A Fast and Easy Way to Calculate Age-Standardized Death Rates in SAS®

Ash Roy

1st April 2011

About Me

- MD
- MPH
- Health Informatics
- Worked for GSK – pharmaceutical
- Working for CIHI for more than one and half years
- Featured in a journal for my passion for HI
- Contact: ashoke.k.roy@gmail.com

Project Background

- OTR DDS
- Ten-year trend of injury deaths
- Injury deaths by cause, sex, age-group
- Injury deaths by county, region and province
- Drowning by pace, age-group
- Alcohol involvement
- Total deaths and age-standardized rates

The Task

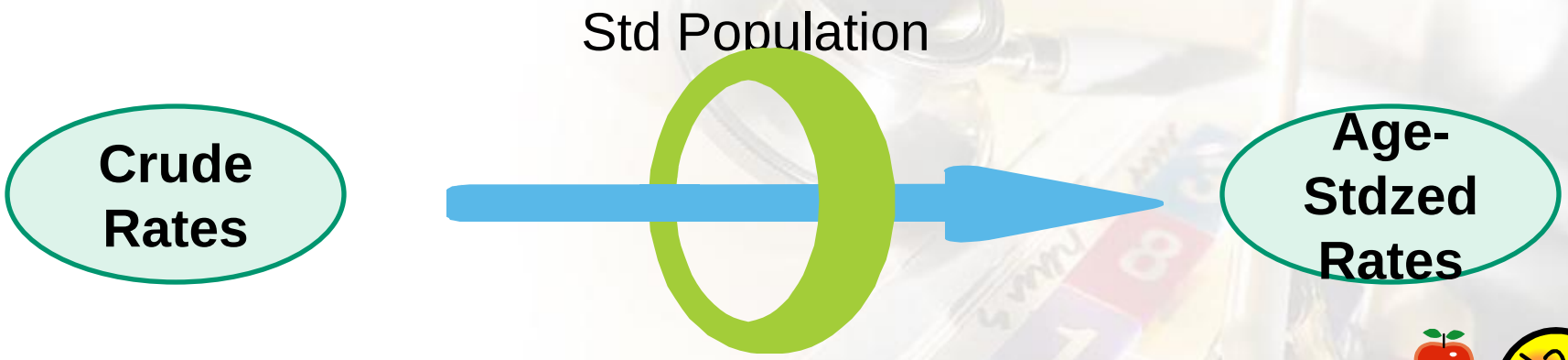
- Counting all number including age-standardized rates by counties (49), health planning regions (7) and province for five years
- Draw an Excel table directly from SAS®

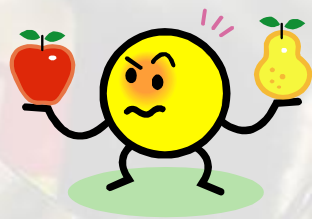
Final Table Look

		2002		2003		2004		2005		2006	
		No.	Rate	No.	Rate	No.	Rate	No.	Rate	No.	Rate
Region: (MOHLTC's 7 Health Planning Region)	County 1										
	….										
	County 49										

Age-Standardization

- Method of applying observed age-specific rates to a standard age distribution
- When comparing more than one group of populations at a single point in time, or
- One group of population at different points in time
- It helps to take away the effect of age as a confounder





Standardization

- Direct and indirect method
- The formula for direct standardized rate for 20 age groups

Age-Standardized Death Rate =

$$\sum_{i=1}^{20} WiPi$$

Where p_i is the age-specific death rate for the age group *i* and w_i (or the weight) is the proportion of age group *i* in the standard population

Conventional Way

- The number of deaths (observed) are calculated
- Age-Standardized Death Rate =

  (Total Observed Deaths / Total Standard Population) * 10,000

Crude death rates

Std pop

Exptd deaths

Stdzed rate

Conventional Way Example

Age Group	Deaths (Col1)	Population (Col2)	Age-specific Death Rate per 10,000 (Col3)	Standard Population Canada 1991 (Col4)	Percentage of Standard Population (Col5)	Expected Deaths (Col6)
0 to 4 years	798	694522	11.5	1958442	6.985101	2252
5 to 9 years	96	793647	1.2	1934715	6.900474	232
10 to 14 years	103	823147	1.3	1902403	6.785228	247
15 to 19 years	309	819901	3.8	1927520	6.874812	732
20 to 24 years	372	800632	4.6	2085828	7.439443	959
25 to 29 years	366	810652	4.5	2508203	8.945912	1129
30 to 34 years	543	898935	6.0	2585279	9.220816	1551
35 to 39 years	862	1031335	8.4	2347683	8.373392	1972
40 to 44 years	1411	1033105	13.7	2131364	7.601855	2920
45 to 49 years	1948	913536	21.3	1673878	5.970157	3565
50 to 54 years	2700	797515	33.9	1342968	4.789913	4553
55 to 59 years	3640	649649	56.0	1235392	4.406226	6918
60 to 64 years	4663	503234	92.7	1191257	4.248811	11043
65 to 69 years	6437	436840	147.4	1086077	3.87367	16009
70 to 74 years	9253	396894	233.1	834625	2.976825	19455
75 to 79 years	12564	322108	390.1	622563	2.220472	24286
80 to 84 years	13221	207714	636.5	382053	1.362654	24318
85 to 89 years	11983	106411	1126.1	192155	0.685352	21639
90 years and over	10963	51252	2139.0	95015	0.338886	20324
	82232	12091029	68.0	**28037420**	100	**164105**

Age-Standardized Death Rate = (Total Expected Deaths / Total Standard Population) * 10,000

= (164,105 / 28,037,420) * 10,000

= **58.5**

Calculations: How Many Times??

49 Counties * 5 years	=	245 times
7 HP Regions * 5 years	=	35 times
Province * 5 years	=	5 times
Total		**285 times**

Quick Way

Age-specific death rates * Std pop weight for the age grp = Cross Products

$\sum$

Age-Standardized Death Rate

Quick Way Example

Age Group	Deaths (Col1)	Population (Col2)	Age-specific Death Rate per 10,000 (Col3)	Standard Population Canada 1991 (Col4)	Population Weight (Col5)	Cross Product (Col6)
0 to 4 years	798	694522	11.5	1958442	0.069851	0.802582
5 to 9 years	96	793647	1.2	1934715	0.069005	0.083469
10 to 14 years	103	823147	1.3	1902403	0.067852	0.084903
15 to 19 years	309	819901	3.8	1927520	0.068748	0.259094
20 to 24 years	372	800632	4.6	2085828	0.074394	0.345661
25 to 29 years	366	810652	4.5	2508203	0.089459	0.403898
30 to 34 years	543	898935	6.0	2585279	0.092208	0.556982
35 to 39 years	862	1031335	8.4	2347683	0.083734	0.699856
40 to 44 years	1411	1033105	13.7	2131364	0.076019	1.03825
45 to 49 years	1948	913536	21.3	1673878	0.059702	1.27306
50 to 54 years	2700	797515	33.9	1342968	0.047899	1.621633
55 to 59 years	3640	649649	56.0	1235392	0.044062	2.46882
60 to 64 years	4663	503234	92.7	1191257	0.042488	3.936977
65 to 69 years	6437	436840	147.4	1086077	0.038737	5.707996
70 to 74 years	9253	396894	233.1	834625	0.029768	6.94003
75 to 79 years	12564	322108	390.1	622563	0.022205	8.661073
80 to 84 years	13221	207714	636.5	382053	0.013627	8.673295
85 to 89 years	11983	106411	1126.1	192155	0.006854	7.717785
90 years and over	10963	51252	2139.0	95015	0.003389	7.24891
	82232	12091029	68.0	28037420	1	**58.52427**

Key Tasks

Create a data subset from the complex databases that contains

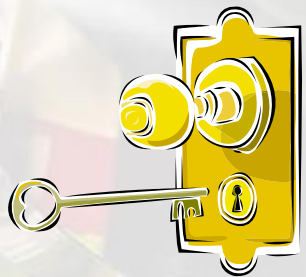
Every case of death by id number

Year of occurrence for five years

County code for 49 counties

7 Health Planning Regions in Ontario

- Count population by county, region and province by age-group
- Standard population by age-group
- Draw the desired table



Steps

Step 1
- No. of deaths by year, county and age-group

Step 2
- Population count by county, year and age-group

Step 3
- Calculating population weight of standard population by age-group

Step 4
- Assigning population weight to county population

Step 5
- Spreading population weight to every death case

Step 1 – Code

Actual codes

Number of Deaths

```
proc sql;
      create table cc_cnty as
            select coroner_case_id,
fiscal_year, age_group, county_name,
          from coroner_case
      (where=(2002<=fiscal_year<=2006)) ;
quit;
```

Step 1 - Actual Output

Log - (Untitled)

```
74                    end as prov
75
76                  from otrdds.coroner_case (where=(2002<=fiscal_year<=2006 and primary_ecode ne '0'))
76   ! as a
77                       left join otrdds.death_cause (where=(ENVIRONMENT_PRIMARY_INDICATOR = 'Y')) as b
78                           on a.coroner_case_id=b.coroner_case_id,
79                           otrdds.municipality as c
80                       where b.municipality_num=c.municipality_num
81                       ;
NOTE: A CASE expression has no ELSE clause. Cases not accounted for by the WHEN clauses will result in
      a missing value for the CASE expression.
NOTE: Table WORK.C_C_CNTY_RGN created, with 20245 rows and 8 columns.

82    quit;
NOTE: PROCEDURE SQL used (Total process time):
      real time           11.38 seconds
      cpu time            1.81 seconds
```

VIEWTABLE: Work.C_c_cnty_rgn

	FISCAL_YEAR	MUNICIPALITY_NUM	county_code	age_group	county	region	prov
1	2002	772	20	17	Toronto	Toronto	ON
2	2002	149	24	12	Halton R.M	Central West	ON
3	2002	772	20	18	Toronto	Toronto	ON
4	2002	1080	20	17	Toronto	Toronto	ON
5	2002	352	20	16	Toronto	Toronto	ON
6	2002	1006	26	19	Niagara	Central South	ON
7	2002	1080	20	20	Toronto	Toronto	ON
8	2002	614	39	15	Middlesex	Southwest	ON
9	2002	772	20	18	Toronto	Toronto	ON
10	2002	3700	37	16	Essex	Southwest	ON
11	2002	1080	20	17	Toronto	Toronto	ON
12	2002	1080	20	14	Toronto	Toronto	ON
13	2002	1080	20	18	Toronto	Toronto	ON
14	2002	1080	20	16	Toronto	Toronto	ON
15	2002	564	10	20	Frontenac	East	ON
				16	Grey	Southwest	ON

Step 2 - Code

Population Count

```
proc sql ;
 create table cnty_pop as
      select estimate_year, age_group,
sum(county_population) as cnty_pop, county_name
            from population (where = (2002 <=
estimate_year<=2006))
                  group by estimate_year, age_group,
county_name;
quit;
```

Step 2 - Output

```
89     /* Ontario county population by year and age group*/
90     proc sql ;
91      create table cnty_pop_rgn as
92         select estimate_year, age_group, sum(county_population) as cnty_pop,
93              input(county_code, 2.0) as county_code
94
95                   from otrdds.population (where = (2002 <= estimate_year<=2006))
96                   group by estimate_year, age_group, county_code;
NOTE: Table WORK.CNTY_POP_RGN created, with 4900 rows and 4 columns.

97
98     quit;
NOTE: PROCEDURE SQL used (Total process time):
      real time               0.25 seconds
      cpu time                0.15 seconds
```

table4

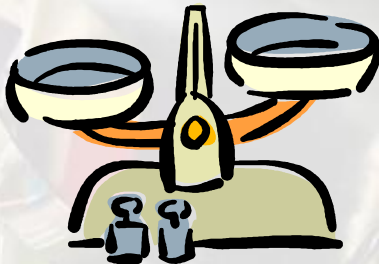
VIEWTABLE: Work.Cnty_pop_rgn

	ESTIMATE_YEAR	AGE_GROUP	cnty_pop	county_code
1	2002	1	1085	1
2	2002	1	1264	10
3	2002	1	373	11
4	2002	1	1222	12
5	2002	1	228	13
6	2002	1	663	14
7	2002	1	1128	15
8	2002	1	577	16
9	2002	1	5888	18
10	2002	1	7886	19
11	2002	1	876	2
12	2002	1	29924	20
13	2002	1	12944	21
14	2002	1	638	22
15	2002	1	2158	23
16	2002	1	3968	24

Step 3 - Code

Population Weight

```
proc sql;
      create table pop_w91 as
   select age_group, standard_pop91,
standard_pop91/sum(standard_pop91) as Weight91
                  from standard_pop91;
quit;
```



Step 3 - Output

Log - (Untitled)

```
      cpu time               0.15 seconds


99    /*Population Weight 1991 ******************/
100   proc sql;
101       create table pop_w91 as /* Standard population and its weight by age group for the year
101 ! 1991*/
102           select age_group, standard_pop91, standard_pop91/sum(standard_pop91) as Weight91
103               from otrdds.standard_pop91;
NOTE: The query requires remerging summary statistics back with the original data.
NOTE: Table WORK.POP_W91 created, with 20 rows and 3 columns.

104   quit;
NOTE: PROCEDURE SQL used (Total process time):
      real time              0.00 seconds
      cpu time               0.00 seconds
```

table4

VIEWTABLE: Work.Pop_w91

	AGE_GROUP	STANDARD_POP91	Weight91
1	1	403061	0.0143335728
2	2	1550285	0.0551309181
3	3	1953045	0.0694537868
4	4	1913115	0.0680338043
5	5	1926090	0.0684952186
6	6	2109452	0.0750159006
7	7	2529239	0.08994428
8	8	2598289	0.092399822
9	9	2344872	0.0833878585
10	10	2138891	0.0760628043
11	11	1674153	0.0595358866
12	12	1339902	0.0476493209
13	13	1238441	0.0440411855
14	14	1190217	0.0423262535
15	15	1084588	0.0385698966
16	16	834024	0.0296593909

Step 4 - Code

Assign Population Weight to County

```
proc sql;
  create table cnty_pop_w91 as
      select a.estimate_year, a.age_group,
a.county_code, a.cnty_pop, b.Weight91
            from cnty_pop as a
                left join pop_w91 as b
                        on a.age_group=b.age_group
                group by estimate_year,
a.age_group, county_code;
quit;
```

Step 4 - Output

Log - (Untitled)

```
107   proc sql;
108     create table cnty_pop_w91 as /*County population by age group and its weight*/
109       select a.estimate_year, a.age_group, a.county_code, a.cnty_pop, b.Weight91
110           from cnty_pop_rgn as a
111               left join pop_w91 as b
112                   on a.age_group=b.age_group
113                       group by estimate_year, a.age_group, county_code;
WARNING: A GROUP BY clause has been transformed into an ORDER BY clause because neither the SELECT
         clause nor the optional HAVING clause of the associated table-expression referenced a summary
         function.
NOTE: Table WORK.CNTY_POP_W91 created, with 4900 rows and 5 columns.

114   quit;
NOTE: PROCEDURE SQL used (Total process time):
      real time               0.04 seconds
      cpu time                0.04 seconds
```

table4

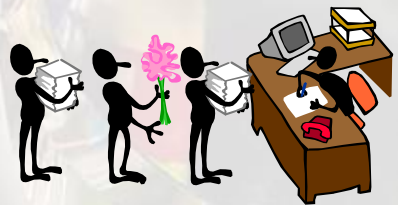
VIEWTABLE: Work.Cnty_pop_w91

	ESTIMATE_YEAR	AGE_GROUP	county_code	cnty_pop	Weight91
1	2002	1	1	1085	0.0143335728
2	2002	1	2	876	0.0143335728
3	2002	1	6	8586	0.0143335728
4	2002	1	7	965	0.0143335728
5	2002	1	9	632	0.0143335728
6	2002	1	10	1264	0.0143335728
7	2002	1	11	373	0.0143335728
8	2002	1	12	1222	0.0143335728
9	2002	1	13	228	0.0143335728
10	2002	1	14	663	0.0143335728
11	2002	1	15	1128	0.0143335728
12	2002	1	16	577	0.0143335728
13	2002	1	18	5888	0.0143335728
14	2002	1	19	7886	0.0143335728
15	2002	1	20	29924	0.0143335728
16	2002	1	21	12944	0.0143335728

Step 5 - Code

Assign Population Weight to Every Case

```
proc sql;
      create table cnty_std91 as
            select a.coroner_case_id, a.fiscal_year,
a.region, a.county, 1e4/cnty_pop*Weight91 as
thou10_w
            from cc_cnty as a
                  left join cnty_pop_w91 as b
                        on
a.fiscal_year=b.estimate_year and
                  a.age_group=b.age_group and
                  a.county_code=b.county_code;
quit;
```



Step 5 - Output

```
115   proc sql;
116       create table cnty_pw91 as /* weight per 10 thousand*//*<================*/
117           select a.coroner_case_id, a.fiscal_year, a.region, a.county, 1e4/cnty_pop*Weight91 as
117 ! thou10_w
118                from c_c_cnty_rgn as a
119                    left join cnty_pop_w91 as b
120                        on a.fiscal_year=b.estimate_year and
121                           a.age_group=b.age_group and
122                           a.county_code=b.county_code;
NOTE: Table WORK.CNTY_PW91 created, with 20245 rows and 5 columns.

123   quit;
NOTE: PROCEDURE SQL used (Total process time):
      real time           0.15 seconds
      cpu time            0.21 seconds
```

table4

VIEWTABLE: Work.Cnty_pw91

	FISCAL_YEAR	region	county	thou10_w
1	2002	East	Frontenac	0.1133985193
2	2002	East	Frontenac	0.1133985193
3	2002	Central East	Peterborough	0.1270706812
4	2002	Central East	Victoria	0.2484154739
5	2002	Central East	Durham R.M.	0.0243437039
6	2002	Central East	York R.M.	0.0181759737
7	2002	Toronto	Toronto	0.0047899923
8	2002	Toronto	Toronto	0.0047899923
9	2002	Toronto	Toronto	0.0047899923
10	2002	Toronto	Toronto	0.0047899923
11	2002	Toronto	Toronto	0.0047899923
12	2002	Toronto	Toronto	0.0047899923
13	2002	Toronto	Toronto	0.0047899923
14	2002	Toronto	Toronto	0.0047899923
15	2002	Toronto	Toronto	0.0047899923
16	2002	Toronto	Toronto	0.0047899923

Final Step - Code

Draw Table

```
ods chtml file = 'H:\SAS\standardization\table4.xls';
proc tabulate data=cnty_pw91 ;
      class region county fiscal_year ;
      var thou10_w;
      table region=''*county='',
fiscal_year=''*(n='No.'*f=5.0
thou10_w='Rate'*sum=''*f=4.1)/row=float;
      title 'Total Number and Age-Standardized Death
Rates per 10,000 Population by county of Injury -
Ontario, 2002-2003 through 2006-2007';
run;
ods chtml close;
```

Final Step - Output

Log - (Untitled)

```
124   proc tabulate data=cnty_pw91 ;
125       class region county fiscal_year ;
126       var thou10_w;
127       table region=''*county='', fiscal_year=''*(n='No.'*f=5.0
127 ! thou10_w='Rate'*sum=''*f=4.1)/row=float;
128       title 'Injury Death Rates per 10,000 Population by county of Injury - Ontario, 2002-2003
128 ! through 2006-2007';
129
130   run;

NOTE: There were 20245 observations read from the data set WORK.CNTY_PW91.
NOTE: PROCEDURE TABULATE used (Total process time):
      real time             0.06 seconds
      cpu time              0.12 seconds
```

Output - (Untitled)

Injury Death Rates per 10,000 Population by county of Injury - Ontario, 2002-2003 through 2006-2007

09:21 Monday, March 21, 201

		2002		2003		2004		2005		2006	
		No.	Rate	No.	Rate	No.	Rate	No.	Rate	No.	Rate
North	Timiskaming District	20	5.6	20	5.9	26	7.7	18	5.0	19	4.3
Southwest	Bruce	31	4.8	18	2.5	22	3.3	32	3.9	37	5.1
	Elgin	29	3.3	24	3.0	25	2.8	39	4.3	39	4.4
	Essex	154	3.6	156	3.7	139	3.2	127	2.9	137	3.1
	Grey	50	5.2	42	3.9	33	3.0	47	4.5	57	5.5
	Huron	28	3.9	38	5.5	35	5.8	20	2.9	23	3.3
	[illegible]	34	3.2	36	3.0	37	3.4	40	3.6	47	4.3
	Lambton	50	3.7	49	3.5	43	3.2	35	2.4	48	3.4
	Middlesex	157	3.6	153	3.3	161	3.5	186	3.9	152	3.2

SAS ODS to Excel - Codes

```
ods listing close;
ods chtml file = 'H:\OTR_DDS\QuickStat\table4.xls';
proc tabulate data=cnty_pw91 ;
        class region county fiscal_year ;
        var thou10_w;
        table region=''*county='', fiscal_year=''*(n='No.'*f=5.0
thou10_w='Rate'*sum=''*f=4.1)/row=float;
run;
proc tabulate data=rgn_pw91;
        class region fiscal_year;
        var thou10_w;
        table region='', fiscal_year=''*(n='No.'*f=5.0
thou10_w='Rate'*sum=''*f=4.1)/row=float;
run;
proc tabulate data=prov_pw91;
        class fiscal_year;
        var thou10_w;
        table all='Ontario', fiscal_year=''*(n='No.'*f=comma5.0
thou10_w='Rate'*sum=''*f=4.1)/row=float;
run;
ods chtml close;
ods listing;
```

Output

Total Number and Age-Standardized Death Rates pe 10,000 Population by county Ontario, 2002-2003 to 2006-2007

Injury Deaths Rate per 10,000 Population by County/Region/District of Injury - Ontario, Fiscal Year 2002-2003 to 2006-2007

By County		2002		2003		2004		2005		2006	
		No.	Rate	No.	Rate	No.	Rate	No.	Rate	No.	Rate
Central East	Durham R.M.	184	3.7	171	3.3	166	3	182	3.1	174	3
	Haliburton	16	9.8	15	10.7	18	8.2	17	7	18	9.7
	Northumberland	38	4.1	33	3.3	42	5.1	21	2.4	48	5.4
	Peterborough	63	4.4	74	4.6	50	2.9	69	4.5	56	3.5
	Simcoe	130	3.2	136	3.2	127	2.9	175	3.8	173	3.7
	Victoria	41	4.7	28	3.9	45	5.5	46	5.1	44	4
	York R.M.	158	2.1	150	1.9	181	2.1	172	2	191	2.1
Central South	Brant	47	3.6	46	3.1	54	3.9	60	4.3	58	4.1
	Haldimand-Norfo	41	3.8	42	3.4	35	3.2	62	5.6	56	4.1
	Hamilton-Wentwo	176	3.1	182	3.1	181	3	235	3.8	210	3.3
	Niagara	155	3.5	141	2.9	196	4.2	154	3.3	180	3.5
Central West	Dufferin	21	4.6	25	4.8	22	4.1	27	4.9	20	3.4
	Halton R.M	92	2.2	141	3.2	137	3	135	2.8	137	2.6
	Peel R.M.	231	2.4	217	2.1	242	2.4	242	2.3	261	2.3
	Waterloo R.M.	120	2.6	136	2.8	116	2.3	132	2.6	121	2.4
	Wellington	64	3.1	63	3.1	51	2.3	65	2.9	77	3.5
East	Frontenac	51	3.3	62	4	54	3.2	55	3.4	63	3.7
	Hastings	47	3.2	47	3.4	53	3.7	55	3.7	52	3.3
	Lanark	21	3.4	23	3.1	23	3.5	20	2.9	21	2.8
	Leeds & Grenvil	27	2.3	30	2.4	46	4.1	55	4.5	45	4.1
	Lennox & Adding	4	0.9	12	2.7	15	3.7	14	3.5	16	3.3
	Ottawa-Carleton	152	1.8	182	2.1	154	1.7	168	1.9		
	Prescott &Russe	26	3.4	32	4	25	2.8	22	2.9	31	
	Prince Edward	10	4	14	4.2	8	2.7	10	4.3		
	Renfrew	38	3.8	40	4.2	34			3.5	40	4.1
	Stormont, Dunda	47	3.8	56	4.8	50		56	5	76	5.8
North	Algoma District	42	3.2	53	4.3	56	4	54	5	60	4.5
	Cochrane Distri	42	4.9	36	4.3	39	4.8	37	4.1	37	4.2
	Kenora District	86	12.6	81	12.3	82	12.8	74	11.7	75	12
	Manitoulin Dist	10	7.5	9	7.4	9	7.3	15	13.2	14	8.8
	Muskoka Distric	32	5	37	6.1	55	9.3	40	6.2	41	6.6
	Nipissing Distr	43	5	41	4	47	4.7	47	5	54	5.6
	Parry Sound Dis	38	7.7	49	10.8	33	7.7	33	8.6	41	8.6
	Rainy River Dis	12	5.5	16	6.7	15	6.6	7	2.7	12	5.3
	Sudbury Distric	26	11.7	17	7.3	26	11.6	22	9.3	20	9.5
	Sudbury R.M.	61	3.6	84	4.6	60	3.3	57	3.2	69	3.5
	Thunder Bay Dis	60	3.7	74	4.6	82	4.8	78	4.6	75	4.5
	Timiskaming Dis	20	5.6	20	5.9	26	7.7	18	5	19	4.3

Done

Thank You

Acknowledgement
Yingbo Na
Sr. Analyst, CIHI