2=**2**;

NOTE: The above syntax identifies those with at least two of the six domains coded as at least SOME DIFFICULTY (2): SUM_234 >= 2, OR those who have one domain that is coded A LOT OF DIFFICULTY (3) or CANNOT DO AT ALL (4): SUM_34 = 1.

> **Proc Freq Data**=SS.Funcdisb13;
> **Tables** Disability2;
> **Run**;

DISABILITY2

		Frequency	Percent	Valid Percent	Cumulative Percent	Weighted Percent*
Valid	without disability	12707	73.3	75.7	75.7	78.3
	with disability	4070	23.5	24.3	100.0	21.7
	Total	16777	96.8	100.0		100.0
Missing		549	3.2			
Total		17326	100.0			

*Weighted estimate provided – but is not part of the SAS syntax.

Step 6: Calculate Disability Identifier: DISABILITY3

The syntax below calculates the third disability identifier: **DISABILITY3** where the level of inclusion is: any 1 domain/question is coded A LOT OF DIFFICULTY or CANNOT DO AT ALL.
MISSING (9) are those who have coded 7, 8 or 9 on all six domains.
THIS IS THE CUT-OFF RECOMMENDED BY THE WG.

> **If** missing(Vision) and missing(Hearing) and missing(Mobility) and missing(Cognition) and missing(Self_Care) and missing(Communication) **then** Disability3=**.**;
> **Else If** ((Vision = **3** OR Vision = **4**) OR (Hearing= **3** OR Hearing = **4**) OR (Mobility= **3** OR Mobility = **4**) OR (Communication= **3** OR Communication = **4**) OR (Self_Care = **3** OR Self_Care = **4**) OR (Cognition = **3** OR Cognition = **4**)) **then** Disability3=**1**;
> **Else** Disability3 = **2**;

> **Proc Freq Data**=SS.Funcdisb13;
> **Tables** Disability3;
> **Run**;

DISABILITY3

		Frequency	Percent	Valid Percent	Cumulative Percent	Weighted Percent*
Valid	without disability	14905	86.0	88.8	88.8	90.5
	with disability	1872	10.8	11.2	100.0	9.5
	Total	16777	96.8	100.0		100.0
Missing		549	3.2			
Total		17326	100.0			

*Weighted estimate provided – but is not part of the SAS syntax.

Step 7: Calculate Disability Identifier: DISABILITY4

The syntax below calculates the fourth disability identifier: **DISABILITY4** where the level of inclusion is any one domain is coded CANNOT DO AT ALL (4).
MISSING (.) are those who have coded 7, 8 or 9 on all six domains.

> **If** missing(Vision) and missing(Hearing) and missing(Mobility) and missing(Cognition) and missing(Self_Care) and missing(Communication) **then** Disability4=**.**;
> **Else If** ((Vision = **4**) OR (Hearing = **4**) OR (Mobility = **4**) OR (Communication = **4**) OR (Self_Care = **4**) OR (Cognition = **4**)) **then** Disability4=**1**;
> **Else** Disability4 = **2**;
>
> **Proc Freq Data**=SS.Funcdisb13;
> **Tables** Disability4;
> **Run**;

DISABILITY4

		Frequency	Percent	Valid Percent	Cumulative Percent	Weighted Percent*
Valid	without disability	16312	94.1	97.2	97.2	97.8
	with disability	465	2.7	2.8	100.0	2.2
	Total	16777	96.8	100.0		100.0
Missing		549	3.2			
Total		17326	100.0			

*Weighted estimate provided – but is not part of the SAS syntax.

APPENDIX

SAS Code Used with the 2013 NHIS Data File

```
Data SS.Funcdisb13;
     Set NHIS.Funcdisb13 ;

*The syntax below produces frequency distributions on each the six domains. Codes 7
(REFUSED), 8 (NOT ASCERTAINED) and 9 (DON'T KNOW) are INCLUDED as MISSING.;

*Vision;
*Generate frequency distribution for each domain question. Convert 7,8,9 to
missing;
     If VIS_SS2 in (1,2,3,4) then Vision=VIS_SS2;
     Else Vision=.;

*Communication;
*Generate frequency distribution for each domain question. Convert 7,8,9 to
missing;
     If COM_SS in (1,2,3,4) then Communication=COM_SS;
     Else Communication=.;

*Hearing;
*Generate frequency distribution for each domain question. Recode 7,8,9 to .;
     If HEAR_SS2 in (1,2,3,4) then Hearing=HEAR_SS2;
     Else If HEAR_SS2 in (7,8,9 ) then Hearing=.;

*Cognition: Degree of difficulty remembering or concentrating;
*Generate frequency distribution for each domain question. Recode 7,8,9 to .;
     If COG_SS in (1,2,3,4) then Cognition=COG_SS;
     Else If COG_SS in (7,8,9) then Cognition=.;

*Self care;
     If UB_SS in (1,2,3,4) then Self_Care=UB_SS;
     Else Self_Care=.;


*Mobility;
     If MOB_SS2 in (1,2,3,4) then Mobility=MOB_SS2;
     Else Mobility=.;

*Step 1: Generate frequency distributions on each of the six domain variables.;

*Step 2: Calculate a variable, SUM_234. SUM_234 summates the number of domains
coded SOME DIFFICULTY (2) or A LOT OF DIFFICULTY (3) or CANNOT DO AT ALL (4)for
each person. This new variable is used in the determination of disability
identifiers: DISABILITY1 and DISABILITY2.
The syntax below counts the number of domains/questions a person has that are coded
SOME DIFFICULTY (2) or A LOT OF DIFFICULTY (3) or CANNOT DO AT ALL (4).
Possible range 0: no difficulties in any domain, to 6: all six domains coded SOME
DIFFICULTY (2) or A LOT OF DIFFICULTY (3) or CANNOT DO AT ALL (4).
MISSING (9) are those who have coded 7, 8 or 9 on all six domains.;

     If missing(Vision) and missing(Hearing) and missing(Mobility) and
     missing(Cognition) and  missing(Self_Care) and missing(Communication) then
     SumPoints=.;
     Else If (Vision =1) and (Hearing =1) and (Mobility =1) and (Cognition =1) and
     (Self_Care =1) and (Communication =1) then SumPoints=0;
```

```
      Else SumPoints=SUM( (Vision in(2,3,4)),(Hearing in(2,3,4)),(Mobility
      in(2,3,4)),(Cognition in(2,3,4)),(Self_Care in(2,3,4)),(Communication
      in(2,3,4)) );

            If SumPoints =.     then  SUM_234 =.;
      Else If SumPoints = 1    then  SUM_234=1;
      Else If SumPoints = 2    then  SUM_234=2;
      Else If SumPoints = 3    then  SUM_234=3;
      Else If SumPoints = 4    then  SUM_234=4;
      Else If SumPoints = 5    then  SUM_234=5;
      Else If SumPoints = 6    then  SUM_234=6;
      Else If SumPoints = 0    then  SUM_234=0;
```

Step 3: Calculate a variable, SUM_34. SUM_34 summates the number of domains coded
A LOT OF DIFFICULTY (3) or CANNOT DO AT ALL (4) for each person.
This new variable is used in the determination of disability identifier:
DISABILITY2.
The syntax below counts the number of domains/questions a person has that are coded
A LOT OF DIFFICULTY (3) or CANNOT DO AT ALL (4)
Possible range 0: no difficulties coded A LOT OF DIFFICULTY (3) or CANNOT DO AT ALL
(4) in any domain, to 6: all six domains coded A LOT OF DIFFICULTY (3)or CANNOT DO
AT ALL (4). MISSING (9) are those who have coded 7, 8 or 9 on all six domains.;

```
      If missing(Vision) and missing(Hearing) and missing(Mobility) and
      missing(Cognition) and  missing(Self_Care) and missing(Communication) then
      SumPoints2=.;
      Else If (Vision in(1,2)) and (Hearing in(1,2)) and (Mobility in(1,2)) and
      (Cognition in(1,2)) and (Self_Care in(1,2)) and (Communication in(1,2))
      then SumPoints2=0;
      Else SumPoints2=SUM( (Vision in(3,4)),(Hearing in(3,4)),(Mobility
      in(3,4)),(Cognition in(3,4)),(Self_Care in(3,4)),(Communication in(3,4)) );

            If SumPoints2 =.    then  SUM_34 =.;
      Else If SumPoints2 = 1   then  SUM_34=1;
      Else If SumPoints2 = 2   then  SUM_34=2;
      Else If SumPoints2 = 3   then  SUM_34=3;
      Else If SumPoints2 = 4   then  SUM_34=4;
      Else If SumPoints2 = 5   then  SUM_34=5;
      Else If SumPoints2 = 6   then  SUM_34=6;
      Else If SumPoints2 = 0   then  SUM_34=0;
```

Step 4: Calculate Disability Identifier: DISABILITY1. The syntax below calculates
the first disability identifier: DISABILITY1 where the level of inclusion is at
least one domain/question is coded SOME DIFFICULTY or A LOT OF DIFFICULTY or CANNOT
DO AT ALL. MISSING (9) are those who have coded 7, 8 or 9 on all six domains;

```
      If missing(Vision) and missing(Hearing) and missing(Mobility) and
      missing(Cognition) and  missing(Self_Care) and missing(Communication) then
      Disability1=.;
      Else If SUM_234 >=1 then Disability1=1;
      Else Disability1=2;
```

Step 5: Calculate Disability Identifier: DISABILITY2. The syntax below calculates
the second disability identifier: DISABILITY2 where the level of inclusion is: at
least 2 domains/questions are coded SOME DIFFICULTY or any 1 domain/question is
coded A LOT OF DIFFICULTY or CANNOT DO AT ALL.
MISSING (9) are those who have coded 7, 8 or 9 on all six domains;

```sas
        If missing(Vision) and missing(Hearing) and missing(Mobility) and
        missing(Cognition) and  missing(Self_Care) and missing(Communication) then
        Disability2=.;
        Else If (SUM_234 >=2 OR SUM_34=1 ) then Disability2=1;
        Else Disability2=2;

*Step 6: Calculate Disability Identifier: DISABILITY3.The syntax below calculates
the third disability identifier: DISABILITY3 where the level of inclusion is: any 1
domain/question is coded A LOT OF DIFFICULTY or CANNOT DO AT ALL. MISSING (9) are
those who have coded 7, 8 or 9 on all six domains.
THIS IS THE CUT-OFF RECOMMENDED BY THE WG.;

        If missing(Vision) and missing(Hearing) and missing(Mobility) and
        missing(Cognition) and  missing(Self_Care) and missing(Communication) then
        Disability3=.;
        Else IF ((vision = 3 OR vision = 4) OR (Hearing= 3 OR Hearing = 4) OR
        (mobility= 3 OR mobility = 4) OR (Communication= 3 OR Communication = 4) OR
        (Self_Care = 3 OR Self_Care = 4) OR (Cognition = 3 OR Cognition = 4)) then
        Disability3=1;
        Else Disability3 = 2;

*Step 7: Calculate Disability Identifier: DISABILITY4.The syntax below calculates
the fourth disability identifier: DISABILITY4 where the level of inclusion is any
one domain is coded CANNOT DO AT ALL (4). MISSING (9) are those who have coded 7, 8
or 9 on all six domains;

        If missing(Vision) and missing(Hearing) and missing(Mobility) and
        missing(Cognition) and  missing(Self_Care) and missing(Communication) then
        Disability4=.;
        Else IF ((vision = 4) OR (Hearing = 4) OR (mobility = 4) OR (Communication =
        4) OR (Self_Care = 4) OR (Cognition = 4)) then Disability4=1;
        Else Disability4 = 2;

        Label
                Vision="Degree of difficulty seeing"
                Communication="Degree of difficulty communicating using usual language"
                Hearing="Degree of difficulty hearing"
                Cognition="Degree of difficulty remembering or concentrating"
                Self_Care="Degree of difficulty with self-care"
                Mobility="Degree of difficulty walking or climbing steps";

         Format Vision Communication Hearing HEAR_3_R HEAR_4_R Cognition Self_Care
         UB_1_R UB_2_R Mobility MOB_4_R MOB_5_R Diff.
         Disability1 Disability2 Disability3 Disability4 DisabF.;
         Run;

Title "NHIS 2013: Unweighted frequencies with missing included in the percent";
        Proc freq data=SS.Funcdisb13;
         Tables  Vision Hearing Mobility Communication Self_Care Cognition SUM_234
         SUM_34 Disability1 Disability2 Disability3 Disability4/missing;
        Run;
Title;
Title "NHIS 2013: Unweighted frequencies";
        Proc freq data=SS.Funcdisb13;
         Tables  Vision Hearing Mobility Communication Self_Care Cognition SUM_234
         SUM_34 Disability1 Disability2 Disability3 Disability4;
        Run;
Title;
```

```
Proc format library=SS.SS;
     Value Diff
          1="No Difficulty"
          2="Some Difficulty"
          3="A lot of Difficulty"
          4="Cannot do at all"
          .="Missing"
     ;
     Value DisabF
          1="With Disability"
          2="Without Disability"
     ;
Run;
```