CHAPTER 10

Association measures

Collocations and collostructions

What will you learn from this chapter:

Collocations, as well as colligations and other co-occurrence patterns, play an important role in corpus linguistics, psycholinguistics and usage-based grammar and lexicology. To measure the degree of attraction between words and other units, one can use diverse association measures, such as collostructional strength, Pointwise Mutual Information or *ΔP*. From this chapter you will learn how to compute a variety of association measures using a small set of different co-occurrence frequencies. The case study is based on co-occurrence frequencies of different verbs in the Russian ditransitive construction.

10.1 Measures of association: A brief typology

10.1.1 Frequencies that you will need in order to compute association measures

There are dozens of possible measures of association proposed in the literature (e.g. Evert 2004). All of them require all or at least some values from a 2-by-2 contingency table shown in Table 10.1. For two words (or any other units) *X* and *Y*, these frequencies can be defined as follows:

- *a* corresponds to the number of co-occurrences of *X* and *Y*;
- *b* is the number of occurrences of *X* without *Y* (in all other contexts), i.e. the total number of occurrences of *X* minus *a*;
- *c* corresponds to the sum frequency of all other target words that co-occur with *Y*, i.e. the total frequency of *Y* minus *a*;
- *d* is the frequency of all units except *X* that occur with all units except *Y*. In practice, this is the total number of words, sentences or other units in the corpus minus *a*, *b* and *c*. A more specific definition of this total number depends on the research question.

Table 10.1 The main co-occurrence frequencies that are needed for measuring association between *X* and *Y*

	Unit *Y*	**¬ *Y* (all other units)**
Unit *X*	*a*	*b*
¬ *X* (all other units)	*c*	*d*

If you know these four frequencies, you can compute any popular association measure between words (lexical collocations), or words and constructions, as in collostructional analysis. These measures can be subdivided into different classes, which are presented in the following subsection.

10.1.2 Unidirectional (asymmetric) vs. bidirectional (symmetric) measures

The main difference between unidirectional (asymmetric) and bidirectional (symmetric) measures is that the former will change if the rows and columns in Table 10.1 are swapped, whereas the latter will remain the same. The most important unidirectional measures are the conditional probability of *X* given *Y* ($X|Y$) and, conversely, the conditional probability of *Y* given *X* ($Y|X$). Clearly, these probabilities may be very different, and relationships between different parts of collocations need not be symmetric (Michelbacher et al. 2011). For instance, the first component *bonsai* in *bonsai tree* suggests, or predicts, the second component *tree* better than the other way round. Consider the data in Table 10.2.

Table 10.2 Co-occurrence frequencies of word forms *bonsai* and *tree* (based on the British National Corpus)

	bonsai	**¬ *bonsai***
tree	*5*	*6131*
¬ *tree*	*50*	*96980521*

The conditional probability of *tree* given *bonsai* can be computed as the co-occurrence frequency of *bonsai* and *tree* divided by the total frequency of *bonsai*: 5/(50 + 5) = 0.09, or about 9%. The conditional probability of *bonsai* given *tree* is equal to their co-occurrence frequency divided by the total frequency of *tree*: 5/(6131 + 5) = 0.0008, or less than 0.1%. Thus, *bonsai* should be a much stronger clue for *tree* than the other way round. If one swaps the rows and columns in Table 10.1, the numbers will be reversed.

Unidirectional measures are also known in corpus-based Construction Grammar studies, where *X* is a word and *Y* is a construction. They are called Attraction (the

conditional probability of the lexeme given the construction, i.e. collexeme|construction), and Reliance, or Faith (the conditional probability of the construction given the lexeme, i.e. construction|collexeme) (Schmid 2000; Gries et al. 2005). Again, high Attraction does not imply high Reliance. Consider the verb *make.* One can expect it to be frequent in the *way*-construction, as in *She made her way to the bar*, but in comparison with all other uses of *make* its appearance in the *way*-construction should be very modest. That means that it will have a high Attraction score and a low Reliance score with regard to the *way*-construction.

An example of a bidirectional measure is the collostructional strength based on an independence test (Stefanowitsch & Gries 2003). The relationships between a construction and a collexeme (e.g. the verb *give* and the English ditransitive construction) are most commonly measured with the help of statistical independence tests (e.g. the Fisher exact test), which show whether the co-occurrence frequency is significantly different from what one could expect under the assumption of no association (the null hypothesis). Regardless of what serves as a cue, the construction or the collexeme, the attraction score is the same. For all bidirectional association measures, all four frequencies in the table are needed, and the outcome will remain the same if one swaps the rows and columns.

10.1.3 Contingency-based vs. non-contingency-based measures

Let us begin with a conceptual question. If a unit A, e.g. the word *apple*, is a cue that triggers a unit B, e.g. the word *orange*, does the frequency of A as a cue for other units C, D, E, etc. (e.g. *pear*, *cider*, *cinnamon*) influence its role as a trigger of B? Psychologists would argue that this information is important in category learning.

> Consider how, in the learning of the category of birds, while eyes and wings are equally frequently experienced features in the exemplars, it is wings which are distinctive in differentiating birds from other animals. Wings are important features to learning the category of birds because they are reliably associated with class membership, eyes are neither. Raw frequency of occurrence is less important than the contingency between the cue and interpretation. (Ellis & Ferreira-Junior 2009: 194)

For example, if the verb *make* is frequently used in the transitive construction *X makes Y*, but it is also frequently used in other contexts, e.g. X *makes Y Z,* as in *make me a cup of tea*, or in the periphrastic causative *X makes Y do Z*, as in *he made me do it*, does this cognitively 'devalue' in any way the strength of association between *make* and the transitive construction? Does this have an effect on learning of the transitive construction? Are speakers sensitive to such contingency information and to what extent? These are questions that are still open.

There is a vast inventory of measures that take into account contingency information (they will be referred here as contingency-based measures). All bidirectional measures

except for the simple co-occurrence of *X* and *Y* include contingency information. Some unidirectional measures contain it, as well, e.g. ΔP ('delta *P*'), a psychological cue-response measure introduced by Allan (1980) and recently used in constructionist studies by Ellis (2006) and Ellis and Ferreira-Junior (2009). See the next section for details.

In the remaining part of the chapter, you will learn how to compute different measures of association between constructions and collexemes. The object of the case study is the Russian ditransitive construction and its collexemes.

10.2 Case study: The Russian ditransitive construction and its collexemes

10.2.1 Theoretical background and data

To perform this case study, you will need several add-on packages. You should install and load them, if you have not done so yet.

```
> install.packages(c("ggplot2", "corrgram"))
> library(Rling); library(ggplot2); library(corrgram)
```

Ditransitive constructions exist in many languages. They normally denote transfer and consist of a verb of transfer and three arguments: the Agent (giver), the Recipient and the Theme (the object of transfer). The semantics of the construction can be very diverse. For example, transfer can be literal (e.g. *pass me the salt*) or metaphorical (*he told me a story*, *she gave him a punch*), actual (*he gave her a diamond ring*) or future (*he promised her a diamond ring*), positive (*she gave him an apple*) or negative (*it cost me a fortune*).

The Russian ditransitive construction has two objects, which correspond to the Recipient and Theme, which are marked with the Dative and Accusative case, respectively. It is very similar semantically to its English counterpart. The construction expresses events of transfer, and also has various semantic extensions that relate to transfer of information, future giving, and so on. Some of the unique extensions are malefactive events with external possession (*slomat' komu-to nogu,* lit. 'break somebody a leg') and what might be called 'causing to undergo' (*podvergat' kogo-to nakazaniju*, lit.: 'subject somebody to punishment'). Unlike the English ditransitive, the Russian construction is not used in some functions of negative transfer (e.g. cost or deny somebody something).

This case study will demonstrate how to compute some popular measures of association between the Russian ditransitive construction and its collexemes (e.g. *davat'* 'to give', *posylat'* 'to send', *darit'* 'to give as a gift'). The data come from a larger-scale study (Levshina, In preparation). The sample is available as the `ditr` data frame in the `Rling` package. It contains 47 verb lemmata, which constitute the rows of a data frame. The

variables (columns) are *Freq_VC*, which shows the frequency of a verb in the ditransitive construction in the syntactically parsed segment of the Russian National Corpus, and *Freq_V*, which represents the total frequency of a verb in the corpus.

```
> data(ditr)
> head(ditr)
                 Freq_VC   Freq_V
brat             1         443
darit            10        28
davat            131       682
demonstrirovat   5         73
govorit          6         1160
nahodit          1         333
```

10.2.2 Computation of some popular association measures

To compute association measures, one first has to derive the co-occurrence frequencies that are shown in Table 10.3, which is a customized version of Table 10.1 for this case study.

Table 10.3 Main types of co-occurrence frequencies of the ditransitive construction and its collexemes

	Ditransitive construction	**¬ Ditransitive construction (all other verbal constructions)**
Collexeme *X*	*a*	*b*
¬ Collexeme *X* (all other collexemes of the ditransitive construction)	*c*	*d*

Let us begin by creating four vectors that contain *a*, *b*, *c* and *d* counts for all verbs in the dataset. Some additional information is needed. First, the total frequency of the ditransitive construction in the corpus is 667. Second, the total number of verbs (104162) will be used as an approximation of the total frequency of all verbal constructions. This number is needed in order to obtain the frequencies *d*.

Note that the frequencies from the table should be computed for each of 47 collexemes. Computing them individually would be very time-consuming. Fortunately, one can easily apply arithmetic operations to all values in a vector:

```
> a <- ditr$Freq_VC
> b <- ditr$Freq_V - a
> c <- 667 - a
> d <- 104162 - (a + b + c)
```

For instance, the expression `667 - a` means that the algorithm subtracts each value in `a` (the corpus frequencies of each verb in the construction) from 667. The result is a vector of numbers:[1]

```
> head(c)
[1] 666 657 536 662 661 666
```

Some other useful measures can be derived from these four. The most important is the expected frequency of a verb in the construction. The expected frequency, which was discussed in the previous chapter, is the frequency that would be observed if the proportion of the verb in the construction were equal to the proportion of the verb in all other constructions. The vector of expected frequencies for all 47 verbs can be obtained as follows:

```
> aExp <- (a + b)*(a + c)/(a + b + c + d)
> head(aExp)
[1] 2.8367447 0.1792976 4.3671780 0.4674545 7.4280448 2.1323611
```

Now we are ready to compute a few popular association scores. The R code for them is given in Table 10.4. The measures vary with regard to the directionality of the relationships between words and constructions (unidirectional vs. bidirectional) and with regard to the use of contingency information (contingency-based vs. contingency-free). The simplest measures are Attraction and Reliance (Schmid 2000). In this case, Attraction is the relative frequency of a verb in the ditransitive construction based on all uses of the construction in the corpus. Reliance, in contrast, is the relative frequency of a verb in the ditransitive construction with regard to all uses of the given verb. Both measures are unidirectional and do not include contingency information. Using the frequencies from Table 10.3, one can compute Attraction and Reliance as follows:

$$Attraction = \frac{a}{a+c}$$

$$Reliance = \frac{a}{a+b}$$

For convenience, the measures can be expressed as percentages by multiplying the result by 100:

```
> Attr <- 100*a/(a+c)
> Rel <- 100*a/(a+b)
```

1. Note that there is a built-in function `c()` in R, which concatenates objects. However, R can disambiguate overlapping names of the user's objects and functions.

Let us look at the top five verbs, which are the most attracted to the construction. Since the created vectors only contain numbers, let us add the names (individual verbs) to the vector elements in order to facilitate interpretation:

```
> names(Attr) <- rownames(ditr)
> head(Attr)
  brat        darit       davat         demonstrirovat
  0.1499250   1.4992504   19.6401799    0.7496252
  govorit     nahodit
  0.8995502   0.1499250
> sort(Attr, decreasing = TRUE)[1:5]
  davat       pridavat  predlagat  otdavat    peredavat
  19.640180   3.598201  3.448276   3.298351   3.298351
```

The top verbs are the generic *davat'* 'to give', followed at a distance by the prefixal verbs meaning 'to attach', 'to offer', 'to give away' and 'to pass, transfer', respectively. Three of them contain the same root as *davat'* and represent a specification of the generic meaning.

The verb *davat'* 'to give' is clearly the leader. There is evidence that high-frequency collexemes play a special role in constructional acquisition. For example, Goldberg et al. (2004) have shown that the presence of one high-frequency collexeme in the input facilitates learning of a construction. According to Goldberg and her colleagues, high-frequency collexemes help language learners note a correlation between the meaning of the word and the construction itself. Presumably, *davat'* plays such a role in acquisition of the Russian ditransitive construction.

The top five Reliance scores look as follows:

```
> names(Rel) <- rownames(ditr)
> sort(Rel, decreasing = TRUE)[1:5]
podkidyvat  pridavat  odalzhivat  navjazyvat  prepodnosit
100.00000   68.57143  66.66667    53.33333    40.00000
```

The first verb has 100% Reliance because the only time it occurs in the corpus, it is used in the ditransitive construction. The translations of the five verbs are, respectively, 'to put, plant (often secretly)', 'to attach (importance), to give (taste)', 'to lend', 'to impose' and 'to present' (e.g. a gift or, metaphorically, a surprise). Only one verb, *pridavat'* 'to attach (importance), to give (taste)', is also in the top five Attraction scores. On the other hand, *davat'* 'to give', the verb with the highest Attraction score, has only the top eleventh Reliance score. This means that these measures represent very different kinds of information. Figure 10.1 demonstrates how the verbs are distributed with regard to Attraction and Reliance, with *davat'* being an outlier with its high Attraction score, and most verbs having low Attraction and Reliance scores. To create a plot with text labels, you first have to create an

empty plot (`type = "n"`) and then add text. The argument `cex = 0.7` specifies the font size.

```
> plot(Attr, Rel, type = "n", main = "Attraction and Reliance scores
of verbs in Russian Ditransitive Cx")
> text(Attr, Rel, rownames(ditr), cex = 0.7)
```

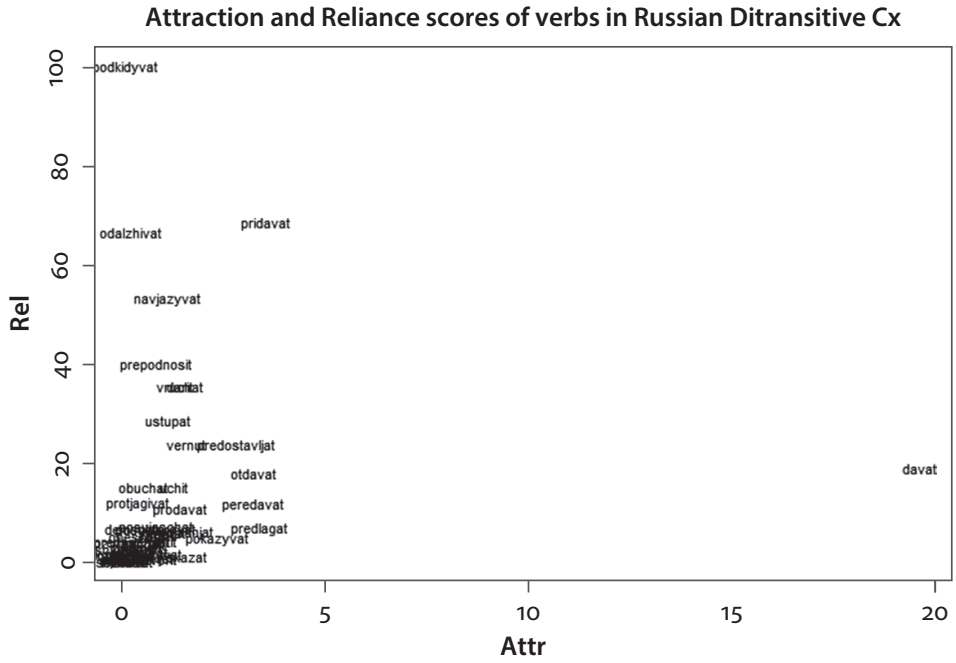
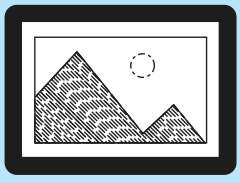


Figure 10.1. Attraction and Reliance scores



How to create a scatter plot with text labels using `ggplot2`

If you want to make a `ggplot2` version of the plot in Figure 10.1, you can use the code below. The result is shown in Figure 10.1a.

```
> ggplot(data.frame(Attr, Rel), aes(x = Attr, y = Rel)) + geom_
text(label = names(Attr), size = 3)
```

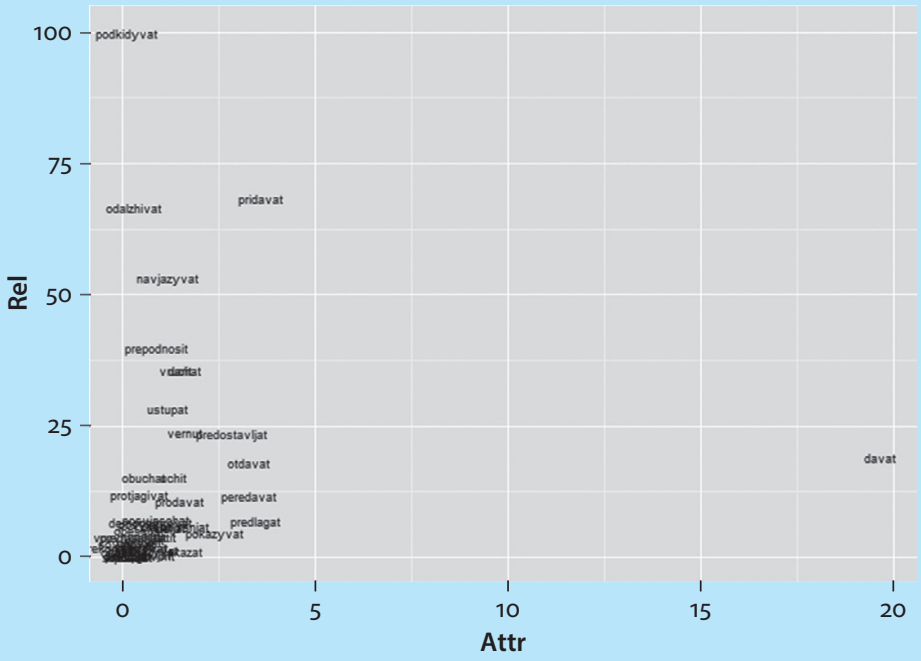


Figure 10.1a. A `ggplot2` version of scatter plot with text labels in Figure 10.1

In fact, if the Attraction and Reliance scores are sorted in decreasing order and plotted, the distributions closely resemble the Zipfian curve (see Chapter 3), as Figure 10.2 demonstrates. The plots can be created with the help of the following code:

```
> plot(sort(Attr, decreasing = TRUE), type = "l", main = "Attraction",
ylab = "Attraction, in %")
> plot(sort(Rel, decreasing = TRUE), type = "l", main = "Reliance",
ylab = "Reliance, in %")
```

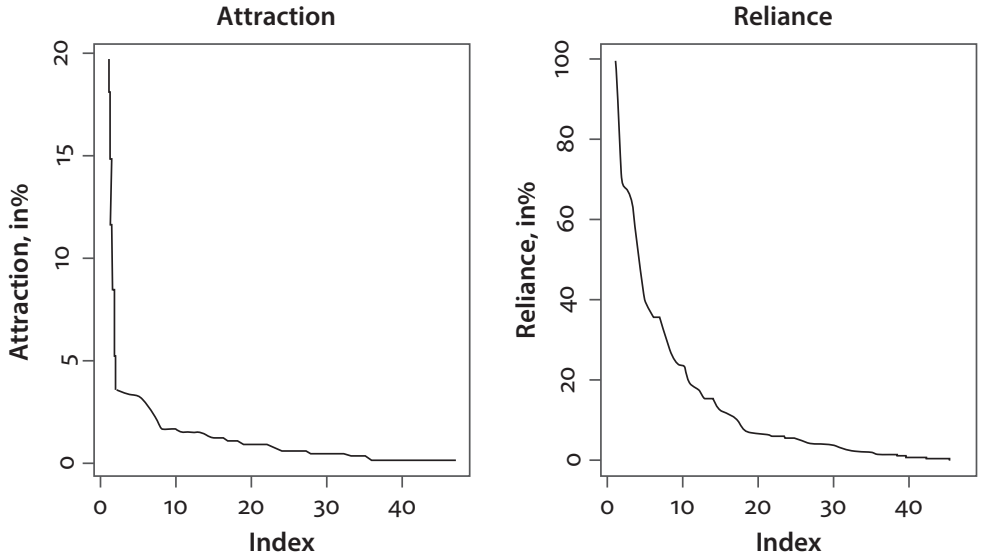


Figure 10.2. Distribution of Attraction and Reliance scores

This result corroborates previous findings about constructional collexemes in English (Ellis & Ferreira-Junior 2009).

The next measures are slightly more complex. Consider *ΔP*, recently used in constructionist studies by Ellis (2006) and Ellis & Ferreira-Junior (2009). It is a unidirectional contingency-based measure, which requires all four frequencies and comes in two 'flavours'. In one of them, the construction is the cue, and the collexeme is the response. The scores show the difference between the probability of the verb in the ditransitive construction and in all other contexts. The other version, conversely, treats the collexeme as the cue, and the construction as the response. It corresponds then to the difference between the probability of the ditransitive construction for a given verb, and that for all other verbs.

```
> dP.cueVerb <- a/(a + b) - c/(c + d)
> dP.cueCx <- a/(a + c) - b/(b + d)
```

If you explore the verbs with the top `dP.cueVerb` and `dP.cueCx` scores the way we did it above, you will see that the results are very similar to Reliance and Attraction, respectively.

So far, we have computed only unidirectional, or asymmetric measures. Let us now compute several popular bidirectional, or symmetric measures. All of them take into account contingency information. An example is the log-transformed Fisher exact test *p*-value (Stefanowitsch & Gries 2005). One could create for each set of frequencies a 2-by-2 table and use the Fisher exact test implemented in `fisher.test()` to obtain the *p*-values (see Chapter 9), but, to save time, you can use the function `pv.Fisher.collostr()` in the `Rling` package, which immediately returns a vector with *p*-values for all verbs:

```
> pvF <- pv.Fisher.collostr(a, b, c, d)
> round(head(pvF), 3)
[1] 0.539 0.000 0.000 0.000 0.852 0.729
```

To make the interpretation more intuitive, it is common in collostructionist studies to log-transform the *p*-values, so that they range from – infinity (mutual repulsion) to + infinity (mutual attraction), with the zero showing the lack of any association, either positive or negative. This is achieved by taking the negative logarithm of the *p*-value, and changing the sign to minus when the observed frequency is smaller than the expected one. Normally, the logarithm with the base 10 is used:

```
> logpvF <- ifelse(a < aExp, log10(pvF), -log10(pvF))
```

The function `ifelse()` takes three arguments: the condition ("if the observed frequency is smaller than the expected frequency"), the instruction what to do when the condition is met ("take a logarithm", which will result in a negative value because all numbers between 0 and 1 produce negative logarithms), and, finally, the instruction what to do when the condition is NOT met, i.e. the observed frequency is greater than expected ("take a negative

logarithm of the *p*-value", which will result in a positive number, since two minuses give a plus).

Now, compare the simple *p*-values and the log-transformed ones:

```
> round(head(pvF), 3)
[1] 0.539 0.000 0.000 0.000 0.852 0.729
> round(head(logpvF), 3)
[1] -0.268 14.892 151.008  3.954 -0.070 -0.137
```

One can see that large and insignificant *p*-values have become small values somewhere around zero. In contrast, very small and significant *p*-values have turned into large absolute numbers after the log-transformation. Note also that the sign has appeared. The negative log-values show that the verb occurs in the construction less frequently than one could expect. In other words, the collexeme is 'repelled'. The positive log-values, in contrast, indicate a very strong mutual attraction between a collexeme and the construction. The greatest score (151.008) belongs to *davat'* 'to give'.

Since minus log_{10} of 0.05 is approximately 1.3, this threshold can be used as a cut-off point to identify significantly attracted and repelled collexemes:

```
> -log10(0.05)
[1] 1.30103
```

Thus, the collexemes with the log-value greater than 1.3 are significantly attracted to the ditransitive construction, whereas the collexemes with the scores smaller than –1.3 are significantly repelled from it. However, this cut-off value should be taken with a grain of salt. The distinction between central and marginal collexemes is not clear-cut. Rather, it represents a continuum, and any cut-off point is arbitrary. One should also remember that *p*-values depend on the sample size. That is, a larger corpus will yield normally lower *p*-values and larger log-transformed scores than a smaller one.

Another popular measure is the log-likelihood ratio score (e.g. Dunning 1993), although the Fisher exact test is considered to be more powerful in case of low frequencies. To compute the log-likelihood scores, you can use another ready-made function from the `Rling` package:

```
> LL <- LL.collostr(a, b, c, d)
> LL1 <- ifelse(a < aExp, -LL, LL)
```

A list of these and other popular association measures and the R code is provided in Table 10.4.

How similar are these measures? This question can be answered with the help of correlation analysis (see Chapter 6). For contrastive purposes, one can also add a random score with replacement. That is, the algorithm picks up a random number from a pool, e.g. all integers from 0 to 100, *n* times (here, *n* is the total number of verbs, or the length of the frequency vector *a*). Imagine drawing lotto balls with numbers from a box *n* times.

Table 10.4 Association measures, based on Evert (2004) with some additions

Measure	Characteristic	R code
Attraction	unidirectional, non-contingency-based	`Attr <- 100*a/(a + c)`
Reliance	unidirectional, non-contingency-based	`Rel <- 100*a/(a + b)`
ΔP, construction as cue, collexeme as response	unidirectional, contingency-based	`dP.cueCx <- a/(a + c) - b/(b + d)`
ΔP, verb as cue, construction as response	unidirectional, contingency-based	`dP.cueVerb <- a/(a + b) - c/(c + d)`
Log-transformed Fisher's Exact Test *p*-value	bidirectional, contingency-based	`pvF <- pv.Fisher.collostr(a, b, c, d)` `logpvF <- ifelse(a < aExp, log10(pvF), -log10(pvF))`
Log-likelihood ratio	bidirectional, contingency-based	`LL <- LL.collostr(a, b, c, d)` `LL1 <- ifelse(a < aExp, -LL, LL)`
Pointwise Mutual Information	bidirectional, contingency-based	`PMI <- log(a/aExp)`[2]
MI2	bidirectional, contingency-based	`MI2 <- log(a^2/aExp)`
MI3	bidirectional, contingency-based	`MI3 <- log(a^3/aExp)`
local MI	bidirectional, contingency-based	`MIloc <- a*log(a/aExp)`
normalized PMI	bidirectional, contingency-based	`nPMI <- PMI/(-log(a/(a + b + c + d)))`
z-score	bidirectional, contingency-based	`z.score <- (a - aExp)/sqrt(aExp)`
t-score	bidirectional, contingency-based	`t.score <- (a - aExp)/sqrt(a)`
Pearson's χ^2-test statistic	bidirectional, contingency-based	`dExp <- (d + b)*(d + c)/(a + b + c + d)` `chisq <- (a + b + c + d)*(a - aExp)^2/(aExp*dExp)`

(*Continued*)

2. Different authors use different logarithmic bases, which can be specified in R as `log()` (natural logarithm to base $e \approx 2.718$), `log2()` (logarithm to base 2) or `log10()` (logarithm to base 10). Changing the base will change the magnitude of association scores, but the ranking of collexemes will remain the same.

Table 10.4 (Continued)

Measure	Characteristic	R code
Minimum sensitivity	bidirectional, contingency-based	`MS <- apply(cbind(Attr, Rel), 1, min)`
Jaccard coefficient	bidirectional, contingency-based	`Jaccard <- a/(a + b + c)`
Dice coefficient	bidirectional, contingency-based	`Dice <- 2*a/(2*a + b + c)`
Log odds ratio	bidirectional, contingency-based	`logOR <- log(a*d/(b*c))`
Discounted log odds ratio	bidirectional, contingency-based	`logOR.disc <-log((a + 0.5)*(d + 0.5)/ ((b + 0.5)*(c + 0.5)))`
Geometric mean	bidirectional, contingency-based	`gmean <- a/sqrt((a + b)*(a + c))`
Liddell's difference of proportions	unidirectional, contingency-based	`Liddell <- (a*d - b*c)/((a + c)* (b + d))`

The phrase 'with replacement' means that each number can be drawn from the total pool any number of times, as if you put a ball that you have drawn back into the box. Since the procedure is random, your scores will diverge from the ones below:

```
> random <- sample(0:100, length(a), replace = TRUE)
> head(random)
[1] 77 7 80 100      1 59
```

To obtain the correlations between all pairs of measures simultaneously, all association scores will be combined as columns in a matrix:

```
> assoc <- cbind(Attr, Rel, dP.cueCx, dP.cueVerb, logpvF, LL1, random)
```

Next, the function `cor()` can be used to compute all pairwise correlation coefficients. The Pearson product-moment correlation coefficient is computed as the default option. For convenience, the coefficients are rounded up to three decimal points in order to get a more compact representation:

```
> assoc.cor <- cor(assoc)
> round(assoc.cor, 3)
            Attr  Rel    dP.cueCx dP.cueVerb logpvF  LL1    random
Attr        1.000 0.111  0.997    0.112      0.983   0.983  0.054
Rel         0.111 1.000  0.140    1.000      0.220   0.220  -0.253
dP.cueCx    0.997 0.140  1.000    0.141      0.988   0.988  0.070
dP.cueVerb 0.112 1.000  0.141    1.000      0.221   0.221  -0.253
logpvF      0.983 0.220  0.988    0.221      1.000   1.000  0.041
LL1         0.983 0.220  0.988    0.221      1.000   1.000  0.040
random      0.054 -0.253 0.070    -0.253     0.041   0.040  1.000
```

The correlation matrix tells us that there are two very strongly correlated groups of scores. The first one includes Attraction, ΔP with the construction as the cue, the log-transformed Fisher exact test p-value and the log-likelihood measure. The second group is formed by Reliance and ΔP with the verb as the cue. The random variable, as one might have expected, is not strongly correlated with any parameters, although there is a weak negative correlation with Reliance and ΔP with the verb as the cue.

For those who prefer visual ways of displaying information, an attractive option might be a correlogram, which was introduced in Chapter 6. Figure 10.3 displays one of the graphical options. The strength of correlation is represented by the intensity of shading and also by the size of the coloured segments of the pie charts. The direction is represented by colours and the orientation of the coloured segments in the pie charts. The argument `order = TRUE` orders the variables in such a way that one can see the groups of strongly correlated variables.

```
> corrgram(assoc, order = TRUE, lower.panel = panel.shade, upper.
panel = panel.pie)
```

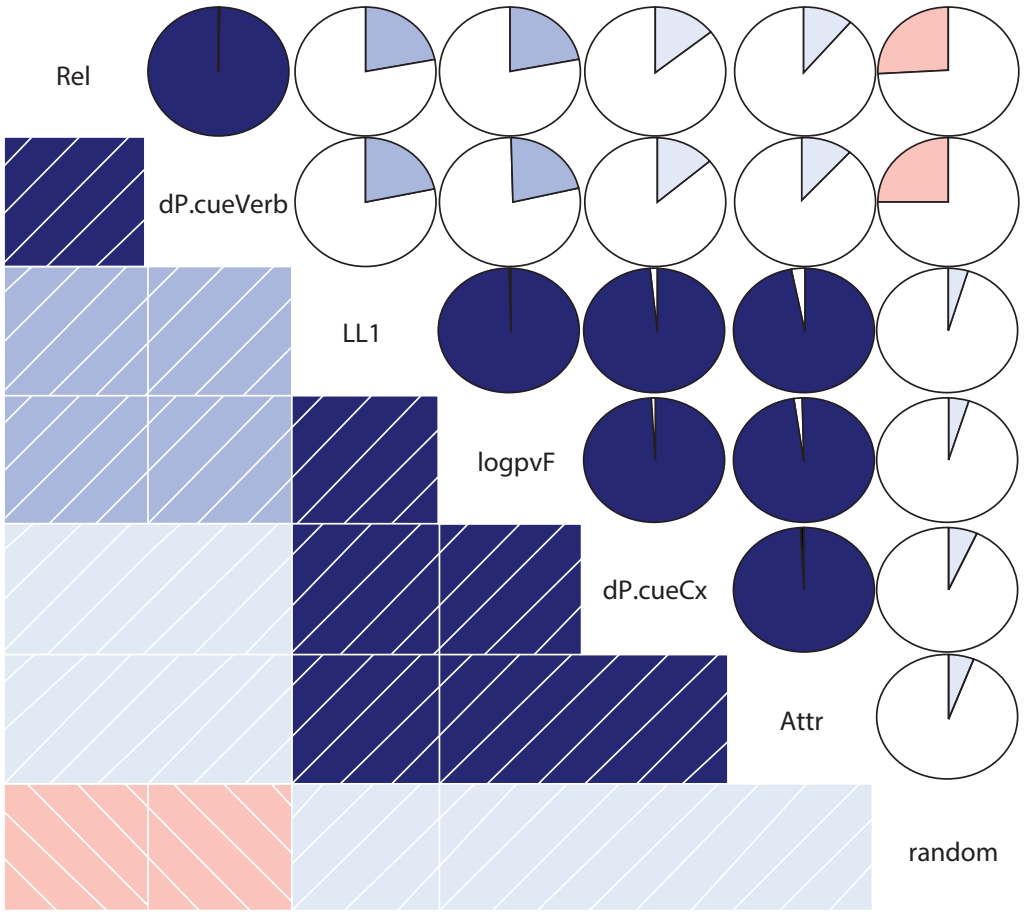


Figure 10.3. A correlogram of association scores with shaded squares and pie charts

For the purposes of more precise diagnostics of the relationships, it is also possible to visualize the observations as points plotted against the values of each variable compared, and ellipses that show the direction and strength of association. The rounder the ellipse, the weaker the correlation. One can also see smoothed curves, which show the direction of the relationship. The result is shown in Figure 10.4.

```
> corrgram(assoc, order = TRUE, lower.panel = panel.ellipse, upper.
panel = panel.pts)
```

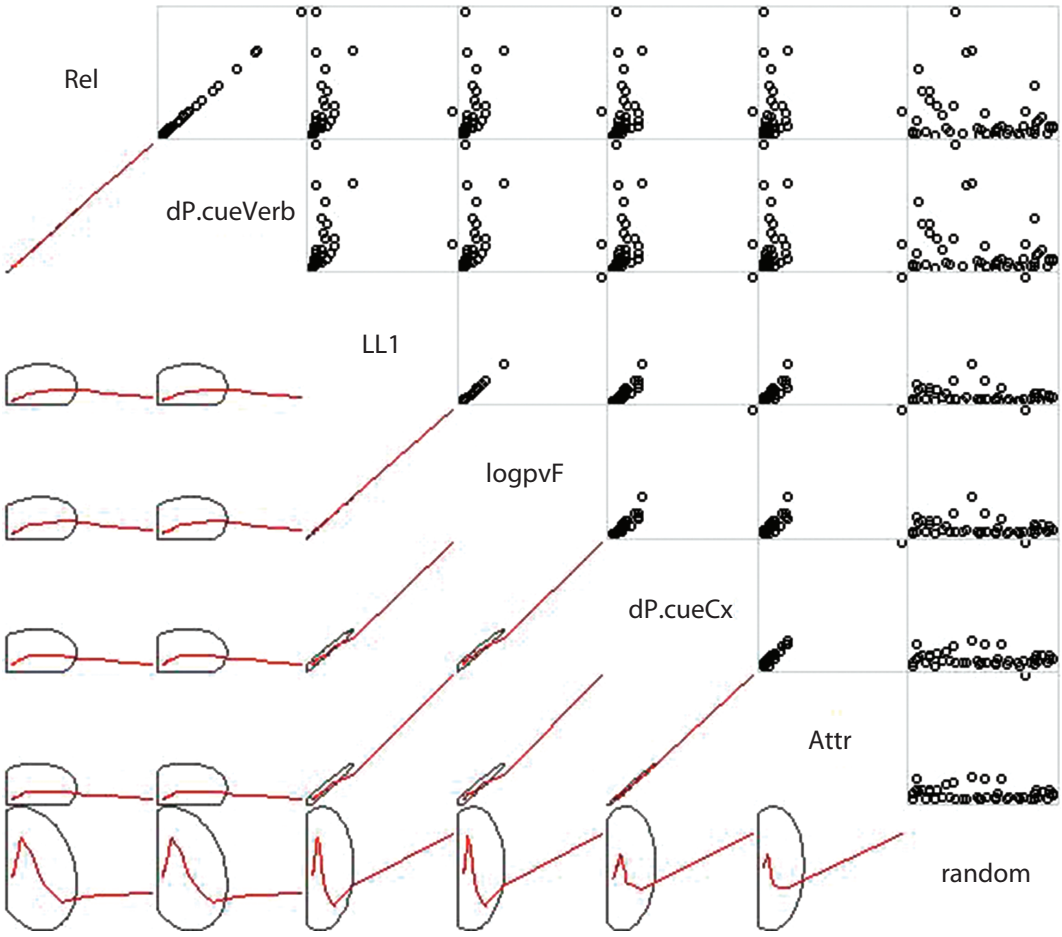


Figure 10.4. Correlogram of association measures: points (individual verbs) and ellipses with smoothed lines

Consider the nearly perfectly linear relationships between *Rel* and *dP.cueVerb*, or between *Attr*, *dP.cueCx*, *logpvF* and *LL1*. One can also detect the presence of outliers with the help of the scatter plots in the upper panel. Notably, all plots with *Attr*, *dP.cueCx*, *logpvF* and *LL1*, show the presence of a powerful outlier. A quick examination of the initial scores reveals that the outlier is again *davat'* 'to give'. In such situations, it is more correct to use a non-parametric correlation test, such as Kendall's τ. The latter gives somewhat different results:

```
> assoc.cor1 <- cor(assoc, method = "kendall")
> round(assoc.cor1, 3)
            Attr   Rel    dP.cueCx  dP.cueVerb  logpvF LL1    random
Attr        1.000  0.442  0.864     0.442       0.738  0.724  0.076
Rel         0.442  1.000  0.591     1.000       0.716  0.731  0.045
dP.cueCx    0.864  0.591  1.000     0.591       0.870  0.859  0.007
dP.cueVerb  0.442  1.000  0.591     1.000       0.716  0.731  0.045
logpvF      0.738  0.716  0.870     0.716       1.000  0.985  0.014
LL1         0.724  0.731  0.859     0.731       0.985  1.000  0.019
random      0.076  0.045  0.007     0.045       0.014  0.019  1.000
```

According to the ranks method, Attraction is less strongly correlated with ΔP with the construction as the cue, the log-transformed Fisher exact test p-value and the log-likelihood measure than when the Pearson correlation coefficients were used. On the other hand, Attraction and Reliance are now correlated more strongly than before. These differences in the results demonstrate the importance of diagnostics and visualization of data, and show how the choice of a statistic may change the picture.

10.3 Summary

This chapter has introduced a few popular measures of collocational and collostructional strength. The reader may wonder which ones he or she should use. There is no easy and definite answer to this question. It is still unclear which measures represent the information that the speakers store in their minds more adequately than the others, since recent empirical studies based on corpus-based and experimental evidence have yielded divergent results. Yet, it is possible to give some general recommendations. For low-frequency data, such measures as t-scores and z-scores are less reliable than the log-likelihood values and especially the log-transformed Fisher exact p-values. Log-odd ratios are not sensitive to frequency information, unlike the hypothesis-testing statistics based on p-values or log-likelihood ratios. As a consequence, it would make more sense to use odds ratios if you want to compare results based on datasets of different sizes. The PMI measure gives more weight to co-occurrences of low-frequency elements, and has been widely used in computational studies with very many target words and contextual features (see Chapter 16). In addition, as the correlogram plots have shown, most measures are highly correlated, so another recommendation is to be cautious when reporting the 'best' measures. The next chapter will demonstrate how collostructional strength can be applied in quantitative models of semantics and language variation.

How to report association measures

There are no general rules for reporting association measures. The format depends on the specific task. For example, collostructional studies often present the results in tables with ranked collexemes, their observed and expected frequencies, as well as the association measure scores.

More on association measures

To find out more on different association measures, one can refer to Evert (2004). For applications in collostructional analysis, see numerous publications by Gries, Stefanowitsch and their colleagues, beginning from Stefanowitsch & Gries (2003). A comparison of different measures of association between collexemes and constructions can be found in Wiechmann (2008). See also Barnbrook et al. (2013) for a broader background and applications of collocational analysis.