The Stata Journal (2024)
24, Number 2, pp. 287–300 DOI: 10.1177/1536867X241257804

Estimating Skellam distribution and regression parameters in Stata

Vincenzo Verardi
CRED, DEFIPP, FNRS
Université de Namur
Namur, Belgium
vincenzo.verardi@unamur.be

Catherine Vermandele
LMTD
Université libre de Bruxelles
Brussels, Belgium
catherine.vermandele@ulb.be

Abstract. The Skellam distribution is a discrete probability distribution related to the difference between two independent Poisson-distributed random variables. It has been used in a variety of contexts, including sports or supply and demand imbalances in shared transportation. Stata does not support the Skellam distribution or Skellam regression. We present a command, `skellamreg`, to estimate the parameters of a Skellam distribution and Skellam regression model using Mata's optimize function.

Keywords: st0748, skellamreg, skellamreg postestimation, Skellam distribution, Skellam regression, modified Bessel function of the first kind, maximum likelihood, optimize

1 Introduction

Skellam (1946), generalizing the work of Irwin (1937), investigated the properties of a discrete random variable Y, which is the difference between two independent Poisson-distributed random variables Y_1 and Y_2.

The Skellam distribution is often used to model the number of points that separate two teams in sports such as hockey and soccer. Kendall (1951) and Dobbie (1961) show that it can also be used in the problem of taxis and customers coming to a waiting area in different Poisson flows (that is, with different rates). The number of taxis waiting is the (integer) variable of interest. This number can be positive if taxis are waiting, zero in the absence of both taxis and customers waiting, or negative if customers are waiting. More recently, Liu and Pelechrinis (2021b) look at the case of shared transportation. They use a Skellam regression to predict the difference in overall demand and supply at a particular bike station over a certain period.

The Skellam distribution and the Skellam regression are not implemented in Stata, although they are in R and Python (see Lewis, Brown, and Tsagris [2017] and Liu and Pelechrinis [2021a]). This short article presents a command, `skellamreg`, that may be used to fit the parameters of a Skellam distribution (and subsequently those of a Skellam regression model).

Let Y_1 and Y_2 be two independent Poisson-distributed random variables with means μ_1 and μ_2. Then $Y = Y_1 - Y_2$ has a Skellam distribution. Its probability mass function is given by

$$\Pr(Y = k) = e^{-(\mu_1+\mu_2)} \left(\frac{\mu_1}{\mu_2}\right)^{k/2} I_k\left(2\sqrt{\mu_1\mu_2}\right)$$

where $k \in \mathbb{Z}$ and $I_k(\cdot)$ is the modified Bessel function of the first kind: for $x \in \mathbb{R}^+$,

$$I_k(x) = \sum_{m=0}^{\infty} \frac{1}{m!\,\Gamma(m+k+1)} \left(\frac{x}{2}\right)^{2m+k}$$

where $\Gamma(\cdot)$ is the Gamma function [$\Gamma(z) = \int_0^\infty t^{z-1}e^{-t}dt$]. Because the possible values k of the Skellam distribution are integer, $I_{-k}(x) = I_k(x)$ (see Abramowitz and Stegun [1972, 375, eq. 9.6.6]). Function $I_k(\cdot)$ can thus be replaced by $I_{|k|}(\cdot)$ in the above formula.

To guarantee the positiveness of μ_1 and μ_2 when estimating the parameters, we can reparameterize the probability mass function by defining $\mu_1 = \exp(\lambda_1)$ and $\mu_2 = \exp(\lambda_2)$.

Generally, Y_1 and Y_2 are unavailable and only $Y = Y_1 - Y_2$ is observed.

Stata does not offer Bessel functions; however, C++ does, and these functions may be written almost exactly in the same way in Mata. In the command described in this article, we use the programs provided by Jean-Pierre Moreau, with his permission. Moreau acknowledges the use of the Numath Library in Fortran 77, developed by Tuan Dang Trong, as the primary reference. Please refer to Moreau (2011) for the computation of the modified first-kind Bessel function of integer order k for any real value x. It is then simple to create a maximum likelihood program once Mata has access to the modified Bessel function of the first kind. In the `skellamreg` postestimation `predict` command, some functionalities are provided (predicted difference, as well as linear prediction and predicted counts for both underlying processes). Using Mata's optimization techniques, we maximize the log-likelihood by taking advantage of analytical formulations for the first and second derivatives. Both numerical accuracy and speed are greatly improved by this, particularly when contrasted with numerical derivatives.

2 Maximum likelihood estimator

The likelihood function is given by

$$\begin{aligned}\mathcal{L}(\lambda_1, \lambda_2; k_1, \ldots, k_n) &= \prod_{i=1}^{n} \Pr(Y_i = k_i|\lambda_1, \lambda_2) \\ &= \prod_{i=1}^{n} \left\{ e^{-\left(e^{\lambda_1}+e^{\lambda_2}\right)} \left(e^{\lambda_1-\lambda_2}\right)^{k_i/2} I_{|k_i|}\left(2\sqrt{e^{\lambda_1+\lambda_2}}\right)\right\}\end{aligned}$$

The maximum likelihood estimates $\widehat{\lambda}_1$ and $\widehat{\lambda}_2$ of the two parameters of the Skellam distribution are solutions of the maximization problem

$$\max_{\lambda_1,\lambda_2\in\mathbb{R}} \ln \mathcal{L}(\lambda_1,\lambda_2;k_1,\ldots,k_n) = \max_{\lambda_1,\lambda_2\in\mathbb{R}} \sum_{i=1}^{n} L(\lambda_1,\lambda_2;k_i)$$

where

$$L(\lambda_1,\lambda_2;k) = -(e^{\lambda_1}+e^{\lambda_2}) + (\lambda_1-\lambda_2)\frac{k}{2} + \ln I_{|k|}\left(2\sqrt{e^{\lambda_1+\lambda_2}}\right), \qquad k\in\mathbb{Z}$$

To solve this maximization problem, we can easily compute the gradient and the Hessian, with respect to λ_1 and λ_2, of the log-likelihood function, and hence of function $L(\lambda_1,\lambda_2;k)$. Because, for $k\in\mathbb{Z}$,

$$I_k'(z) = \frac{d}{dz}I_k(z) = \frac{I_{k-1}(z)+I_{k+1}(z)}{2}$$

(see Abramowitz and Stegun [1972, 376, eq. 9.6.26]), we have the following first derivatives for the gradient:

$$\frac{\partial}{\partial\lambda_1}L(\lambda_1,\lambda_2;k) = -e^{\lambda_1} + \frac{k}{2} + \frac{\sqrt{e^{\lambda_1+\lambda_2}}}{2}\left(\frac{I_{||k|-1|}\left(2\sqrt{e^{\lambda_1+\lambda_2}}\right)+I_{|k|+1}\left(2\sqrt{e^{\lambda_1+\lambda_2}}\right)}{I_{|k|}\left(2\sqrt{e^{\lambda_1+\lambda_2}}\right)}\right)$$

$$\frac{\partial}{\partial\lambda_2}L(\lambda_1,\lambda_2;k) = -e^{\lambda_2} - \frac{k}{2} + \frac{\sqrt{e^{\lambda_1+\lambda_2}}}{2}\left(\frac{I_{||k|-1|}\left(2\sqrt{e^{\lambda_1+\lambda_2}}\right)+I_{|k|+1}\left(2\sqrt{e^{\lambda_1+\lambda_2}}\right)}{I_{|k|}\left(2\sqrt{e^{\lambda_1+\lambda_2}}\right)}\right)$$

For the Hessian, let's first calculate the cross derivatives:

$$\begin{aligned}\frac{\partial^2}{\partial\lambda_1\partial\lambda_2}L(\lambda_1,\lambda_2;k) &= \frac{\partial^2}{\partial\lambda_2\partial\lambda_1}L(\lambda_1,\lambda_2;k)\\ &= \frac{e^{\lambda_1+\lambda_2}}{2} + \frac{e^{\lambda_1+\lambda_2}}{4}\left(\frac{I_{||k|-2|}\left(2\sqrt{e^{\lambda_1+\lambda_2}}\right)+I_{|k|+2}\left(2\sqrt{e^{\lambda_1+\lambda_2}}\right)}{I_{|k|}\left(2\sqrt{e^{\lambda_1+\lambda_2}}\right)}\right)\\ &\quad + \frac{\sqrt{e^{\lambda_1+\lambda_2}}}{4}\left(\frac{I_{||k|-1|}\left(2\sqrt{e^{\lambda_1+\lambda_2}}\right)+I_{|k|+1}\left(2\sqrt{e^{\lambda_1+\lambda_2}}\right)}{I_{|k|}\left(2\sqrt{e^{\lambda_1+\lambda_2}}\right)}\right)\\ &\quad \times\left\{1-\sqrt{e^{\lambda_1+\lambda_2}}\left(\frac{I_{||k|-1|}\left(2\sqrt{e^{\lambda_1+\lambda_2}}\right)+I_{|k|+1}\left(2\sqrt{e^{\lambda_1+\lambda_2}}\right)}{I_{|k|}\left(2\sqrt{e^{\lambda_1+\lambda_2}}\right)}\right)\right\}\end{aligned}$$

The second derivatives are given by

$$\frac{\partial^2}{\partial\lambda_1^2}L(\lambda_1,\lambda_2;k) = -e^{\lambda_1} + \frac{\partial^2}{\partial\lambda_1\partial\lambda_2}L(\lambda_1,\lambda_2;k)$$
$$\frac{\partial^2}{\partial\lambda_2^2}L(\lambda_1,\lambda_2;k) = -e^{\lambda_2} + \frac{\partial^2}{\partial\lambda_1\partial\lambda_2}L(\lambda_1,\lambda_2;k)$$

In the context of Skellam regression, the parameters λ_1 and λ_2 of the two independent Poisson distributions are expressed as linear functions of covariates. That is to say that, for $i = 1, \ldots, n$,

$$\Pr(Y_i = k_i) = e^{-\left(e^{\lambda_{1i}}+e^{\lambda_{2i}}\right)}\left(e^{\lambda_{1i}-\lambda_{2i}}\right)^{k_i/2} I_{|k_i|}\left(2\sqrt{e^{\lambda_{1i}+\lambda_{2i}}}\right)$$

where $\lambda_{1i} = \mathbf{x}_i^T\boldsymbol{\beta}$ and $\lambda_{2i} = \mathbf{v}_i^T\boldsymbol{\gamma}$, with $\mathbf{x}_i = (1, x_{i1}, \ldots, x_{ip})^T$ and $\mathbf{v}_i = (1, v_{i1}, \ldots, v_{iq})^T$. We have here to estimate two vectors of parameters—the $(p+1)$-vector $\boldsymbol{\beta}$ and the $(q+1)$-vector $\boldsymbol{\gamma}$—by solving the maximization problem

$$\max_{\boldsymbol{\beta}\in\mathbb{R}^{p+1},\boldsymbol{\gamma}\in\mathbb{R}^{q+1}} \sum_{i=1}^{n} L(\boldsymbol{\beta},\boldsymbol{\gamma};k_i,\mathbf{x}_i,\mathbf{v}_i)$$

where, for $i = 1, \ldots, n$,

$$L(\boldsymbol{\beta},\boldsymbol{\gamma};k_i,\mathbf{x}_i,\mathbf{v}_i) = -\left(e^{\mathbf{x}_i^T\boldsymbol{\beta}} + e^{\mathbf{v}_i^T\boldsymbol{\gamma}}\right) + \left(\mathbf{x}_i^T\boldsymbol{\beta} - \mathbf{v}_i^T\boldsymbol{\gamma}\right)\frac{k_i}{2} + \ln I_{|k_i|}\left(2\sqrt{e^{\mathbf{x}_i^T\boldsymbol{\beta}+\mathbf{v}_i^T\boldsymbol{\gamma}}}\right)$$

The first and second derivatives presented above have to be modified and multiplied, respectively, by $\mathbf{x}_i^T$ or $\mathbf{v}_i^T$ for the gradient and by $\mathbf{x}_i\mathbf{x}_i^T$, $\mathbf{v}_i\mathbf{v}_i^T$ or $\mathbf{x}_i\mathbf{v}_i^T$ for the second and cross derivatives. Naturally, λ_1 and λ_2 (which have to be indexed by i in this situation) have to be replaced by their expressions ($\lambda_{1i} = \mathbf{x}_i^T\boldsymbol{\beta}$ and $\lambda_{2i} = \mathbf{v}_i^T\boldsymbol{\gamma}$) in all formulas above in the maximum likelihood setup.

3 Skellam regression in the case of nonindependent Poisson distributions

Karlis and Ntzoufras (2003) have shown that the Skellam distribution and regression can be used in situations involving nonindependent Poisson random variables.

They consider random variables X_j, $j = 1, 2, 3$, which follow independent Poisson distributions with parameters $\mu_j > 0$. They then show, relying on the existing literature, that random variables $Y_1 = X_1 + X_3$ and $Y_2 = X_2 + X_3$ follow jointly a bivariate Poisson distribution $\text{BP}(\mu_1, \mu_2, \mu_3)$, with joint probability function

$$\Pr(Y_1 = k_1, Y_2 = k_2) = e^{-(\mu_1+\mu_2+\mu_3)}\frac{\mu_1^{k_1}}{k_1!}\frac{\mu_2^{k_2}}{k_2!}\sum_{k=0}^{\min(k_1,k_2)}\binom{k_1}{k}\binom{k_2}{k}k!\left(\frac{\mu_3}{\mu_1\mu_2}\right)^k$$

The authors explain that marginally each random variable follows a Poisson distribution with mean $E(Y_1) = \mu_1 + \mu_3$ and $E(Y_2) = \mu_2 + \mu_3$ and that the covariance between Y_1 and Y_2 is $\text{Cov}(Y_1, Y_2) = \mu_3$. As discussed by Karlis and Ntzoufras (2003), this BP distribution can be considered, with some caveats, for modeling dependence in team sports. In the authors' own words, a natural interpretation of the parameters of this BP model is that μ_1 and μ_2 reflect the "net" scoring ability of each team, whereas μ_3 reflects common game conditions (for example, the speed of the game, the weather conditions, and stadium environment).

Karlis and Ntzoufras (2003) then show that if, in this setup, we consider the difference $Y = Y_1 - Y_2$ of the goals scored by two opposing teams, the probability function of Y is independent of μ_3 and is the same as that derived from two independent Poisson variables—that is, Y follows a Skellam distribution with parameters μ_1 and μ_2. So if we are interested in a regression model for the difference Y of the scores of two opposing teams, we may use Skellam regression. Note, however, that while the results of this Skellam regression do allow to predict $E(Y|\mathbf{x})$, they do not give us an estimate of the parameter μ_3 and hence do not allow to correctly predict the marginal mean scores $E(Y_1|\mathbf{x})$ and $E(Y_2|\mathbf{x})$.

4 The skellamreg command

4.1 Syntax

The syntax of the `skellamreg` command is as follows. A first alternative for this syntax is

`skellamreg` *depvar* [*indepvars*] [*if*] [*in*] [`, robust cluster(`*varname*`) nolog noconstant stub(`*string*`) technique(`*string*`) nodofcorrection level(`*cilevel*`)`]

Here all the explanatory variables are assumed to be the same for the two underlying Poisson equations. If no explanatory variables are declared, only a constant is considered among regressors (which brings us to the unconditional estimation of rate parameters).

A second alternative for the syntax is

`skellamreg` *depvar* `(`*indepvars*$_1$`)` `(`*indepvars*$_2$`)` [*if*] [*in*] [`, robust cluster(`*varname*`) nolog noconstant stub(`*string*`) technique(`*string*`) nodofcorrection level(`*cilevel*`)`]

Here the explanatory variables are split into two groups (the first one is related to the first Poisson equation, and the second one to the second Poisson equation) that are not constrained to be the same. If parentheses are left empty, only a constant is considered for the associated underlying Poisson, and the unconditional rate is considered.

By default, the code will automatically generate two temporary variables, naming them as the dependent variable and adding `_count_1` and `_count_2` to it. If these variables are already in use, an error will be raised. When the name is already present in the dataset, for instance, users have the opportunity to select a different name using the `stub()` option.

4.2 Options

`robust` specifies to use the sandwich variance formula to compute standard errors of the estimated parameters.

`cluster(`*varname*`)` specifies to compute cluster-corrected standard errors of the estimated parameters.

`nolog` specifies to hide iteration logs.

`noconstant` specifies to fit a model without constants.

`stub(`*string*`)` specifies to provide a stub for the dependent variable.

`technique(`*string*`)` specifies to change the optimization technique.

`nodofcorrection` specifies to avoid the correction for the degrees of freedom.

`level(`*cilevel*`)` specifies to set the confidence level.

4.3 Predictions

Three groups of predicted values are available with `predict` postestimation:

`ndiff` generates predicted differences in counts between the two Poisson processes (the default).

`xb1` generates linear predictions for the first process.

`xb2` generates linear predictions for the second process.

`n1` generates predicted counts [that is, exp(`xb1`)] for the first process.

`n2` generates predicted counts [that is, exp(`xb2`)] for the second process.

More precisely, option `n1` takes the exponential of the first Poisson process's linear prediction and `n2` of the second. Option `ndiff` takes the difference of the exponential of the linear predictions.

4.4 Stored results

`skellamreg` stores the following in `e()`:

Scalars	
`e(N)`	number of observations
`e(ll)`	log likelihood
Macros	
`e(cmd)`	`skellamreg`
`e(cmdline)`	command as typed
`e(depvar)`	name of dependent variable
`e(title)`	title in estimation output
`e(properties)`	`b V`
`e(predict)`	program used to implement `predict`
Matrices	
`e(b)`	coefficient vector
`e(V)`	variance–covariance matrix of the estimators
Functions	
`e(sample)`	marks estimation sample

5 Synthetic data example

To illustrate how to use the command in practice, we generate $n = 250$ bivariate observations (x_i, y_i) from the following process: $X \sim \mathcal{N}(2,1), Y = Y_1 - Y_2$ with Y_1 and Y_2 independent, $Y_1 \sim \mathcal{P}\{\mu_1(X) = \exp(0 + 0.6X)\}$ and $Y_2 \sim \mathcal{P}\{\mu_2(X) = \exp(0 + 0.4X)\}$ setting the seed to 1234 for replicability. We then regress Y on X using the command

```
skellamreg y x, stub(a)
```

and test whether the coefficients associated with X in both equations are the same and equal to zero, using the command

```
test [a1]:x=[a2]:x=0
```

The results are as follows:

```
. skellamreg y x*, stub(a) nolog
                                                Number of obs =     250
Log likelihood = -580.9047
------------------------------------------------------------------------------
           y | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
a1           |
           x |   .6389506   .0755365     8.46   0.000     .4909018    .7869993
       _cons |   .0055461   .1848015     0.03   0.976    -.3566581    .3677502
-------------+----------------------------------------------------------------
a2           |
           x |   .5019903   .1108581     4.53   0.000     .2847124    .7192681
       _cons |  -.0193426   .2401862    -0.08   0.936     -.490099    .4514137
------------------------------------------------------------------------------
```

```
. test [a1]:x=[a2]:x=0
 ( 1)  [a1]x - [a2]x = 0
 ( 2)  [a1]x = 0
           chi2(  2) =   92.79
         Prob > chi2 =    0.0000
```

Figure 1 presents, in the upper graph, the scatterplot of the dataset $\{(x_i, y_i)\,; i = 1, \ldots, 250\}$ and the predicted count difference line, which gives, for $x \in [\min\{x_i\}, \max\{x_i\}]$, the value of $\widehat{E}(Y \mid X = x) = \widehat{\mu}_1(x) - \widehat{\mu}_2(x) = \exp(\widehat{\beta}_0 + \widehat{\beta}_1 x) - \exp(\widehat{\gamma}_0 + \widehat{\gamma}_1 x)$. The lower graph allows to distinguish the estimation results relative to the two underlying independent Poisson processes. It presents the following:

- squares and long-dashed line: the scatterplot of the data (x_i, y_{i1}), $i = 1, \ldots, 250$, and the predicted counts $\widehat{E}(Y_1 \mid X = x) = \widehat{\mu}_1(x) = \exp(\widehat{\beta}_0 + \widehat{\beta}_1 x)$;
- triangles and short-dashed line: the scatterplot of the data (x_i, y_{i2}), $i = 1, \ldots, 250$, and the predicted counts $\widehat{E}(Y_2 \mid X = x) = \widehat{\mu}_2(x) = \exp(\widehat{\gamma}_0 + \widehat{\gamma}_1 x)$.

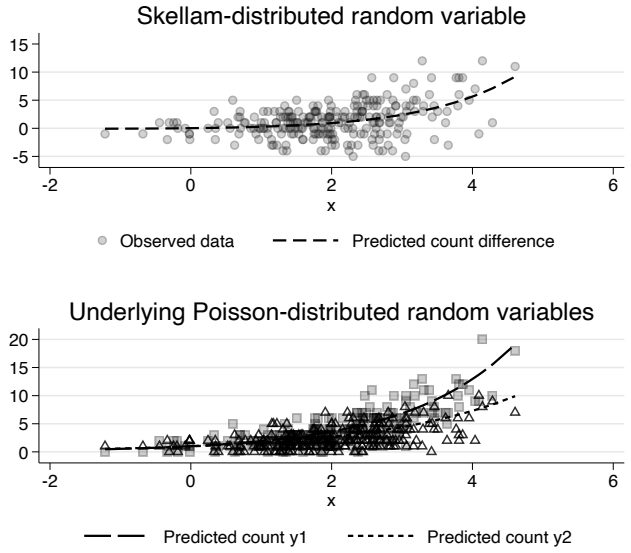

Figure 1. Predicted count difference (upper graph) and predicted counts for the underlying independent Poisson-distributed random variables (lower graph), resulting from the Skellam regression

The test clearly rejects that the coefficients associated with X in both equations are equal and equal to zero.

One might also be interested in understanding how variable Y changes after an infinitesimal variation of X. This can easily be done using the `margins` command:

```
margins, dydx(x) at(x=(-2(0.1)5))
marginsplot, recastci(rarea)              ///
   plotopts(lcolor(black) msymbol(none)) ///
   ciopts (color (black%20))
```

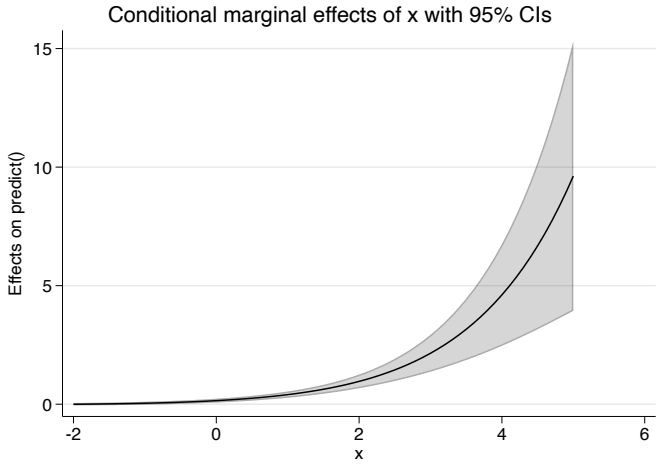

Figure 2. Marginal effect on $E(Y)$ of an infinitesimal variation of X, in function of the value of X

Figure 2 shows that the effect of an infinitesimal variation of X on the expectation of the dependent variable Y depends, in an exponential way, on the value x of X.

6 Real-data example

This specific case study examines how the day of the week on which a match is played may affect the dynamics of goal scoring in association football (soccer). We use information from the English Football Premier League from the season 2007–2008 to the season 2021–2022. All data come from https://www.football-data.co.uk/englandm.php. The dependent variable (`dftg`) that we create corresponds, for each match, to the difference between the number of goals scored by the home team (`fthg`) and the number of goals scored by the visiting team (`ftag`). Bet365 odds are incorporated in the regression model and serve as an indirect measure for accounting for the comparative strength of teams in the game. Bet365 is an online company, offering services in sports betting. More precisely, `b365h` is Bet365 home win odds, `b365d` is Bet365 draw odds, and `b365a` is Bet365 away win odds. In addition, year dummies are included to account for potential time-specific effects.

According to Karlis and Ntzoufras (2003), previous studies have found a small correlation between the number of goals scored by both teams involved in a match. However, as seen in section 3, the use of the Skellam regression model aims to eliminate the additive component (the covariance) that is common to both Poisson processes. In this context, the great advantage of the Skellam regression procedure is that, without having to model the covariance between the two Poisson processes, it provides estimates of the $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ regression coefficients and correctly predicts the conditional expectation of the goal difference. However, keep in mind that it does not correctly predict the conditional expectations of the number of goals scored by each of the teams.

We start the analysis by importing the data and generating year variable `t`:

```
tempfile dataset
quietly import delimited                              ///
   "https://www.football-data.co.uk/mmz4281/2122/E0.csv", clear
generate t = 2022
save `dataset', replace
local t 2022
foreach v in 2021 1920 1819 1718 1617 1516 1415 ///
   1314 1213 1112 1011 0910  0809 0708 {
       local t = `t'-1
       local url="https://www.football-data.co.uk/mmz4281/`v'/E0.csv"
       quietly import delimited "`url'", clear
       append using `dataset'
       replace t = `t'  if t == .
       save `dataset', replace
}
```

We then generate the difference in goals scored by the home and visiting team,

```
generate dftg = fthg-ftag
```

and create a categorical variable with the days of the week:

```
generate date2 = date(date, "DM20Y")
format date2 %td
drop date
rename date2 date
generate day = dow(date)
label def day 0 "Sunday" 1 "Monday" 2 "Tuesday" 3 "Wednesday" ///
   4 "Thursday " 5 "Friday" 6 "Saturday"
label value day day
```

We then run the estimation command and store the results:

```
skellamreg dftg b365h b365d b365a i.day i.t, stub(ftg_)
estimates store Goals
```

The `e(ll)` stored result makes it easier to do a number of tasks after the estimation. These include likelihood-ratio tests and pseudo-R^2 calculations, which are important for testing hypotheses and evaluating models in statistical analysis. For example, using Jann's (2005) `estadd` command for the latter, one could do the following:

```
local ll = e(ll)
quietly skellamreg dftg
local r2 = 1-`ll'/e(ll)
estimates restore Goals
estadd scalar r2_p `r2'
```

The `esttab` command by Jann (2007) is used to prepare a table presenting the estimated regression coefficients:

```
esttab Goals using skellam.tex, noabbrev tex unstack                    ///
   keep(b36* *day) nodep nonumb nobase star(* 0.10  ** 0.05 *** 0.01)   ///
   noobs label note(Time dummies coefficients not reported) replace     ///
   addnotes($t$ statistics in parentheses) ///
   stats(N r2_p ll, labels("Observations" "Pseudo-R²" "Log likelihood"))  ///
   eqlabels("Home" "Away")
```

	Goals	
	Home	Away
B365H	-0.198***	0.0219
	(-9.47)	(1.01)
B365D	0.0745**	0.0600
	(2.28)	(0.98)
B365A	-0.000777	-0.125***
	(-0.07)	(-5.55)
Monday	-0.0896	-0.143
	(-1.17)	(-1.44)
Tuesday	-0.0749	-0.0968
	(-0.96)	(-0.99)
Wednesday	-0.132**	-0.0925
	(-2.05)	(-1.14)
Thursday	-0.298*	-0.0725
	(-1.93)	(-0.45)
Friday	-0.0899	0.141
	(-0.60)	(0.94)
Saturday	-0.0902**	-0.00237
	(-2.26)	(-0.05)
Observations	5700	
Pseudo-R^2	0.0689	
Log likelihood	-10757.5	

Time dummies coefficients not reported

t statistics in parentheses

* $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

It turns out that the number of goals scored on average by the home team is significantly smaller (working at a probability level of 5%) on Wednesdays and Saturdays with respect to Sundays. We failed to find any significant effect on the visiting team.

7 Conclusion

The main objective of this article is to provide a command for the estimation of the parameters of Skellam distributions (and consequently of Skellam regressions), typically used for modeling the difference between two uncorrelated random variables that follow a Poisson distribution. This is achieved through the use of optimization tools in Stata and Mata and C++ codes by Moreau (2011) that calculate the modified Bessel function of

the first kind of integer order. Nevertheless, our present focus has been restricted to the traditional Skellam distribution. This analysis could be expanded by including excess zeros or overdispersion (and underdispersion) in the underlying Poisson distributions, which might be interesting in specific empirical scenarios. They will be considered in future research.

Currently, weights are omitted from the analysis. This requires a more thorough investigation and will probably be considered in future research.

8 Acknowledgments

We thank Jean-Pierre Moreau, who gave us permission to use his C++ codes on Bessel functions almost verbatim in Stata and Mata.

We also thank the anonymous referee and the editor for their insightful comments. Their suggestions have allowed us to greatly improve the quality of the article.

Vincenzo Verardi gratefully acknowledges the financial support of the Fonds National de la Recherche Scientifique (FNRS).

9 Programs and supplemental material

To install the software files as they existed at the time of publication of this article, type

```
. net sj 24-2
. net install st0748     (to install program files, if available)
. net get st0748         (to install ancillary files, if available)
```

10 References

Abramowitz, M., and I. A. Stegun, eds. 1972. *Handbook of Mathematical Functions: With Formulas, Graphs, and Mathematical Tables*. New York: Dover.

Dobbie, J. M. 1961. Letter to the editor—A doubled-ended queuing problem of Kendall. *Operations Research* 9: 755–757. https://doi.org/10.1287/opre.9.5.755.

Irwin, J. O. 1937. The frequency distribution of the difference between two independent variates following the same Poisson distribution. *Journal of the Royal Statistical Society,* A ser., 100: 415–416. https://doi.org/10.2307/2980526.

Jann, B. 2005. Making regression tables from stored estimates. *Stata Journal* 5: 288–308. https://doi.org/10.1177/1536867X0500500302.

———. 2007. Making regression tables simplified. *Stata Journal* 7: 227–244. https://doi.org/10.1177/1536867X0700700207.

Karlis, D., and I. Ntzoufras. 2003. Analysis of sports data using bivariate Poisson models. *Journal of the Royal Statistical Society,* D ser., 52: 381–393. https://doi.org/10.1111/1467-9884.00366.

Kendall, D. G. 1951. Some problems in the theory of queues. *Journal of the Royal Statistical Society,* B ser., 13: 151–185. https://doi.org/10.1111/j.2517-6161.1951.tb00080.x.

Lewis, J. W., P. E. Brown, and M. Tsagris. 2017. skellam: Densities and sampling for the Skellam distribution version 0.2.1. R-Forge. https://rdrr.io/rforge/skellam/.

Liu, X., and K. Pelechrinis. 2021a. xinliupitt/skellam_regression. GitHub. https://github.com/xinliupitt/skellam_regression.

———. 2021b. Excess demand prediction for bike sharing systems. *PLOS ONE* 16: e0252894. https://doi.org/10.1371/journal.pone.0252894.

Moreau, J.-P. 2011. Program to calculate the first kind modified Bessel function of integer order N, for any real X, using the function BESSI(N,X).

Skellam, J. G. 1946. The frequency distribution of the difference between two Poisson variates belonging to different populations. *Journal of the Royal Statistical Society,* A ser., 109: 296. https://doi.org/10.2307/2981372.

About the authors

Vincenzo Verardi is associated researcher of the FNRS and professor of econometrics at the University of Namur and the Université libre de Bruxelles (Belgium). His research interests are in applied econometric topics covering, among others, development economics and political economics.

Catherine Vermandele is professor of statistics at the Université libre de Bruxelles (Belgium) and is responsible for the LMTD. She is particularly interested in nonparametric statistics, robust statistical methods, and sampling theory.