

Loef, Hans-Edi

Working Paper

Distributed lags and truncated Pascal distribution

Diskussionsbeiträge, No. 107

Provided in Cooperation with:

Department of Economics, University of Konstanz

Suggested Citation: Loef, Hans-Edi (1978) : Distributed lags and truncated Pascal distribution, Diskussionsbeiträge, No. 107, Universität Konstanz, Fachbereich Wirtschaftswissenschaft und Statistik, Konstanz

This Version is available at:

<https://hdl.handle.net/10419/75115>

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.

FACHBEREICH
WIRTSCHAFTSWISSENSCHAFT UND STATISTIK
UNIVERSITÄT KONSTANZ

Distributed Lags and Truncated
Pascal Distribution

Hans - Edi Loeff

Nr. 107

DISKUSSIONSBEITRÄGE

A 9 611 78
Wirtschaftswissenschaften
Kiel

D-7750 Konstanz
Postfach 7733

Distributed Lags and Truncated
Pascal Distribution

Hans - Edi | Loef

Nr. 107

Januar 1978

As 611 78
Wirtschaft
Kiel

Diskussionsbeiträge des Fachbereichs
Wirtschaftswissenschaft und Statistik
der Universität Konstanz

Introduction

Distributed lag models based on the Pascal distribution assume implicitly that the lag weights are distributed infinitely. These distributed lag procedures cannot select an optimal lag duration. This paper shows how a truncation of the Pascal distribution can be used to estimate the optimal lag length. Starting with an empirical investigation by van Alphen and Merkies (I) it is shown (II) that their procedure cannot answer questions about the optimal time structure of the impact of the exogenous lagged variable on the endogenous variable. Section III develops a more general procedure which is based on a truncation of the Pascal distribution and can select optimal values for the parameters in the Pascal distribution as well as the optimal time structure. The van Alphen and Merkies procedure turns out to be a special case of the truncation method developed. Using van Alphen and Merkies data their results are simulated (IV). It is shown that using their optimum criterion the optimal lag structure is different from the one they assume a priori. But usually the kind of a priori information required is not available and the actual lag duration is not known in advance and must be estimated instead. In an appendix I present the computer program which relies on the model developed previously in the text (section III) and which was used to compute the results in section IV.

In an empirical study van Alphen and Merckies [2] tried to forecast the production of houses (y_t) explained by housing starts (x_t), by means of distributed lags, in the Netherlands for the period IV.1963 - III.1973. As they used a decomposition of the variables over vintages they were able to estimate the lag structure directly. The optimal length of the lag was found to be 11 quarters [2, 415]. This direct estimation was then used to select the most appropriate hypothetical lag structure. The comparison of major candidates (among them the Almon lag procedure [1] and Solow's Pascal distributed coefficient [5]) led them to conclude that "the Pascal approach seems to be the most appropriate" one [2, 420].

Normally, however, such a decomposition of variables is not available and a direct estimation procedure must be substituted by an indirect one. In doing so none of the hypothetical lag structures commonly used generated useful results. In the Pascal case the coefficient λ was either found to be too low or too close to one, compared to $\lambda = 0.5918$ in the Pascal distribution with $r = 2$ using direct estimation. Moreover some of the coefficients in the Almon lag procedure came out with the wrong (negative) sign.

Van Alphen and Merckies proposed a slightly different procedure than that used by Solow, a procedure which was also suggested by Maddala and Rao [3]. The search process by van Alphen and Merckies differs from that by Maddala and Rao only in another factorization of the Pascal distributed lag weights ([2], 427 f.16 and [3], 81 f.1) and starts with the general distributed lag equation (1) [2, 427]:

$$(1) \quad y(t) = \sum_{i=0}^{\infty} w(i) x(t-i) + u(t)$$

Taking Pascal distributed coefficients for the lag weights $w(i)$

$$(2) \quad w(i) = \frac{(1+i)!}{(r-1)!i!} (1-\lambda)^r \lambda^i$$

and setting $r = 2$ and $j = i - t$ van Alphen and Merckies develop

$$(3) \quad y^*(t) = \gamma_1 t \lambda^t + \gamma_2 \lambda^t + u(t)$$

where

$$(4) \quad y^*(t) = y(t) - (1-\lambda)^2 \sum_{i=0}^{t-1} (i+1) \lambda^i x(t-i)$$

$$(5) \quad \gamma_1 = \sum_{j=0}^{\infty} (1-\lambda)^2 \lambda^j x(-j)$$

$$(6) \quad \gamma_2 = \sum_{j=0}^{\infty} (1-\lambda)^2 (j+1) \lambda^j x(-j)$$

Using all available data of $x(t)$ and $y(t)$ with $t = 1, \dots, N=36$ equation (3) can be estimated by means of ordinary least squares, given λ .

The search process used by van Alphen and Merckies for an optimal value of λ can be described as follows:

1. Vary λ in the range $0 < \lambda < 1$ and calculate (4), λ^t and $t \lambda^t$.
2. Estimate γ_1 and γ_2 in (3) by ordinary least squares.
3. Choose that value of λ which minimizes the sum of squares of $u(t)$, taking account of the degrees of freedom, that is

$$\text{Min} \sum_{t=1}^N \left[y(t) - (1-\lambda)^2 \sum_{i=0}^{t-1} (i+1) \lambda^i x(t-i) - \gamma_1 t \lambda^t - \gamma_2 \lambda^t \right]^2$$

with respect to λ .

II

The numerical results of this process are reported by van Alphen and Merckies in their empirical investigation to be $\lambda_{\text{opt}} = 0.50$ with a RE (=square root of sum of squared residuals divided by degrees of freedom) = 69.55. (This result refers to the seasonally unadjusted data where only the lagged variable $x(t)$ is used to explain $y(t)$).

The search process suggested by van Alphen and Merckies rests on two important assumptions: 1) r is set equal to 2, and 2) the upper limit of the lag length in (1) is assumed to approach infinity.

Especially the last assumption is crucial for a) the procedure cannot select an optimal lag duration and b) in using (1) as a forecasting procedure with limited information about $x(t)$ (and $y(t)$), as it is usually the case, the number of delayed periods differs between and varies

for every time period. Certainly the weights $w(i)$ will be close to zero [2, 420] for some $i \geq k+1$, but there is normally no a priori information available about k (which in the van Alphen and Merckies case is a priori known to be 11). For high values of λ_{opt} the value of k for which $w(i)$ vanishes may be quite large.

It might be the case that for some other investigations knowledge about the lag length will be as important as the numerical values of the lag weights.

III

I suggest a modification of the search process mentioned above which can detect the optimal lag structure as well as the optimal λ -value. The more general procedure contains the van Alphen and Merckies method as a special case. It truncates the Pascal distribution of the lag weights at that time period where the squared sum of errors will be minimized. Let L be the number of periods in which $x(t)$ has an impact on $y(t)$ with the weights w as a function of the delayed periods and two parameters λ and r (Pascal distribution):

$$(7) \quad y(t) = \sum_{i=0}^L \binom{r+i-1}{i} \lambda^i (1-\lambda)^r x(t-i) + u(t)$$

where $1 \leq t \leq N$ and $r < L \leq N$
and N is the number of observations.

If we set $L \rightarrow \infty$, or for numerical computations $L = N$ (which yields the same results as $L \rightarrow \infty$), the van Alphen and Merckies case results. To ensure that the number of predetermined

variables in the following regressions does not exceed the number of observations used, L must be greater than r.

The first step follows the method suggested by van Alphen and Merkies. Let

$$(8) \quad y^*(t) = y(t) - \sum_{i=0}^t \binom{r+i-1}{i} \lambda^i (1-\lambda)^r x(t-i)$$

For $t < L$ set

$$(9) \quad y^*(t) = \sum_{k=1}^r \gamma_k \delta_k + u(t) \quad \Big|_{t < L}$$

$$\text{where } \delta_k = t^{k-1} \lambda^t$$

$$\gamma_k = \gamma_k(j, r, \lambda, x(-j)) \\ j = i - t$$

and calculate δ_k given λ and use ordinary least squares to estimate γ_k .

For $t \geq L$ set

$$(10) \quad y^*(t) = - \sum_{j=0}^t \binom{r+j+L-1}{j+L} \lambda^{j+L} (1-\lambda)^r x(t-j-L) + u(t) \quad \Big|_{t \geq L}$$

$$\text{where } j = i - L.$$

Select a value (positive integer) for r and vary λ and L in the ranges

$$0 < \lambda < 1 \\ r < L \leq N$$

in such a way that the sum of squared errors will be a minimum:

$$(11) \quad \text{Min}_{\lambda, L, \tau = \tau_0} \left\{ y^*(t) - \left[\sum_{k=1}^{\tau} \gamma_k \delta_k \right]_{t < L} + \left[\sum_{j=0}^t \binom{r+j+L-1}{j+L} \lambda^{j+L} (1-\lambda)^r x(t-j-L) \right]_{t \geq L} \right\}^2$$

Examples: For $r = 1$ equation (7) reduces to the case of the simple geometric distributed lag model but with the qualification of selecting an optimal L-value. (9) and (10) reduces to

$$y^*(t) = \gamma_1 \delta_1 + u(t) \Big|_{t < L} = \lambda^t (1-\lambda) \sum_{j=0}^L \lambda^j x(-j) + u(t) \Big|_{t < L}$$

and

$$y^*(t) = - \sum_{j=0}^L \lambda^{j+L} (1-\lambda) x(t-j-L) + u(t) \Big|_{t \geq L}$$

In the case for $r = 2$ equation (11) reads

$$\text{Min}_{\lambda, L, \tau=2} \left\{ y^*(t) - \left[\gamma_1 \delta_1 + \gamma_2 \delta_2 \right]_{t < L} + \left[(1-\lambda)^2 \sum_{j=0}^t (1+j+L) \lambda^{j+L} x(t-j-L) \right]_{t \geq L} \right\}^2$$

$$\text{where } \gamma_1 = \lambda^t \quad \delta_1 = (1-\lambda)^2 \sum_{j=0}^L (1+j) \lambda^j x(-j)$$

$$\gamma_2 = t \lambda^t \quad \delta_2 = (1-\lambda)^2 \sum_{j=0}^L \lambda^j x(-j)$$

and for $r = 4$ equation (11) appears in a similar way as,

$$\text{Min}_{\lambda, L, \tau=4} \left\{ y^*(t) - \left[\gamma_1 \delta_1 + \gamma_2 \delta_2 + \gamma_3 \delta_3 + \gamma_4 \delta_4 \right]_{t < L} + \left[(1-\lambda)^4 \sum_{j=0}^t (1+j+L)(2+j+L) \cdot (3+j+L) \lambda^{j+L} x(t-j-L) \right]_{t \geq L} \right\}^2$$

where

$$\begin{aligned} \delta_1 &= \lambda^t ; \quad \gamma_1 = (1-\lambda)^4 \sum_{j=0}^L (j+1)(j+2)(j+3) \lambda^j x(-j) \\ \delta_2 &= t \lambda^t ; \quad \gamma_2 = (1-\lambda)^4 \sum_{j=0}^L [(j+1)(j+2) + (j+3)(2j+3)] \lambda^j x(-j) \\ \delta_3 &= t^2 \lambda^t ; \quad \gamma_3 = (1-\lambda)^4 \sum_{j=0}^L (j+3) \lambda^j x(-j) \\ \delta_4 &= t^3 \lambda^t ; \quad \gamma_4 = (1-\lambda)^4 \sum_{j=0}^L \lambda^j x(-j) \end{aligned}$$

For large values of r and low values of λ it is quite possible that the corresponding δ_k are very alike. In this case the ordinary least square procedure will not be usable in estimating (9) because of singularity in the relevant matrix. The combination $r-\lambda-L$ cannot be selected as an optimal point in this case.

As the range of i in (7) is limited to L the corresponding Pascal distribution is truncated at the point $i = L$. Therefore the requirement for the weights to sum up to one will normally not be fulfilled. But empirical investigations show that the sum of the weights will be in general not lower than 0.95. It seems to be the case that small values of L are associated with small values of λ , so that the weights $w(i)$ are relatively high in the first delayed periods. Figure 2 and the table below show some combinations of L and λ which were estimated using the data of van Alphen and Merckies.

L	λ	$\sum_{i=0}^L w(i)$
4	0.34	0.9805
6	0.43	0.9864
8	0.47	0.9935
10	0.49	0.9974
12	0.50	0.9991

To ensure that the sum of the weights is nevertheless equal to one define the adjusted beta-weights in (12), as

$$(12) \quad \beta(i) = \frac{w(i)}{\sum_{i=0}^L w(i)}$$

which gives

$$\sum_{i=0}^L \beta(i) = 1.$$

IV

Using the van Alphen and Merckies data for x and y alone as before, setting $r = 2$ and varying λ and L in the following ranges

$$\begin{aligned} 0.01 \leq \lambda \leq 0.99 & \quad \text{with steps of } 0.01 \\ 3 \leq L \leq 36 & \quad \text{with steps of } 1 \end{aligned}$$

I found the optimal lag length to be $L = 8$ with $\lambda = 0.47$ and $RE^2 = 4805.86$. For $L = 36$ the van Alphen and Merckies results, $\lambda = 0.50$ and $RE^2 = 4844.19$, are reestimated by our method. The combination $L = 8$, $\lambda = 0.47$ is superior to the latter one and represents the global minimum of RE^2 in the relevant ranges of λ and L. Figure 1 shows λ_{opt} , given L, the chosen L and the corresponding squared residuals adjusted for the degrees of freedom. Surprisingly the RE^2 shows a local maximum at $L = 11$. This is the number of quarters van Alphen and Merckies detected, in using the direct estimation procedure, to be the time in which at least 98% of housing starts are completed. Moreover the lag coefficients in the Almon lag technique show their unwarranted negative sign after (or in few cases beginning with) the eighth quarter [2, 423, table 6]. The procedure of van Alphen and Merckies in comparing different lag techniques is by the way somewhat arbitrary,

for in using some lag techniques (for example Almon) they used a priori information (lag duration) which is normally difficult to get and in others (their search procedure) they do not. In comparing the forecasting qualities of the different procedures they ignore the unequal information input.

Certainly the differences between the RE^2 for different L values are not large and one can doubt if the combination of λ , r and L with the lowest RE^2 is significantly superior to the other combinations (for example Kmenta [3, 494f]). But it makes a great difference if one is interested in knowing the optimal lag duration, in the case where no a priori information is available. The knowledge of the lag duration may in some investigations be as important as the knowledge of the time pattern of the lag weights.

The inclusion of other exogenous variables besides the lagged one [2, 428-429] and setting $r = 2$ leads to the optimal values $\lambda = 0.61$ and $L = 7$ with $RE^2 = 2361.07$, while using the search process of van Alphen and Merckies ($L = 36$) the Pascal coefficient λ is selected to be 0.60 and the RE^2 is 2676.4. The results differ slightly from the reported ones by van Alphen and Merckies because I did not include the other exogenous variables in (9) or (10) but took the differences between $y(t)$ and $y^*(t)$ as a new endogenous variable and used the ordinary least squares procedure. This is necessary because one cannot simply introduce other exogenous variables in (10) and use OLS. But it would be possible if only (9) is used in the case $L = N$. By including other exogenous variables $L = 11$ shows again the highest RE^2 for all combinations λ and L .

The procedure when including other exogenous variables besides the lagged one follows three steps (the explanation refers to the example above):

1) Estimate

$$y^*(t) = \left[\gamma_1 \lambda^t + \gamma_2 t \lambda^t \right]_{t < L} - \left[(1-\lambda)^2 \sum_{j=0}^t (1+j+L) \lambda^{j+L} x(t-j-L) \right]_{t \geq L} + u(t)$$

by choosing λ between 0.01 and 0.99, and L between 3 and 36, where $y^*(t)$ is defined by (8).

2) take the differences

$$y^{**}(t) = y^*(t) - \left[\gamma_1 \lambda^t + \gamma_2 t \lambda^t \right]_{t < L} + \left[(1-\lambda)^2 \sum_{j=0}^t (1+j+L) \lambda^{j+L} x(t-j-L) \right]_{t \geq L}$$

form the equation

$$(13) \quad y^{**}(t) = \sum_{p=1}^3 \delta_p S_p + \theta [z(t) - \alpha - \beta t] + v(t)$$

where S_p = seasonal dummies

$z(t)$ = total construction not yet built at the start of quarter t [2, 421, table 5]

$z(t) - \alpha - \beta t$ = represents business fluctuations around a linear trend

and apply ordinary least squares on (13).

3) Select those values of λ , L , δ_p , θ , α and β for which $[v(t)]^2$ is a minimum.

The global minimum for RE^2 was found (see section above) at $\lambda = 0.61$ and $L = 7$. The parameters of the exogenous variables are as follows (t-ratio in brackets below the parameters):

$$\theta = -0.094549 \\ (-2.41)$$

$$\delta_1 = -34.61 \\ (-1.52)$$

$$\delta_2 = 59.48 \\ (2.65)$$

$$\delta_3 = 72.24 \\ (3.25)$$

$$-\theta\beta = 24.185560 \\ (3.46)$$

$$-\theta\alpha = 222.513993 \\ (2.06)$$

which lead to

$$\beta = 255.80$$

$$\alpha = 2353.42$$

Summary statistics:

$R^2 = 0.82$, F-ratio = 27.47; Degrees of freedom = 30 ;
Durbin Watson = 1.79.

The result for the ordinary least square procedure by including other exogenous variables and assuming $L = 36$, $r = 2$, are reported in the appendix as an output example for the presented computer model.

It is also possible to vary r . In doing so I found the following optimal combinations (the number of other exogenous variables was again assumed to be zero):

r	L	λ	RE^2
2	8	0.47	4805.86
3	8	0.38	4903.41
4	6	0.28	4774.37
5	6	0.24	4767.32

It can be seen that setting $r = 2$ may only lead to a local minimum of RE^2 in selecting $L = 8$ and $\lambda = 0.47$. The general observation that the higher is r the smaller λ will be is in accordance with the results given by Maddala and Rao [4, 81, table 1]. While their minimum residual sum of squares declines steadily when r is increased the data used in my case (which is identical with the data used by van Alphen and Merkies) show first an increase in RE^2 until $r = 3$ and then a decline.

Setting $r = 1$ leads to the global minimum $RE^2 = 4645.24$ with $\lambda = 0.68$ and $L = 36$. A local minimum was found at $\lambda = 0.64$ and $L = 9$ ($RE^2 = 4676.97$). This case refers to the simple geometric lag distribution.

For $r = 5$ a "better" result concerning RE^2 was located at $L = 36$ and $\lambda = 0.99$ with $RE^2 = 3769.66$. But the extremely high λ -value is very implausible because it implies a very low weight $w(i)$ for $i = 0$ ($w(0) = 0.000002$) and $i = 1$ ($w(1) = 0.000009$) and then a continuously rising weight $w(i)$ until the strongest influence of x on y is reached in period 36 with $w(36) = 0.114637$. Therefore I chose the next local minimum for $r = 5$ which is the reported one ($L = 6$ and $\lambda = 0.24$) in the table above.

Summary

The lag procedures of Solow, van Alphen and Merckies and Maddala and Rao based on the Pascal distribution with a full range of the lag weights from zero to infinity can not detect the relevant (optimal) lag duration. Allowing truncation of the Pascal distribution and using the search process of van Alphen and Merckies leads to optimal values for λ and r (the two parameters of the Pascal distribution) as well as the optimal lag duration (range of the lag weights or optimal truncation line). Setting the time lag duration equal to the number of observations leads to the normal search process of van Alphen and Merckies.

References

- [1] Almon, S., The Distributed Lag between Capital Appropriation and Expenditures, *Econometrica*, 33 (1965), 178 - 196.
- [2] van Alphen, H. J. and A.H.Q.M. Merkies, Distributed Lags in Construction: An Empirical Study, *International Economic Review*, 17 (1976), 411 - 430.
- [3] Kmenta, J., *Elements of Econometrics*, New York 1971.
- [4] Maddala, G.S. and A.S. Rao, Maximum Likelihood Estimation of Solow's and Jorgenson's Distributed Lag Models, *Review of Economics and Statistics*, 53 (1971), 80 - 88.
- [5] Solow, R.M., On a Family of Lag Distributions, *Econometrica*, 28 (1960), 393 - 406.

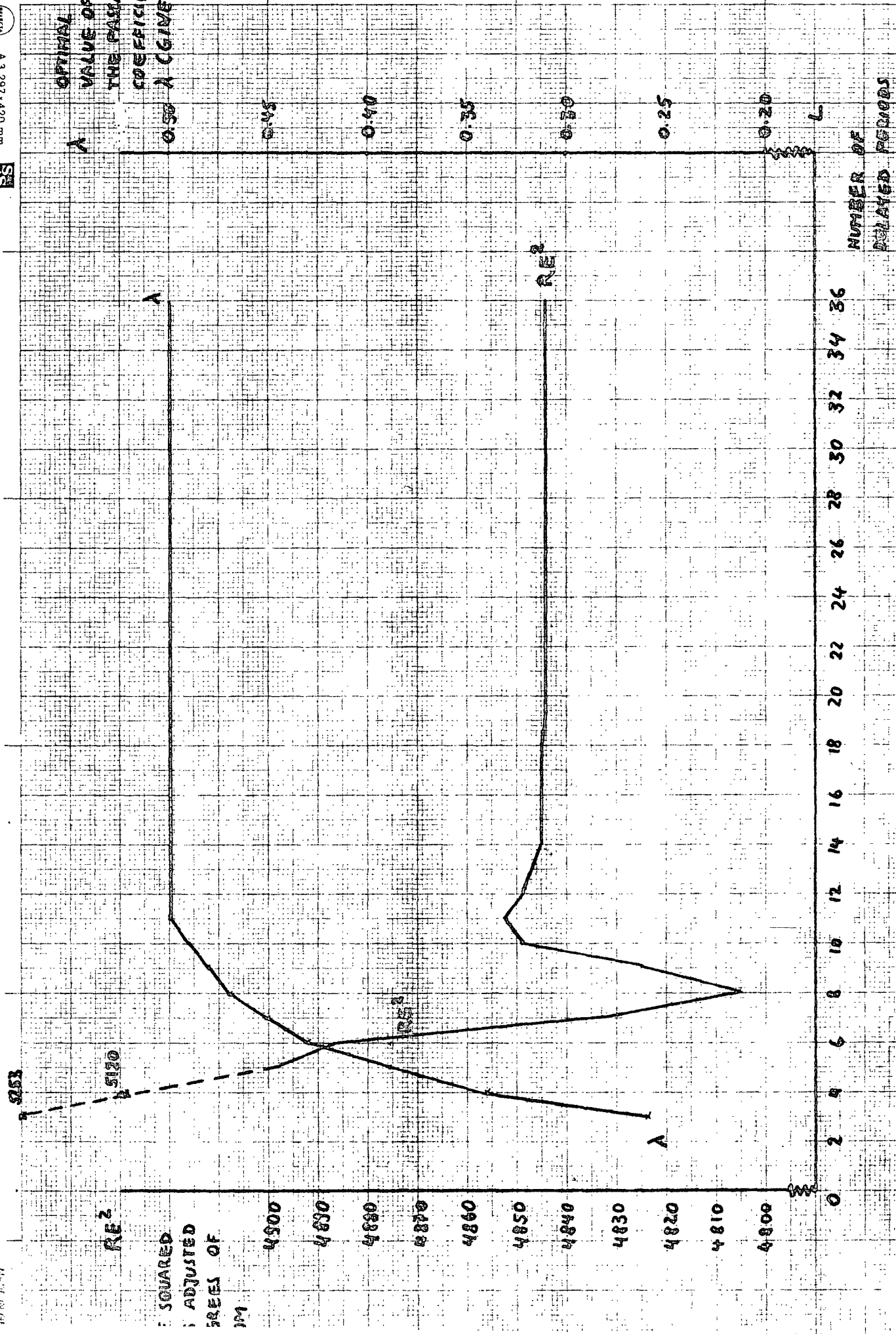


FIGURE 1: THE RELATIONSHIP BETWEEN THE NUMBER OF DELAYED PERIODS (L) AND THE OPTIMAL VALUE OF λ AND THE SUM OF SQUARED ERRORS (RE²)

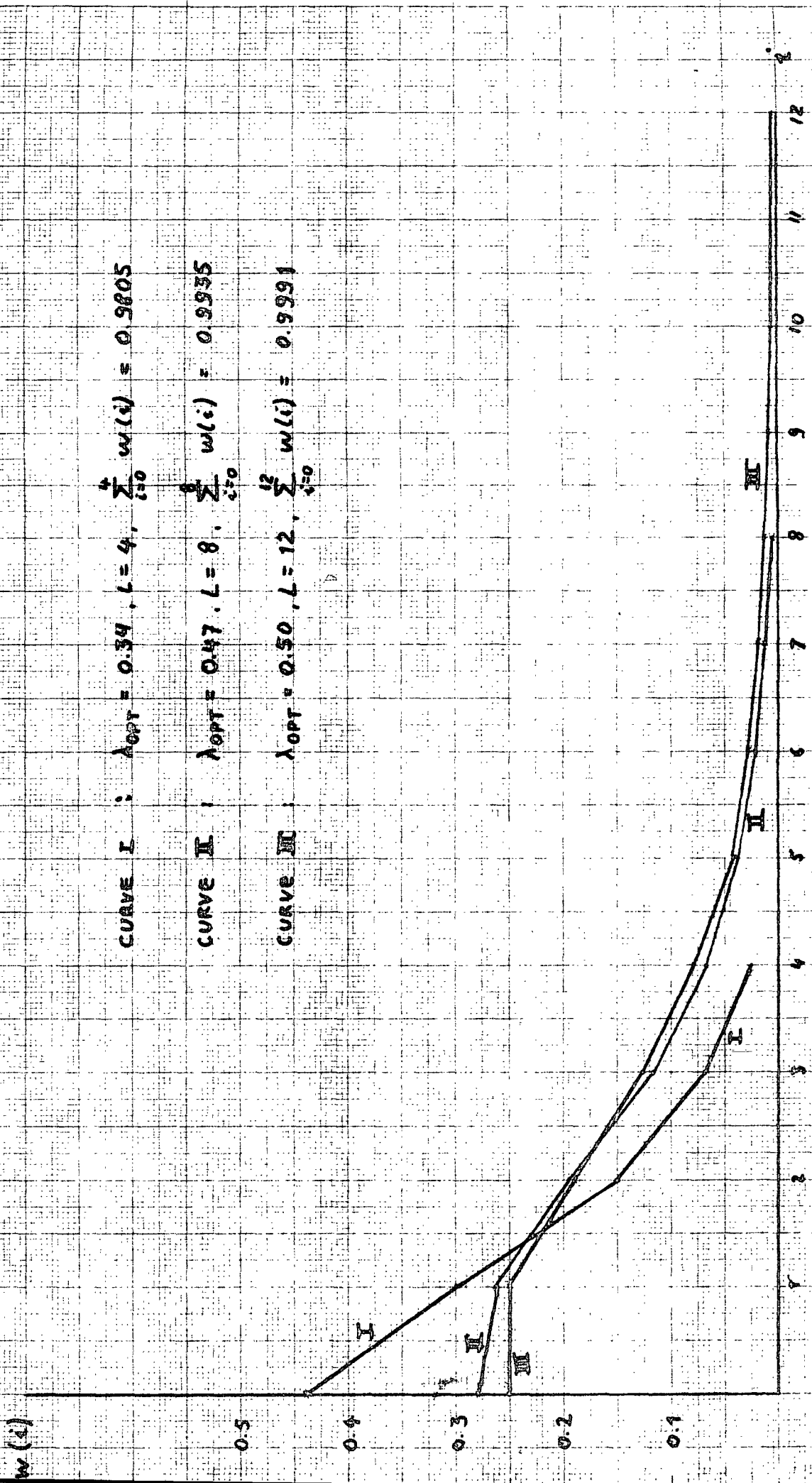


FIGURE 2: RELATIONSHIP BETWEEN λ , L AND SUM OF LAG-WEIGHTS

Appendix: Computer Program for the lag procedure based on truncated Pascal distribution using van Alphen and Merkies' search process

Introduction: The computer program is written in ALGOL 60. There is no data format requirement except that between two numbers there have to be at least two blanks. The input requirements are given below.

Input:

1. N = number of observations
2. L1 = lower limit of lag space in which the optimal number of delayed periods is searched
3. L2 = upper limit of lag space in which the optimal number of delayed periods is searched
4. E = number of exogenous variables besides the lagged variable
5. R = r - parameter of the Pascal distribution (R 1)

all numbers 1) - 5) must be of type integer

6. data for the endogenous variable (1 to N data)
7. data for lagged variable (1 to N data)
8. if E is not equal zero:
data for the other exogenous variables,
each set containing 1 to N data

Output: The following output-example illustrates the way the results are presented.

This example used the data given by van Alphen and Merkies (referred to in the text), for the case $L = 36$, $r = 2$ and additionally six other exogenous variables. The input set for the first five numbers: $N = 36$, $L1 = L2 = 36$, $E = 6$ and $R = 2$.

DISTRIBUTED LAGS WITH TRUNCATED PASCAL DISTRIBUTION

THE OPTIMAL LAG DURATION IS SEARCHED BETWEEN 36 AND 36 PERIODS

R-PARAMETER OF THE PASCAL DISTRIBUTION: 2

NUMBER OF EXOGENOUS VARIABLES BESIDES LAGGED VARIABLE: 6

LAG DURATION: 36 PERIODS

OPTIMAL LAMDA VALUE: 0.60

SUM OF SQUARED ERRORS: 2676.40

STANDARD ERROR OF ESTIMATION: 51.73

(BOTH ADJUSTED FOR 28 DEGREES OF FREEDOM)

THE OPTIMAL LAG DURATION IS 36 PERIODS

OPTIMAL LAMDA VALUE: 0.60

SUM OF SQUARED ERRORS: 2676.40

STANDARD ERROR OF ESTIMATION: 51.73

(BOTH ADJUSTED FOR 28 DEGREES OF FREEDOM)

REGRESSION-ANALYSIS FOR THE OTHER (NOT DELAYED) EXOGENOUS VARIABLES
(THE ENDOGENOUS VARIABLE IS ADJUSTED FOR THE EXPLANATION OF THE LAGGED VARIABLE)

VARIAB.	PARAMETER	STANDARD- DEVIATION	T-RATIO	DEGREES OF FREEDOM
1	$\theta =$ -0.096981	0.040761	-2.379294	30
2	$\delta_1 =$ -33.055163	24.259177	-1.362584	30
3	$\delta_2 =$ 62.250072	23.904829	2.604079	30
4	$\delta_3 =$ 69.446710	23.689818	2.931500	30
5	$\theta\beta =$ 20.170105	7.436771	2.712213	30
6	$\theta\alpha =$ 223.438710	115.122486	1.940878	30

R*R (ADJUSTED)	R*R	F-RATIO	DEG. OF FREEDOM
0.54258	0.61882	9.74061	30

DURBIN-WATSON	AUTOREGRESSION- COEFFICIENT OF THE RESIDUALS	DEGREES OF FREEDOM N-1
1.55002441	.16060123	35

DISTRIBUTION OF THE LAG WEIGHTS AND THE CORRESPONDING BETA WEIGHTS

DELAYED PERIODS	WEIGHTS	BETA-WEIGHTS
0	0.160000	0.160000
1	0.192000	0.192000
2	0.172800	0.172800
3	0.138240	0.138240
4	0.103680	0.103680
5	0.074650	0.074650

6	0.052255	0.052255
7	0.035832	0.035832
8	0.024186	0.024186
9	0.016124	0.016124
10	0.010642	0.010642
11	0.006966	0.006966
12	0.004528	0.004528
13	0.002926	0.002926
14	0.001881	0.001881
15	0.001204	0.001204
16	0.000767	0.000767
17	0.000487	0.000487
18	0.000309	0.000309
19	0.000195	0.000195
20	0.000123	0.000123
21	0.000077	0.000077
22	0.000048	0.000048
23	0.000030	0.000030
24	0.000019	0.000019
25	0.000012	0.000012
26	0.000007	0.000007
27	0.000005	0.000005
28	0.000003	0.000003
29	0.000002	0.000002
30	0.000001	0.000001
31	0.000001	0.000001

THE WEIGHTS OF THE DELAYED PERIODS 32 TO 36 ARE SMALLER THAN 0.000000


```
'BEGIN'  
'COMMENT' DISTRIBUTED LAGS BASED ON TRUNCATED  
PASCAL DISTRIBUTION;  
'PROCEDURE' MIVACOR(A,N,M) RESULT:(MI,VAR,KOR);  
'VALUE' A,N,M;  
'INTEGER' N,M;  
'ARRAY' A,MI,VAR,KOR;  
'BEGIN'  
  'INTEGER' I,J,K;  
  'REAL' S1,S2;  
  'FOR' J:=1 'STEP' 1 'UNTIL' N 'DO'  
  'BEGIN'  
    S1:=S2:=0;  
    'FOR' I:=1 'STEP' 1 'UNTIL' N 'DO'  
    'BEGIN'  
      S1:=S1+ACI(I,J);  
      S2:=S2+ACI(I,J)*ACI(I,J);  
    'END';  
    MIC(J):=S1/N;  
    VAR(J,J):=(S2-MI(J)*MI(J)*N)/(N-1);  
    KOR(J,J):=1;  
  'END';  
  'FOR' J:=1 'STEP' 1 'UNTIL' N 'DO' 'FOR' K:=1 'STEP' 1 'UNTIL' J-1 'DO'  
  'BEGIN'  
    S1:=0;  
    'FOR' I:=1 'STEP' 1 'UNTIL' N 'DO' S1:=S1+ACI(I,J)*ACI(I,K);  
    VAR(J,K):=(S1-N*MIC(J)*MIC(K))/(N-1);  
    VAR(K,J):=VAR(J,K);  
    S2:=VAR(J,J)*VAR(K,K);  
    'IF' S2 'NOT' GREATER '0' THEN 'KOR(J,K):=0'  
    'ELSE' 'KOR(J,K):=VAR(J,K)/(SQRT(S2));  
    KOR(K,J):=KOR(J,K);  
  'END';  
'END' MIVACOR;  
'PROCEDURE' QUADMULT(A1,A3,N,M);  
'VALUE' A1,N,M;  
'INTEGER' N,M;  
'ARRAY' A1,A3;  
'BEGIN'  
  'INTEGER' I,IO,KO;  
  'REAL' S;  
  'FOR' IO:=1 'STEP' 1 'UNTIL' M 'DO' 'FOR' KO:=1 'STEP' 1 'UNTIL' M 'DO'  
  'BEGIN'  
    S:=0;  
    'FOR' I:=1 'STEP' 1 'UNTIL' N 'DO' S:=S+A1(I,IO)*A1(I,KO);  
    A3(IO,KO):=S;  
  'END';  
'END' QUADMULT;  
'PROCEDURE' TRANSP(M,N,A1,A2);  
'VALUE' N,M,A1;  
'INTEGER' N,M;  
'ARRAY' A1,A2;  
'BEGIN'  
  'INTEGER' I,K;
```

```
'FOR' 'K'=1' 'STEP' '1' 'UNTIL' 'N' 'DO' 'FOR' 'I'=I' 'STEP' '1' 'UNTIL' 'N' 'DO' 'A(ZK, I):=A(I, K);  
'END' 'TRANSP;  
'PROCEDURE' 'INVERSE' (A, N, EPS) 'EXIT' (SINGULAR);  
'VALUE' 'N;  
'ARRAY' 'A;  
'INTEGER' 'N;  
'REAL' 'EPS;  
'LABEL' 'SINGULAR;  
'BEGIN'  
'INTEGER' 'I, J, K;  
'REAL' 'PIVOT, Z;  
'ARRAY' 'B, C[1:N];  
'INTEGER' 'ARRAY' 'P, Q[1:N];  
'FOR' 'K'=1' 'STEP' '1' 'UNTIL' 'N' 'DO'  
'BEGIN'  
PIVOT:=0;  
'FOR' 'I'=K' 'STEP' '1' 'UNTIL' 'N' 'DO' 'FOR' 'J'=K' 'STEP' '1' 'UNTIL' 'N' 'DO' 'IF' 'ABS' (A(I, J))  
'GREATER' 'ABS' (PIVOT) 'THEN'  
'BEGIN'  
PIVOT:=A(I, J);  
P(K):=I;  
Q(K):=J  
'END';  
'IF' 'ABS' (PIVOT) 'NOT' 'GREATER' 'EPS' 'THEN' 'GOTO' 'SINGULAR;  
'IF' 'P(K) 'NOT' 'EQUAL' 'K' 'THEN' 'FOR' 'J'=1' 'STEP' '1' 'UNTIL' 'N' 'DO'  
'BEGIN'  
Z:=A(P(K), J);  
A(P(K), J):=A(K, J);  
A(K, J):=Z  
'END' 'J;  
'IF' 'Q(K) 'NOT' 'EQUAL' 'K' 'THEN' 'FOR' 'I'=1' 'STEP' '1' 'UNTIL' 'N' 'DO'  
'BEGIN'  
Z:=A(I, Q(K));  
A(I, Q(K)):=A(I, K);  
A(I, K):=Z  
'END' 'I;  
'FOR' 'J'=1' 'STEP' '1' 'UNTIL' 'N' 'DO'  
'BEGIN'  
'IF' 'J' 'EQUAL' 'K' 'THEN'  
'BEGIN'  
B(J):=1.0/PIVOT;  
C(J):=1.0  
'END'  
'ELSE'  
'BEGIN'  
B(J):=-A(K, J)/PIVOT;  
C(J):=A(I, J);  
'END';  
A(K, J):=A(I, J);  
'END' 'J;  
'FOR' 'I'=1' 'STEP' '1' 'UNTIL' 'N' 'DO' 'FOR' 'J'=1' 'STEP' '1' 'UNTIL' 'N' 'DO' 'A(I, J):=A(I, J)  
'-B(J)  
'END' 'K;  
'FOR' 'K'=N' 'STEP' '-1' 'UNTIL' '1' 'DO'  
'BEGIN'  
'IF' 'P(K) 'NOT' 'EQUAL' 'K' 'THEN' 'FOR' 'I'=1' 'STEP' '1' 'UNTIL' 'N' 'DO'  
'BEGIN'  
Z:=A(I, P(K));  
A(I, P(K)):=A(I, K);  
A(I, K):=Z;  
'END' 'I;  
'IF' 'Q(K) 'NOT' 'EQUAL' 'K' 'THEN' 'FOR' 'J'=1' 'STEP' '1' 'UNTIL' 'N' 'DO'
```

```
'BEGIN'  
  Z:=A[Q[K],J];  
  A[Q[K],J]:=A[K,J];  
  A[K,J]:=Z;  
'END'  
'END'  
'END'  
PROCEDURE MULT(F,K,L,G,M,N) RESULT: (H) EXIT: (UNVERTRAEGlich);  
  VALUE K,L,M,N;  
  ARRAY F,G,H;  
  INTEGER K,L,M,N;  
  LABEL UNVERTRAEGlich;  
  BEGIN  
    INTEGER I,J,R;  
    REAL S;  
    IF L NOT EQUAL M THEN GOTO UNVERTRAEGlich;  
    FOR I:=1 STEP 1 UNTIL K DO FOR J:=1 STEP 1 UNTIL N DO  
      BEGIN  
        S:=0;  
        FOR R:=1 STEP 1 UNTIL L DO S:=S+F[R,I]*G[R,J];  
        H[I,J]:=S;  
      END  
    END  
  END  
PROCEDURE OLS(AA3,G1,D,P,DI, EPS, SINGULAR, UNVERTRAEGlich);  
  INTEGER DI;  
  REAL EPS;  
  ARRAY AA3,G1,D,P;  
  LABEL SINGULAR, UNVERTRAEGlich;  
  BEGIN  
    INTEGER I,K;  
    FOR K:=1 STEP 1 UNTIL DI DO FOR I:=1 STEP 1 UNTIL DI DO D[I,K]:=AA3[I+1,  
+I];  
    FOR I:=1 STEP 1 UNTIL DI DO G1[I,1]:=AA3[I+1,1];  
    INVERSE(D,DI, EPS) EXIT: (SINGULAR);  
    MULT(D,DI,G1,DI,1) RESULT: (P) EXIT: (UNVERTRAEGlich);  
  END OLS;  
PROCEDURE UDACH(X,P,YD,UD,N1,DI, AKKI, UNVERTRAEGlich);  
  VALUE P;  
  INTEGER DI,N1;  
  ARRAY X,P,YD,UD;  
  REAL AKKI;  
  LABEL UNVERTRAEGlich;  
  BEGIN  
    INTEGER I,K;  
    REAL S1,S2;  
    ARRAY A11[I:1:N1,I:DI];  
    S1:=S2:=0;  
    FOR K:=1 STEP 1 UNTIL DI DO FOR I:=1 STEP 1 UNTIL N1 DO A11[I,K]:=X[I,K];  
    MULT(A11,N1,DI,P,DI,1,YD, UNVERTRAEGlich);  
    FOR I:=1 STEP 1 UNTIL N1 DO UDC[I]:=X[I,1]-YD[I,1];  
    FOR I:=1 STEP 1 UNTIL N1-1 DO S1:=S1+UDC[I]*UDC[I+1];  
    FOR I:=1 STEP 1 UNTIL N1 DO S2:=S2+UDC[I]*UDC[I];  
    AKKI:=(S1/(N1-1))/(S2/(N1-DI));  
  END UDACH;  
PROCEDURE WERTE(Y,YD,UD,N1,DI, RQI,DWI, FWI, UO,U1,YT1,WRI);  
  VALUE Y,YD,UD,N1,DI;  
  INTEGER N1,DI;  
  ARRAY YD,UD,Y;  
  REAL RQI,DWI, FWI,UO,U1,YT1,WRI;  
  BEGIN  
    INTEGER I,K;
```

```

REAL S1, S2, S3;
S1:=S2:=S3:=0;
U0:=U1:=YT1:=0;
FOR I:=1 STEP 1 UNTIL 'N1' DO
  BEGIN
    S2:=S2+Y[I]*Y[I];
    S3:=S3+Y[I];
    U0:=U0+U0[I];
    U1:=U1+U0[I]*U0[I];
    YT1:=YT1+YDC[I,1]*YDC[I,1];
  END;
RQI:=(N1*U1)/(N1*S2-S3*S3);
WRI:=1-RQI;
RQI:=1-RQI*(N1/(N1-DI));
IF WRI GREATER 0 THEN FWI:=((N1-DI)*WRI)/((DI-1)*(1-WI));
ELSE FWI:=0;
FOR I:=1 STEP 1 UNTIL 'N1-1' DO S1:=S1+(U0[I]-U0[I+1])*(U0[I]-U0[I+1]);
DWI:=S1/U1;
END WERTE;
PROCEDURE REG(N, M, UU, ANTON, EN, LL, P, Q);
VALUE N, M, ANTON, EN, LL, UU;
INTEGER N, M, LL, UU, Q;
ARRAY ANTON, EN, P;
BEGIN
  ARRAY A1[1:N-UU, 1:M], AS2[1:M, 1:M], AS2[1:M], EN1[1:N-UU];
  INTEGER I1, K1;
  FOR I1:=UU+1 STEP 1 UNTIL 'N' DO
    BEGIN
      A1[I1-UU, 1]:=EN1[I1];
      EN1[I1-UU]:=EN1[I1];
    END;
  FOR I1:=UU+1 STEP 1 UNTIL 'N' DO FOR K1:=2 STEP 1 UNTIL 'M' DO A1[I1-UU, K1]
  ANTON[K1-1, I1];
  N:=N-UU;
  MIYACOR(A1, N, M, AS2, AS2, KM);
  BEGIN
    REAL EPS;
    ARRAY DC[1:M-1, 1:M-1], G1[1:M-1, 1:1], AA3[1:M, 1:M];
    EPS:=0.00000001;
    QUADMULT(A1, AA3, N, M);
    OLS(AA3, G1, D, P, M-1, EPS, SINGULAR, UNVERTRAEGLICH);
  BEGIN
    REAL YTY, YT1, WR, RQ, AKK, FW, DW, U0, U1, U11;
    ARRAY YTC[1:N, 1:1], MPBA, DSIC[1:M-1], UC[1:M];
    UDACH(A1, P, YTC, U, N, M-1, AKK, UNVERTRAEGLICH);
    IF LL EQUAL 0 THEN GOTO ENDE;
    WERTE(EN1, YTC, U, N, M-1, RQ, DW, FW, U0, U1, YT1, WR);
    FOR I1:=1 STEP 1 UNTIL 'M-1' DO
      BEGIN
        DS1[I1]:=(U1*DC[I1, I1])/(N-M+1);
        IF DS1[I1] LESS 0 THEN DS1[I1]:=0;
        ELSE DS1[I1]:=SORT(DS1[I1]);
        IF DS1[I1] EQUAL 0 THEN MPBA[I1]:=9999;
        ELSE MPBA[I1]:=P[I1, 1]/DS1[I1];
      END;
    OUTPUT(4, '(15/)' ('REGRESSION-ANALYSIS FOR THE OTHER '),
    '(1 (NOT DELAYED) EXOGENOUS VARIABLES) / (1) ');
    OUTPUT(4, '(1)' ('(THE ENDOGENOUS VARIABLE IS ADJUSTED FOR THE '),
    '(EXPLANATION OF THE LAGGED VARIABLE)') (5/1) ');
    OUTPUT(4, '(1)' ('VARIAB. PARAMETER STANDARD- ') (1) ');
    OUTPUT(4, '(1)' (' T-RATIO DEGREES OF ') (1) ');
    OUTPUT(4, '(124B)' (' DEVIATION') (1) (23B) ('FREEDOM') (1) ');
  
```

```
'FOR' I1:=1 'STEP' 1 'UNTIL' N-1 'DO'  
'BEGIN'  
  OUTPUT(4, '(1/2D)', I1);  
  OUTPUT(4, '(3(B-7ZD.6D), 4B&Z)', PC[I1, 1], DS1[I1], MPBAC[I1], N-M+1);  
'END';  
  OUTPUT(4, '(1575B)', I1);  
  OUTPUT(4, '(1)'('R*(ADJUSTED) RMR F-RATIO DEG. OF FREEDOM'  
(I1));  
  OUTPUT(4, '(12/3(B-4ZD.5D), 6ZD, 4/I1), RQ, WR, FW, N-M+1);  
  OUTPUT(4, '(1/1)'('DURBIN-WATSON AUTOREGRESSION- DEGREES CF'))'/'  
(I1);  
  OUTPUT(4, '(117B)'('COEFFICIENT OF THE')'BB'('FREEDOM')'/')');  
  OUTPUT(4, '(117B)'('RESIDUALS')'12B'('N-1')'/')');  
  OUTPUT(4, '(12/2(B-4Z.8D), 6B5Z/1)', DW, AKK, N-1);  
'END';  
'END';  
'GOTO' ENDE;  
UNVERTRAEGLICH:  
  OUTPUT(4, '(1)'('UNVERTRAEGLICH')'/')');  
'GOTO' ENDE;  
SINGULAR:  
  OUTPUT(4, '(1)'('SINGULAR AT K = ')ZD/1)', Q);  
  OUTPUT(4, '(1)'('DO NOT USE THE CORRESPONDING LAG-WEIGHTS')'/')');  
  OUTPUT(4, '(1)'('FOR LAMDA = ')D.DD'('')'/')', Q*0.01);  
ENDE:  
'END';  
'COMMENT' 'PROZEDUR-REG-ENDE';  
'PROCEDURE' 'GEWICHTE(LAM, RR, LL, GEW, BETA);  
'INTEGER' RR, LL;  
'REAL' LAM;  
'ARRAY' GEW, BETA;  
'BEGIN'  
  'REAL' SUM, F;  
  'INTEGER' I, J;  
  SUM:=0, 0;  
  'FOR' I:=0 'STEP' 1 'UNTIL' LL 'DO'  
  'BEGIN'  
    F:=1.0;  
    'FOR' J:=1 'STEP' 1 'UNTIL' RR-1 'DO' F:=F*(I+J)/J;  
    GEW[IJ]:=F*((1-LAM)'POWER' RR)*LAM'POWER' I;  
    SUM:=SUM+GEW[IJ];  
  'END';  
  'FOR' I:=0 'STEP' 1 'UNTIL' LL 'DO' BETA[IJ]:=GEW[IJ]/SUM;  
  OUTPUT(4, '(15/1)'('DISTRIBUTION OF THE LAG WEIGHTS AND THE')'/')');  
  OUTPUT(4, '(1)'('CORRESPONDING BETA WEIGHTS')'/')');  
  OUTPUT(4, '(1)'('DELAYED PERIODS WEIGHTS BETA-WEIGHTS')'/')');  
  'FOR' I:=0 'STEP' 1 'UNTIL' LL 'DO'  
  'BEGIN'  
    'IF' GEW[IJ] 'LESS' 0.0000005 'THEN' 'GOTO' MM1;  
    OUTPUT(4, '(10BZD, 10BZD.6D, 3BZD.6D//)', I, GEW[IJ], BETA[IJ]);  
  'END';  
  'GOTO' MM2;  
MM1:  
  OUTPUT(4, '(1/1)'('THE WEIGHTS OF THE DELAYED')', '(1)'('PERIODS')'ZD, '(1)'('TC'  
ZD, '(1)'('ARE SMALLER THAN 0.0000005')'15/1)', I, LL);  
MM2:  
'END';  
'COMMENT' 'END OF PROCEDURE GEWICHTE';  
'INTEGER' DDD, R, H, N, L, W1, W2;  
'INTEGER' 'ARRAY' DC1:5J;  
INARRAY(2), D);  
DDD:=DC1J;
```

```

R:=DC[2];
H:=0;
N:=DC[1];
'COMMENT' D1 = NUMBER OF OBSERVATIONS          D2 = MIN - PERIODS OF LAG STRU
CTURE          D3 = MAX - PERIODS OF LAG STRU     D4 = NUMBER OF EXOGE
NOUS VARIABLES OTHER THAN LAGGED          D5 = R-PARAMETER OF PASCAL DISTRIBUTIO
N;
'IF' DC[4] 'EQUAL' '0' 'THEN' DC[4] := 1;
'BEGIN'
'REAL' LAMDA, B, A, MIN, E;
'INTEGER' I, J, K, Z, P;
'ARRAY' Y, X, YSTAR, DIFF[1:N], XSTAR[1:R, 1:DC[3]], ALPHA[1:R, 1:1], YY[1:DC[3]], CC[
+DC[2]:N+1], OPT[1:DC[3]-DC[2]+1, 1:99], GG[1:DC[3]-DC[2]+1], EX[1:DC[4], 1:N], KOC[1:DC[4], 1:
1];
OUTPUT(4, ('*5/' ('DISTRIBUTED LAGS WITH TRUNCATED PASCAL ') ('')));
OUTPUT(4, ('(' ('DISTRIBUTION') / ') ('')));
OUTPUT(4, ('(' ('*****') ('')) ('')));
OUTPUT(4, ('(' ('*****') / ') ('')));
OUTPUT(4, ('(' ('THE OPTIMAL LAG DURATION IS SEARCHED BETWEEN') ('AND'
'), BZDB ('PERIODS') (''))', DC[2], DC[3]));
OUTPUT(4, ('///' ('R-PARAMETER OF THE PASCAL DISTRIBUTION: ') 'ZD//')', DC[5]);
OUTPUT(4, ('(' ('NUMBER OF EXOGENOUS VARIABLES BESIDES LAGGED ') ('VARIABLE
E: ') 'ZD3/')', DDD);
INARRAY(21, Y);
INARRAY(21, X);
DC[4] := DDD;
W1 := 1;
W2 := 99;
'IF' DC[4] 'NOT EQUAL' '0' 'THEN' INARRAY(21, EX);
M1:
H := H + 1;
L := H - 1 + DC[2];
M2:
'FOR' K := W1 'STEP' 1 'UNTIL' W2 'DO'
'BEGIN'
GG[H] := 0.01 * K;
OPT[H, K] := 0, 0;
A := (1 - GG[H]) 'POWER' R;
'FOR' I := 1 'STEP' 1 'UNTIL' N 'DO'
'BEGIN'
B := 0, 0;
'FOR' J := 0 'STEP' 1 'UNTIL' I - 1 'DO'
'BEGIN'
E := 1.0;
'FOR' P := 1 'STEP' 1 'UNTIL' R - 1 'DO' E := E * (J + P) / P;
B := B + A * E * X[I - J] * GG[H] 'POWER' J;
'END';
YSTAR[I] := Y[I] - B;
'END';
'FOR' P := 1 'STEP' 1 'UNTIL' R 'DO' ALPHA[P, I] := 0, 0;
'FOR' I := 1 'STEP' 1 'UNTIL' L 'DO'
'BEGIN'
DIFF[I] := Y[I] := YSTAR[I];
XSTAR[1, I] := GG[H] 'POWER' I;
'FOR' P := 2 'STEP' 1 'UNTIL' R 'DO' XSTAR[P, I] := XSTAR[P - 1, I] * I;
'END';
REG(L, R + 1, 0, XSTAR, YY, 0, ALPHA, K);
'FOR' I := 1 'STEP' 1 'UNTIL' L 'DO' 'FOR' P := 1 'STEP' 1 'UNTIL' R 'DO' DIFF[I] := DIFF[I
- ALPHA[P, I] * XSTAR[P, I];
'FOR' I := L + 1 'STEP' 1 'UNTIL' N 'DO'
'BEGIN'
CC[I] := 0, 0;

```

```

FOR J:=0 STEP 1 UNTIL I-L-1 DO
  BEGIN
    E:=1.0;
    FOR D:=1 STEP 1 UNTIL N-1 DO E:=E*(1+P+L)/P;
    CC(I):=CC(I)+A*E*(X(I)-J-L)*GGCH(I)*POWER(I+L);
  END;
DIFF(I):=YSTAR(I)+CC(I);
END;
IF D[4] NOT EQUAL 0 THEN
  BEGIN
    IF W1 NOT EQUAL W2 THEN REG(N,D[4]+1,0,EX,DIFF,0,KD,K);
    IF W1 EQUAL W2 THEN
      BEGIN
        REG(N,D[4]+1,0,EX,DIFF,1,KD,K);
        GOTO M3;
      END;
    IF W4 EQUAL W2 THEN GOTO M3;
  END;
FOR I:=1 STEP 1 UNTIL N DO
  BEGIN
    FOR J:=1 STEP 1 UNTIL D[4] DO DIFF(I):=DIFF(I)-K*EX(J,I);
    OPTCH,K:=OPTCH,K+(DIFF(I)**2)/(N-2-D[4]);
  END;
END;
COMMENT END OF K;
MIN:=OPTCH,1;
GGCH:=0.01;
FOR K:=2 STEP 1 UNTIL 99 DO
  BEGIN
    IF OPTCH,K LESS MIN THEN
      BEGIN
        MIN:=OPTCH,K;
        GGCH:=0.01*K;
      END;
  END;
OUTPUT(4,('('/'('LAG DURATION: ')ZDB('PERIODS')/(')')L));
OUTPUT(4,('('/'('OPTIMAL LAMDA VALUE: ')D.DD3/(')')GGCH));
OUTPUT(4,('('/'('SUM OF SQUARED ERRORS: ')6ZD.DD7/(')')MIN));
OUTPUT(4,('('/'('STANDARD ERROR OF ESTIMATION: ')5ZD.DD7/(')')SQRT(MIN));
OUTPUT(4,('('/'('BOTH ADJUSTED FOR ')ZDB,('DEGREES OF ')(')')N-2-D[4]);
OUTPUT(4,('('/'('FREEDOM')/(')'));
IF H LESS D[3]-D[2]+1 THEN GOTO M1;
MIN:=OPTCH,1;
LAMBDA:=GGCH;
Z:=D[2];
FOR H:=1 STEP 1 UNTIL D[3]-D[2]+1 DO FOR K:=1 STEP 1 UNTIL 99 DO
  BEGIN
    IF OPTCH,K LESS MIN THEN
      BEGIN
        MIN:=OPTCH,K;
        W1:=K;
        Z:=H*D[2]-1;
        LAMBDA:=GGCH;
      END;
  END;
END;
H:=Z-D[2]+1;
GGCH:=LAMBDA;
W2:=W1;
L:=Z;
OUTPUT(4,('(*5/'('THE OPTIMAL LAG DURATION IS: ')ZDB('PERIODS')/(')')Z));
OUTPUT(4,('('/'('OPTIMAL LAMDA VALUE: ')D.DD7/(')')LAMBDA));
OUTPUT(4,('('/'('SUM OF SQUARED ERRORS: ')6ZD.DD7/(')')MIN));
OUTPUT(4,('('/'('STANDARD ERROR OF ESTIMATION: ')5ZD.DD7/(')')SQRT(MIN));

```

```
OUTPUT(4, '( '( (BOTH ADJUSTED FOR ) 'ZZDB' (DEGREES OF FREEDOM) ) '//' ) 'AN-  
2-DE4 ] );  
GOTO M2;  
M3;  
GEWICHTE(LAMDA, R, Z, YY, YY);  
END;  
END;
```