

Déficit de centros educativos rurales en educación Secundaria

Definición: Número de centros educativos del área rural que se requiere debido a la existencia de demanda de educación inicial, primaria o secundaria pero se carece de un centro educativo del nivel correspondiente.

Propósito: Informar sobre la gestión de la oferta educativa en el área rural.

Fuente: Bases de datos del Censo Escolar y Padrón de Instituciones Educativas del Ministerio de Educación-Unidad de Estadística Educativa, disponibles en <http://escale.minedu.gob.pe/escale>.

Método de cálculo:

Nivel Secundaria: A partir del cuadro 501-Resultado del ejercicio educativo del año anterior, por grado de estudio y sexo, de la cédula nº 3 del Censo Escolar, se obtiene la cantidad de alumnos que aprobó (campo "tipdato" = 1 de la base de datos) el 6º grado de primaria el año anterior. Se crean dos campos que identifiquen si el centro educativo es de primaria o secundaria, respectivamente. Tanto la cantidad de aprobados de 6º grado, como la cantidad de centros educativos de primaria y los de secundaria se suman para obtener la cantidad agregada a nivel de centro poblado. Se eliminan los casos de centros poblados duplicados para así contar con una base de datos de casos únicos. Se crea un campo que indique el tener o no tener centro de educación secundaria identificando además los casos que cumplan la condición de pertenencia a área rural, tener al menos 20 alumnos que haya aprobado 6º grado de primaria, contar con al menos un centro educativo de primaria (campo de identificación de primaria no vacío) y con ninguno en secundaria (campo de identificación de secundaria vacío). Se halla una distribución de frecuencias de dicho campo, distinguiendo entre las alternativas: 1=tiene centro educativo de secundaria, 2=no tiene centro educativo de secundaria y necesita, 3=no tiene centro educativo de secundaria pero no necesita. El indicador lo da el resultado de la opción 2.

RESULTADO	TOTAL		GRADO DE ESTUDIO											
			1º		2º		3º		4º		5º		6º	
	H	M	H	M	H	M	H	M	H	M	H	M	H	M
TOTAL MATRÍCULA al 30-11-2008														
APROBADOS TOTAL ^{1/}														
Aprobados al 31-12-2008														
Aprobados en Recup. al 31-03-2009														
DESAPROBADOS TOTAL														
Desaprobados al 31-12-2008														
Desaprobados en Recup. al 31-03-2009														
RETIRADOS ^{2/} al 30-11-2008														
FALLECIDOS al 30-11-2008														

1/ Aprobados total : Alumnos que al término del ejercicio educativo 2008 fueron considerados aptos para ser promovidos al grado inmediato superior.

2/ Retirados : Alumnos que durante el año académico 2008 sobrepasaron el límite de inasistencias y fueron considerados no aptos para la evaluación final. No incluir a los alumnos trasladados a otra institución educativa.

Sintaxis:

***SECUNDARIA**

```
GET TRANSLATE
FILE='D:\Bases\Padrón\archivo.DBF'
/TYPE=DBF /MAP .
SORT CASES BY
id (A) .
SAVE OUTFILE='D:\Bases\Padrón\archivo.sav'
/COMPRESSED.
```

```
GET TRANSLATE
FILE='D:\Bases\Censo\Res5000.dbf'
/TYPE=DBF /MAP .
```

```
FILTER OFF.
USE ALL.
SELECT IF(cuadro = "501" & tipdato = "1" & (niv_mod = "B0" | niv_mod = "F0")).
EXECUTE .
STRING id (A8).
COMPUTE id = CONCAT(cod_mod,anexo) .
EXECUTE .
SORT CASES BY
id (A) .
FREQUENCIES
VARIABLES=id /FORMAT=NOTABLE
/ORDER= ANALYSIS .
MATCH FILES /FILE=*
/TABLE='D:\Bases\Padrón\archivo.sav'
/RENAME (anexo area_sig catcalle cen_edu cod_area cod_car cod_mod cod_tur
codccpp codgeo codlocal
codooii D_R des_mot des_tipo desnutinei dir_cen director email estado fechareg
fechares fecharet
formas ges_dep ges_parr gescong lenguaebi moti_res niv_mod nrocalle nroruc
num_res pagweb pobreza
pobrserv priaico pridain prifape progarti progdist progebi pubmuni pubosec
pubsedu region registro
saanee servicios situadir telefono tipo_res tipoebi tipofina tipogpar tipoiiee
tiponee tipoprolog
tipssexo tservadi unidcost = d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 d19 d20 d21 d22
d23 d24 d25 d26 d27 d28
d29 d30 d31 d32 d33 d34 d35 d36 d37 d38 d39 d40 d41 d42 d43 d44 d45 d46
d47 d48 d49 d50 d51 d52 d53
d54 d55 d56 d57 d58 d59 d60 d61 d62 d63 d64 d65 d66 d67 d68 d69)
/BY id
/DROP= d0 d1 d2 d3 d4 d5 d6 d7 d8 d9 d19 d20 d21 d22 d23 d24 d25 d26 d27
d28 d29 d30 d31 d32 d33
d34 d35 d36 d37 d38 d39 d40 d41 d42 d43 d44 d45 d46 d47 d48 d49 d50 d51
d52 d53 d54 d55 d56 d57 d58
d59 d60 d61 d62 d63 d64 d65 d66 d67 d68 d69.
EXECUTE.
FREQUENCIES
VARIABLES=id /FORMAT=NOTABLE
/ORDER= ANALYSIS .
COMPUTE dato06hm = D11 + D12 .
EXECUTE .
IF (niv_mod = "B0") B0 = 1 .
EXECUTE .
```

```

IF (niv_mod = "F0") F0 = 1 .
EXECUTE .
USE ALL.
COMPUTE filter_$=(area_sig = "2" & codcpsig > "0").
VARIABLE LABEL filter_$ 'area = "2" & codcpsig > "0" (FILTER)'.
VALUE LABELS filter_$ 0 'Not Selected' 1 'Selected'.
FORMAT filter_$ (f1.0).
FILTER BY filter_$.
EXECUTE .
AGGREGATE
  /OUTFILE=*
  /BREAK=codcpsig cen_pob region codooii codgeo
  /dato06hm_sum = SUM(dato06hm) /B0_sum = SUM(B0) /F0_sum = SUM(F0).
AGGREGATE
  /OUTFILE=*
  MODE=ADDVARIABLES
  /BREAK=codcpsig
  /dato06hm_sum_sum = SUM(dato06hm_sum) /B0_sum_sum = SUM(B0_sum)
  /F0_sum_sum = SUM(F0_sum).
* Identify Duplicate Cases.
SORT CASES BY codcpsig(A) .
MATCH FILES /FILE = * /BY codcpsig
  /FIRST = PrimaryFirst /LAST = PrimaryLast.
DO IF (PrimaryFirst).
  COMPUTE MatchSequence = 1 - PrimaryLast.
ELSE.
  COMPUTE MatchSequence = MatchSequence + 1.
END IF.
LEAVE MatchSequence.
FORMAT MatchSequence (f7).
COMPUTE InDupGrp = MatchSequence > 0.
SORT CASES InDupGrp(D).
MATCH FILES /FILE = * /DROP = PrimaryFirst InDupGrp MatchSequence.
VARIABLE LABELS PrimaryLast 'Indicator of each last matching case as Primary' .
VALUE LABELS PrimaryLast 0 'Duplicate Case' 1 'Primary Case'.
VARIABLE LEVEL PrimaryLast (ORDINAL).
FREQUENCIES VARIABLES = PrimaryLast .
EXECUTE.
FILTER OFF.
USE ALL.
SELECT IF(PrimaryLast = 1).
EXECUTE .

IF (dato06hm_sum_sum >= 20 & B0_sum_sum >= 1 & SYSMIS(F0_sum_sum))
  tieneF0 = 2 .
EXECUTE .

```

*El código 2 es que no tiene Secundaria (F0)

*Se recodifica 1=tiene Secundaria, 2= No tiene Secundaria y necesita, 3=No tiene Secundaria pero no necesita porque hay 0 aprobados

```

DO IF (F0_sum_sum >= 1) .
RECODE
  tieneF0 (SYSMIS=1) .
END IF .
EXECUTE .

```

```
DO IF (dato06hm_sum_sum >= 20 & B0_sum_sum >= 1 &
SYSMIS(F0_sum_sum)) .
RECODE
  tieneF0 (SYSMIS=2) .
END IF .
EXECUTE .

RECODE
  tieneF0 (SYSMIS=3) .
EXECUTE .

FREQUENCIES
  VARIABLES=tieneF0
  /ORDER= ANALYSIS .
SORT CASES BY region .
SPLIT FILE
  LAYERED BY region .
FREQUENCIES
  VARIABLES=tieneF0
  /ORDER= ANALYSIS .
SPLIT FILE
  OFF.
FILTER OFF.
USE ALL.
EXECUTE.
```