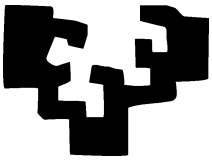
eman ta zabal zazu



Universidad del País Vasco
Euskal Herriko Unibertsitatea

Seminar 3

1 Synopsis.

What we are set out to do. In this seminar, we will turn again to the use micro data from the Estadística de Condiciones de Vida, or Living Conditions Survey (ECV) of 2004. You have used this data in the previous quarter[1]. In case you have missplaced your notes, a refresher on how to manipulate this micro-data is included as an appendix.

In Estadística y Análisis de Datos you had to limit yourselves to descriptive analysis of the data. Now we can go one step further, and use our newly acquired skills on statistical hypothesis testing. In particular, we will learn:

1. How to investigate association between categorical (or qualitative) variables using chi-square tests.
2. How to use Fisher's exact or permutation tests in cases where the assumptions underlying the chi-square test or the ordinary t-test are not met.
3. When to use a test of independence (or homogeneity of proportions) or a one-sided test on a binomial proportion.

Fisher's exact and permutation tests are particularly instructive, in that statistical testing principles appear in all their simplicity, without distractions arising from the need to obtain distributions of statistics, etc.

What you need to know. In order to benefit from this activity you need:

1. To be acquainted with the basics of hypothesis testing theory, and in particular to know how to perform a chi-square test and a two-sample t-test.
2. To have a basic command of R and its use in simulation, which you should have gained from Activities (and Seminars) 1 and 2.

[1] In Estadística y Análisis de Datos, Actividad 1.

2 Conducting hypothesis tests on contingency tables

2.1 The chi-square test.

Assume that we have built the following contingency table `t1` (turn to Appendix A to see how this is done):

```
> t1
```

```
          SitEconObj
TamFamilia Mala Normal Buena
  1           0    414   349
  2         499    581   261
  3-5       392   1241  1123
  6-9        67     55     0
  10 o más    2      0     0
```

In order to to test independence among size of the family and objective (as opposed to self-assessed) economic situation of the families surveyed, we could do:

```
> res.t1 <- loglin(t1,margin=list(1,2),fit=TRUE)
```

```
2 iterations: deviation 4.547474e-13
```

```
> res.t1
```

```
$lrt
[1] 826.3875

$pearson
[1] 698.004

$df
[1] 8

$margin
$margin[[1]]
[1] "TamFamilia"

$margin[[2]]
[1] "SitEconObj"


$fit
          SitEconObj
TamFamilia         Mala       Normal        Buena
  1         146.9662921  350.7289326  265.3047753
  2         258.2985554  616.4187400  466.2827047
  3-5       530.8507223 1266.8531300  958.2961477
  6-9        23.4991974   56.0798555   42.4209470
  10 o más    0.3852327    0.9193419    0.6954254
```

```
> 1 - pchisq(res.t1$pearson,res.t1$df)

[1] 0
```

Function `loglin` fits the model of independence[2] $p_{ij} = p_{i.} \times p_{.j}$; the argument `margin=list(1,2)` tells the function that we want the probability of each cell to be fitted as the product of the two marginals. The chi-square statistic[3] is in component `pearson` of the returned object, and the degrees of freedom in component `df`. The component `lrt` is the likelihood ratio test statistic, which follows the same asymptotic distribution as the chi-square statistic and can be used interchangeably with it. The p-value shows, as was to be expected, that size of the family and self-assessed economic situation are far from independent[4].

> **Remark 1.** We said that we are fitting "the model of independence". This is not quite correct. It would be strictly true if Instituto Vasco de Estadística (EUSTAT) were picking families entirely at random, and classifying them according to the two characters (size and economic situation). As it happens, sampling is usually stratified, so the proportion of large (or small) families may deliberately deviate from the true proportion in the population. They sample each size stratum as if it were a different population. why, we will see in a few weeks.
>
> The result is that what we test here is rather homogeneity of proportions. But all computations (and the result) are the same as if we were doing a genuine independence test.
>
> Why care, then? Because the type of sampling matters! If EUSTAT were sampling a single population, then
>
> ```
> > 100 * sum(t1[1,]) / sum(t1)
>
> [1] 15.30899
> ```
>
> would be a sensible estimate of the proportion of families of size 1 in the population. As things go, however, that figure is merely the percentage of families of size 1 that they have chosen to interview.

2.2 Fisher's exact test

Sometimes, the sample size is much too small for the asymptotic theory on which the chi-square test is built to apply. An alternative is Fisher's exact test. Consider the following 2×2 table[5]:

```
> TeaTasting <- matrix(c(3, 1, 1, 3),2,2)
> dimnames(TeaTasting) <- list(Guess = c("Milk", "Tea"),
                               Truth = c("Milk", "Tea"))
> TeaTasting
```

[2] As a special case of so-called log-linear models, hence the name of the function. The study of log-linear models in its full generality is way beyond the scope of this course; a leisurely introduction is Fienberg (1980).

[3] That is, $\sum_{i=1}^{k}(O_k - E_k)^2/E_k$, where O_k and E_k are the observed and expected counts in the k-th cell.

[4] Some cells are empty or nearly empty, which would cast some doubts on the adequacy of test: the departure from the independence hypothesis is of such magnitude, though, that one could hardly suspect inadequate the rejection of the null.

[5] It is a famous example, with observations from a lady who claimed she was able to distinguish whether milk or tea had been poured first in the cup. You can read the story if you wish in Senn (2012).

```
      Truth
Guess  Milk Tea
  Milk    3   1
  Tea     1   3
```

We could construct all tables with the same marginals, and compute their probabilities under the independence hypothesis. If the observed table were among the $100\alpha\%$ rarest, we would reject the hypothesis of independence. For the case at hand, we would have to consider all the tables (below we write their probability under the independence hypothesis and given margins):

4	0
0	4

$p = 0.0143$

3	1
1	3

$p = 0.2286$

2	2
2	2

$p = 0.5142$

1	3
3	1

$p = 0.2286$

0	4
4	0

$p = 0.0143$

We see that the rarest tables under independence are the first and last, and the most likely the third. The second, which we have observed, is among the "rarest" totalling

$$0.0143 + 0.0143 + 0.2286 + 0.2286 = 0.4858$$

in probability; so we reason that there is probability as large as 0.4858 of observing what we have observed or something even more remote from independence. This is a large probability and therefore we would not reject independence.

Remark 2. The fact that we enumerate all results compatible with the given margins gives us a greater control on how we choose the critical region. It would make sense to consider as H_0 the hypothesis "the lady cannot guess correctly", and reject that hypothesis only if the lady is on target *more often* than could be expected by chance (as she claims she is able to guess correctly). In that case, we would not include in our "critical region" the fourth and fifth tables, as they cannot be considered evidence against H_0 in the sense claimed.

Remark 3. The contingency table is far too unpopulated to warrant the use of a regular chi-square test. If, however, we insist on using it, we obtain a grossly inaccurate p-value:

```
> res <- loglin(TeaTasting,margin=list(1,2),fit=TRUE)

1 iterations: deviation 0

> 1 - pchisq(res$pearson,res$df)

[1] 0.1572992
```

The probabilities under each table are computed easily under independence, but we do not need to do it: the R instruction `fisher.test` does everything for us.

```
> fisher.test(TeaTasting)
```

```
        Fisher's Exact Test for Count Data

data:  TeaTasting
p-value = 0.4857
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
   0.2117329 621.9337505
sample estimates:
odds ratio
  6.408309
```

The p-value of 0.4857 corresponds to our 0.4858; the discrepancy in the last digit is rounding error. If we prefer to test against a one-sided alternative (as would make more sense here, see Remark 2), we would instead do:

```
> fisher.test(TeaTasting,alternative="greater")

        Fisher's Exact Test for Count Data

data:  TeaTasting
p-value = 0.2429
alternative hypothesis: true odds ratio is greater than 1
95 percent confidence interval:
 0.3135693       Inf
sample estimates:
odds ratio
  6.408309
```

The reported p-value is the sum of 0.2286 and 0.0143, corresponding to the first and second tables above.

The test is simple and there are no statistics with distribution to be approximated (like in the chi-square test); however, it is an unfeasible computer task for large tables.

2.3 Permutation tests for the equality of means

Much the same idea is used in permutation tests, that can be used as an alternative to approximate tests based on normal approximations.

If we have two groups for each of which we have estimated a sample mean (or a sample proportion of subjects with a certain character), we may be interested in testing whether those means (or proportions) are significantly different. The following simple procedure might be used:

1. Compute the difference of means (or proportions) for the two groups.
2. Pool the data of the two groups and repeatedly draw samples at random, of the same size as the original groups. Since the samples are taken at random, the differences of means (or proportions) observed give us an idea of what can be expected from sampling variability alone.
3. Now, compare the difference among the means (or proportions) of the two real groups, and decide if it looks plausible that it is due to chance alone.

This is quite easy to implement in R, once you are familiar with loops: we show next how to proceed with two artificially generated groups:

```
> G1 <- runif(50)
> G2 <- runif(80) + 0.2
```

`G1` has mean 0.5 and $\overline{X}_1 = 0.5188$ while `G2` has mean 0.7 and $\overline{X}_2 = 0.69316$; we are left with the task of deciding if $\overline{X}_1 - \overline{X}_2$ is significantly different from zero. Pool `G1` and `G2` into a single group `G`:

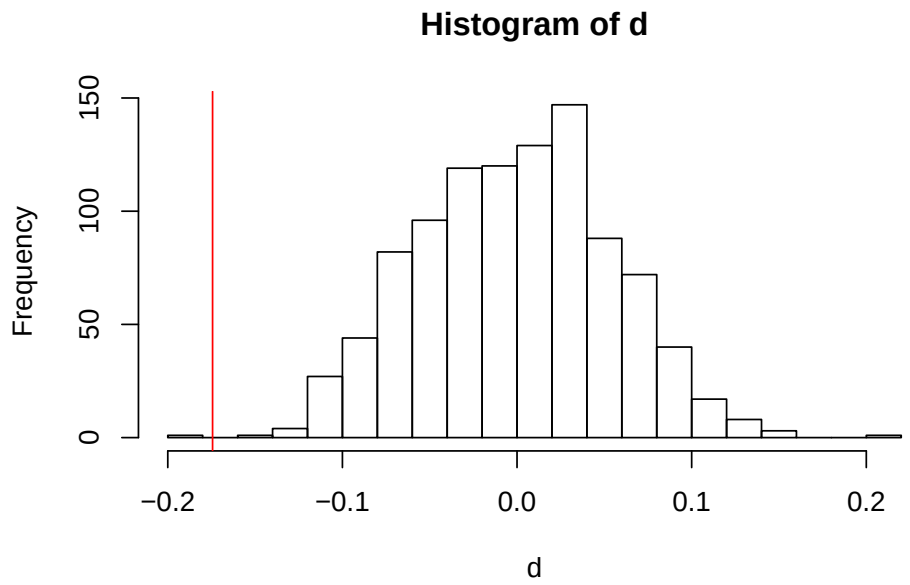
```
> G <- c(G1,G2)
```

Now we are going to pick randomly $n = 999$ times sets of 50 and 80 observations from `G` and see what is the the value of $\overline{X}_1 - \overline{X}_2$ each time.

```
> n <- 999 ; d <- rep(0,n)
> k1 <- length(G1) ; k2 <- length(G2) ; k <- k1 + k2
> for (i in 1:n) {
    j     <- sample(1:k,k1)
    Xbar1 <- mean(G[j])
    Xbar2 <- mean(G[-j])
    d[i]  <- Xbar1 - Xbar2
  }
```

We can now plot the histogram of the values in `d` to obtain:

```
> hist(d,breaks=15)
> abline(v=mean(G1)-mean(G2),col="red")
```



These are the sort of values for $\overline{X}_1 - \overline{X}_2$ we would expect when sampling 50 and 80 observation from the same population. The red line shows the location of $\overline{X}_1 - \overline{X}_2$ for the two original groups; we see that it is in a fairly extreme position, which could hardly have arisen by chance: we conclude that the two groups have different means (which we know to be true, since we generated the data artificially).

If we want to compute an approximate significance level, we can do it as follows:

```
> sum(abs(d) > abs(mean(G1) - mean(G2)))

[1] 2
```

We see that only 2 out of 1000 random values of the statistic are greater in absolute value than the observed $|\overline{X}_1 - \overline{X}_2|$, so the p-value would be approximately 0.003 (the observed value of our statistic is among the 3 rarest values out of 1000).

Even though it is so easy to perform a permutation test, there are canned functions in R that help you do it. One of them is in package `DAAG` and can be invoked as (output not shown):

```
> library(DAAG)
> twot.permutation(G1,G2,n=999)
```

2.4 The classical two-sample t-test.

You should have no problem at all computing the test statistic and checking its significance. You may be interested though in knowing that you have function `t.test` with a wealth of options, that will make testing for equality of means a snap.

3 Questions

We list a sample of questions; please feel free to peruse the survey and answer any others that are more appealing to you, as long as you practice with the all the different tests you are required to use.

1. Does the structure of occupations (as reflected by variable V15 of the ECV) look the same across Territories? What is the approximate probability that the differences observed are a mere outcome of chance? What professional categories appear to be over- or underrepresented in each Territory?

2. Consider now the variables describing origin (V11), instruction level (V12), occupational status (V13), and existence of links with family and relatives (V17).

   (a) Convert into factors and label the relevant variables, so that you obtain self-explanatory output.

   (b) For those of foreign origin (V11 equal to 5), is there any relationship between level of instruction and existence of links with family?

   (c) For the same group, is there any relationship between occupational status and existence of links with family?

   Answer both questions using an ordinary chi-square test and Fisher's exact test, compare the p-values and argue which test seems more adequate. (HINT: Look for tenuously populated cells.)

3. Compute a "household equipment index" such as in Section A.5 above (you might want to make it more comprehensive, or include different items). Then,
   (a) Test equality of index means among two Territories, by means of an ordinary t-test.
   (b) Which assumptions for the above test might have been violated? Explain.
   (c) Perform a permutation test of equality of index means. Compare with the standard theory t-test.

4 If you want to know more:

For questions related to R manipulation you may ask your instructor or consult any of the many good books or on-line available documents on the subject: among them, Kuhnert and Venables (2005), Spector (2008), Ugarte et al. (2008) or Dalgaard (2002).

For questions related to permutation tests, you might want to use Good (1993). For questions on how to extend chi-square tests towards more general log-linear models, Fienberg (1980), Bishop et al. (1975), Plackett (1974) or Agresti (1990) (roughly in increasing order of sophistication; the first has a level compatible with this course, the others are more advanced).

In answering question 3 above you might wonder whether it would be possible to test equality of means of the equipment index in all *three* Territories. This would be an Analysis of Variance (ANOVA) problem; you might consult Peña (2002) or (with much more detail, but harder to tackle) Seber and Lee (1998).

For further information on the ECV you might turn to EUSTAT (2008), a collection of essays available at EUSTAT's web site. For a recent review of similar surveys, you might want to check the issue 51 of Índice (www.revistaindice.com), an on-line resource from the Instituto Nacional de Estadística (INE).

A Apendix

A.1 The Living Conditions Survey.

The ECV is a survey conducted in the Basque Country by EUSTAT with hyper-annual frequency. It serves the purpose of investigating a number of factors affecting the well-being of the population; a perusal of the file `Estructura_familias_ECV04.xls` will give you an idea of the questions asked; the raw data appears in the file `ecv04_familias.dat` and the answer codes in `Metadata_familias_ECV04.doc`.

Notice that we are referring to the "family view" of the survey; there is also information related to individuals, that we do not use here. Notice also that you may find similar information compiled by the INE for the whole of Spain in `http://www.ine.es/en/inebmenu/mnu_nivel_vida_en.htm`.

A.2 How to read and set up fixed format data

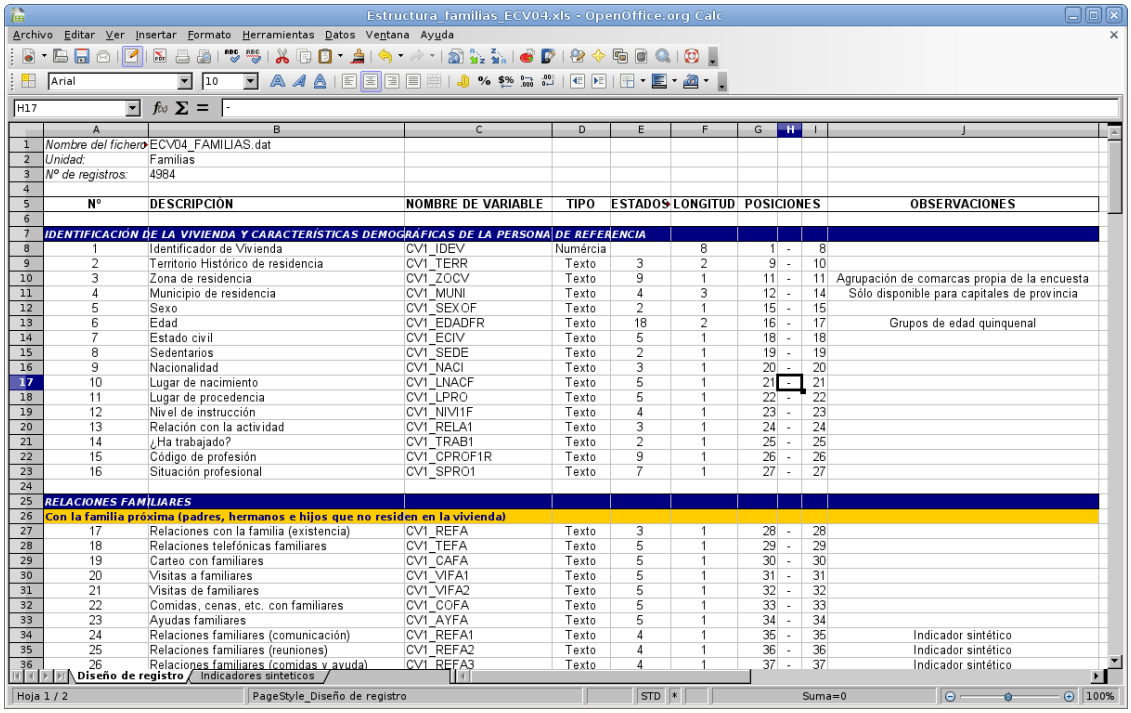
We could supply the date under the form of a native R file, or an Excel-compatible spreadsheet; since our goal is to empower you to find your way in any situation, not just the present one, we will demonstrate how to obtain the data from the raw, fixed-format file provided by EUSTAT[6].

[6] Fixed-format files take relatively little space, are easy to deal with and readable across computers and operating systems; hence their widespread use.

In fixed-format files, data are presented as strings with (usually) no intervening spaces among different data fields. Here is the first line from file `ecv04_familias.dat`:

```
760011059110221444212431533351221322444513313222225111 12 4344111111111 113122221321121222122135131111↩
2214111220100232242311221222221111131111111111111111321 0122201112 70.054
```

Although we have broken it to make it fit within the limits of the printed text, it is one long line (the symbol ↩ means "line continued below"). We can tell where each value begins and ends by referring to the register description in file `Estructura_familias_ECV04.xls` (refer to Figure 1, p. 9).



Estructura_familias_ECV04.xls - OpenOffice.org Calc

	A	B	C	D	E	F	G	H	I	J
1	*Nombre del fichero*	ECV04_FAMILIAS.dat								
2	*Unidad:*	Familias								
3	*Nº de registros:*	4984								
4										
5	**Nº**	**DESCRIPCIÓN**	**NOMBRE DE VARIABLE**	**TIPO**	**ESTADOS**	**LONGITUD**	**POSICIONES**			**OBSERVACIONES**
6										
7	***IDENTIFICACIÓN DE LA VIVIENDA Y CARACTERÍSTICAS DEMOGRÁFICAS DE LA PERSONA DE REFERENCIA***									
8	1	Identificador de Vivienda	CV1_IDEV	Numérica		8	1	-	8	
9	2	Territorio Histórico de residencia	CV1_TERR	Texto	3	2	9	-	10	
10	3	Zona de residencia	CV1_ZOCV	Texto	9	1	11	-	11	Agrupación de comarcas propia de la encuesta
11	4	Municipio de residencia	CV1_MUNI	Texto	4	3	12	-	14	Sólo disponible para capitales de provincia
12	5	Sexo	CV1_SEXOF	Texto	2	1	15	-	15	
13	6	Edad	CV1_EDADFR	Texto	18	2	16	-	17	Grupos de edad quinquenal
14	7	Estado civil	CV1_ECIV	Texto	5	1	18	-	18	
15	8	Sedentarios	CV1_SEDE	Texto	2	1	19	-	19	
16	9	Nacionalidad	CV1_NACI	Texto	3	1	20	-	20	
17	10	Lugar de nacimiento	CV1_LNACF	Texto	5	1	21	-	21	
18	11	Lugar de procedencia	CV1_LPRO	Texto	5	1	22	-	22	
19	12	Nivel de instrucción	CV1_NIVI1F	Texto	4	1	23	-	23	
20	13	Relación con la actividad	CV1_RELA1	Texto	3	1	24	-	24	
21	14	¿Ha trabajado?	CV1_TRAB1	Texto	2	1	25	-	25	
22	15	Código de profesión	CV1_CPROF1R	Texto	9	1	26	-	26	
23	16	Situación profesional	CV1_SPRO1	Texto	7	1	27	-	27	
24										
25	***RELACIONES FAMILIARES***									
26	**Con la familia próxima (padres, hermanos e hijos que no residen en la vivienda)**									
27	17	Relaciones con la familia (existencia)	CV1_REFA	Texto	3	1	28	-	28	
28	18	Relaciones telefónicas familiares	CV1_TEFA	Texto	5	1	29	-	29	
29	19	Carteo con familiares	CV1_CAFA	Texto	5	1	30	-	30	
30	20	Visitas a familiares	CV1_VIFA1	Texto	5	1	31	-	31	
31	21	Visitas de familiares	CV1_VIFA2	Texto	5	1	32	-	32	
32	22	Comidas, cenas, etc. con familiares	CV1_COFA	Texto	5	1	33	-	33	
33	23	Ayudas familiares	CV1_AYFA	Texto	5	1	34	-	34	
34	24	Relaciones familiares (comunicación)	CV1_REFA1	Texto	4	1	35	-	35	Indicador sintético
35	25	Relaciones familiares (reuniones)	CV1_REFA2	Texto	4	1	36	-	36	Indicador sintético
36	26	Relaciones familiares (comidas y ayuda)	CV1_REFA3	Texto	4	1	37	-	37	Indicador sintético

Figure 1: Snapshot of file `Estructura_familias_ECV04.xls`.

We see, for instance, that "Territorio Histórico de residencia" is coded with two characters, starting at position 9 and ending at position 10. Likewise, "Nationality" is coded with just one character starting at position 20, and so on. We can read the file directly into an R data frame typing:

```
> ecv04 <- read.fwf(file="ecv04_familias.dat",
                    widths=c(8,2,1,3,1,2,rep(1,166),8))
```

The instruction `read.fwf` reads with **f**ixed **w**idth **f**ormat, the widths of the fields being given in the argument `widths`. These are taken from file `Estructura_familias_ECV04.xls` (in column `F`, "Longitud"). (We are fortunate that most of the widths are 1, so we can abbreviate the typing of widths by repeating the value "one" 166 times!) If we now display the first few rows and columns of `ecv04` we get:

```
> ecv04[1:3,1:18]
```

```
    V1 TerHist V3 V4 V5 V6 V7 V8 V9 V10 V11 V12 V13 V14 V15 V16 V17 V18
1  760   Araba  1 59  1 10  2  2  1   4   4   4   2   1   2   4   3   1
2  792   Araba  1 59  1  7  1  2  3   5   5   3   1   1   5   6   3   2
3 1177   Araba  1 59  1 10  2  2  1   3   3   4   1   1   7   2   3   1
```

By default variables are named `V1`, `V2`, etc. We may assign more mnemonic names to some or all of them, which will make for easier-to-read output. For instance,

```
> cols <- colnames(ecv04)
> cols[2] <- "TerHist"
> cols[170:172] <- c("SitEconSub","SitEconObj","TamFamilia")
> colnames(ecv04) <- cols
```

We can also convert numeric variables to qualitative (or categorical) variables, *factors* in R parlance. On our way, we can label the different codes with something meaningful:

```
> ecv04[,"TerHist"] <- factor(ecv04[,"TerHist"],
                              labels=c("Araba","Gipuzkoa","Vizcaya"))
> ecv04[,"SitEconSub"] <- factor(ecv04[,"SitEconSub"],
                              labels=c("Mala","Normal","Buena"))
> ecv04[,"SitEconObj"] <- factor(ecv04[,"SitEconObj"],
                              labels=c("Mala","Normal","Buena"))
> ecv04[,"TamFamilia"]     <- factor(ecv04[,"TamFamilia"],
                              labels=c("1 ","2 ","3-5 ","6-9 ","10 o más"))
```

A.3 How to cross-tabulate data

Doing cross-tabulations is now as easy as:

```
> t1 <- with(ecv04,
             table(TamFamilia,SitEconObj)
             )
> t1
```

```
          SitEconObj
TamFamilia Mala Normal Buena
  1           0    414   349
  2         499    581   261
  3-5       392   1241  1123
  6-9        67     55     0
  10 o más    2      0     0
```

A.4 How to select subgroups from your data

You may be interested in picking all cases from Araba, or all cases in which the family does not have heating, or a washing machine, or whatever. It is quite easy. For instance, if we want all cases from Araba, we would do:

```
> sel     <- ecv04[,"TerHist"] == "Araba"
> arabako <- ecv04[sel,]
```

If you want all people who do have a phone but not a washing machine, you could do[7]:

```
> sel <- ( ecv04[,81] == 1 )  & ( ecv04[,84] == 2 )
> PhoneNoWash <- ecv04[sel,]
```

A.5 How to operate on columns of a data frame and construct new variables

Sometimes you may need to build a variable which is not present in your data. For instance, we have in the data frame `ecv04` a number of categorical variables telling us whether a family owns some equipment: air conditioning (V74; refer to the metadata file), heating (V78), phone (V81), refrigerator (V83), washing machine (V84) and dishwasher (V85).

Suppose we want to construct an index whose value is the number of the items above present in the household[8]. One way would be to type[9]:

```
> index <- apply(ecv04[,c(74,78,81,83:85)]==1, 1, sum)
```

and then we can "add" this new variable to the data frame easily:

```
> ecv04 <- cbind(ecv04,index)
```

References

A. Agresti. *Categorical Data Analysis*. Wiley, 1990.

Y. M. M. Bishop, S. E. Fienberg, and P. W. Holland. *Discrete Multivariate Analysis. Theory and Practice*. MIT Press, Cambridge, Mass., 1975.

P. Dalgaard. *Introductory Statistics with R*. Statistics and Computing. Springer-Verlag, 2002. Signatura: 519.682 DAL.

EUSTAT, editor. *Encuesta de Condiciones de Vida. Monográficos. 2004*. Eustat, 2008. ISBN 978-84-7749-451-5.

S. E. Fienberg. *The Analysis of Cross-Classified Categorical Data*. MIT Press, Cambridge, Mass., 1980.

P. Good. *Permutation Tests*. Springer-Verlag, 1993.

P. Kuhnert and W. Venables. *An Introduction to R: Software for Statistical Modelling and Computing*. CSIRO Mathematical and Information Sciences, Cleveland, Australia, 2005.

D. Peña. *Regresión y Diseño de Experimentos*. Alianza Editorial, 2002.

R. L. Plackett. *The Analysis of Categorical Data*. Griffin, London, 1974.

G. A. F. Seber and A. J. Lee. *Linear Regression Analysis*. Wiley, 1998.

S. Senn. Tea for three. *Significance*, 9:30–33, 2012.

Phil Spector. *Data Manipulation with R*. Springer, 2008. doi: 10.1663/978-0-387-74731-6.

M.D. Ugarte, A.F. Militino, and A.T. Arnholt. *Probability and Statistics with R*. CRC Press, 2008.

[7] Check the register description and metadata files to locate the questions carrying the relevant information.
[8] We might want to apply weights, but for now will keep things simple and give each item a weight of 1.
[9] You may want to check the help file for `apply`.