DK9901202

PETRA

- Technical implementation

PETRA Working Paper no. 9

September 1998

COWI

COWI

PETRA
- Technical implementation

PETRA Working Paper no. 9

September 1998

Report no.: 1
Issue no.: 1
Date of issue: September 1998

Prepared: JJD
Checked: MOF
Approved:

Table of Contents

1 Introduction

This report describes the technical implementation of PETRA. The report is intended for specialist users and refers to the series of reports describing the project, a list is given in chapter 5.

The PETRA system consists of a number of econometric models, representing the national travel demand in Denmark. Application of these models requires the definition of a scenario, a number of runs with the individual models and extensive data transfer between the models and a database containing base data and results.

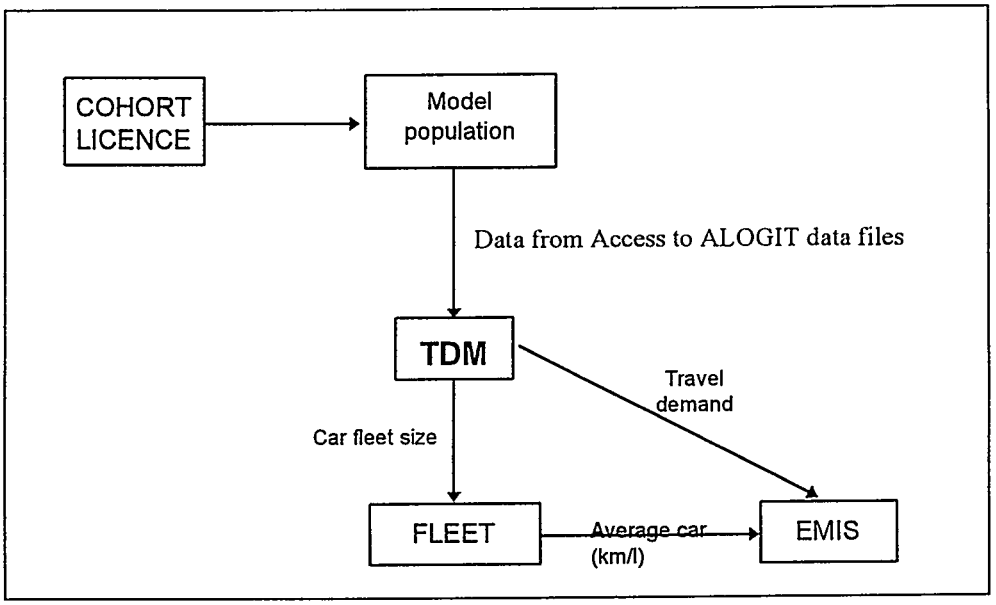
The system contains three basic scenarios to which changes in various assumptions can be applied. It is possible to construct more basic scenarios but this is outside the scope of using the model as it presently stands.

The focus of this report is thus on the specification of changes to basic scenario, on running the model - including description of the data flows, and on the possibilities for analysis of the results.

1.1 Overview of PETRA

The overall structure of PETRA is illustrated in Figure 1-1.

Figure 1-1 Model structure in PETRA



COHORT and LICENCE

COHORT and LICENCE generate licence holding data on individual levels for each of the basic scenarios.

Model population

Licence holding information is combined with other background information on the TU individuals and a set of weights that scale the individual records to the national level. PETRA thus contains a model population representing the Danish population in each of the three basic scenarios that have been constructed.

TDM

The model population is fed through the Travel Demand Module, the TDM, to generate forecast car availability and travel demand for each individual in the population.

FLEET

The total number of cars predicted is output to the (still quite rudimentary) FLEET module.

EMIS

EMIS applies emission rates for the various transport modes to calculate total emissions for the forecast travel.

Changes to scenarios

Basic scenarios are defined by constructing a new model population. Changes to basic scenarios, on the other hand, only involve changes in the running of the TDM - the core of PETRA - which predicts car availability, choice of tour type and mode and destination choice.

2 TDM in PETRA

2.1 Model code

PETRA has been implemented in Microsoft Access version 7 and ALOGIT. Access controls the runs, generates data files for ALOGIT, calls ALOGIT and collects the data results afterwards.

The sequence of commands controlling a run is shown in Figure 2-2. Lines starting with 'x1' refer to apply runs in ALOGIT. Lines starting with 'DoCmd' refer to Access macros.

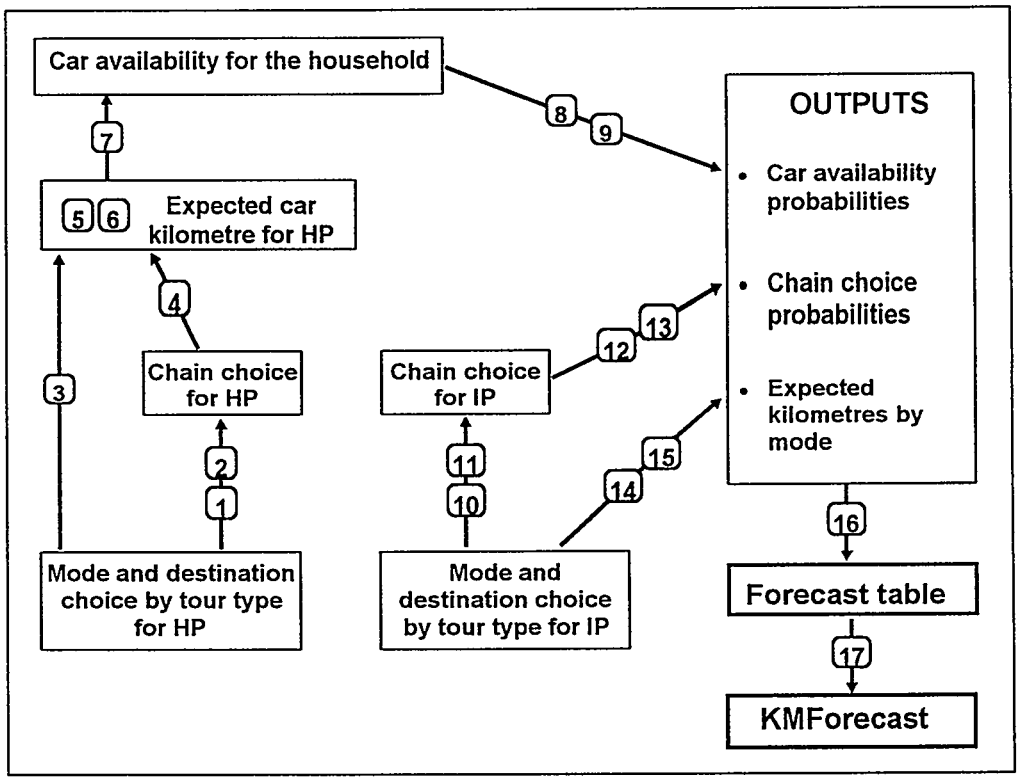
Figure 2-2 Main command sequence

```
1.  x1 = Shell("C:\25127\FORECAST\LsTopHP.BAT", 1)
2.  DoCmd.RunMacro "F_DoTopYesnoHP"
3.  x1 = Shell("C:\25127\FORECAST\KmCarHP.bat", 1)
4.  x1 = Shell("C:\25127\FORECAST\PrTopHP.BAT", 1)
5.  DoCmd.RunMacro "F_ImportKmTranspMiddel"
6.  DoCmd.RunMacro "F_ImportTurtypeProbsCar012"
7.  DoCmd.RunMacro "F_ForecastDelModelKmToYesNoDat"
8.  x1 = Shell("C:\25127\FORECAST\PROBCAR.BAT", 1)
9.  DoCmd.RunMacro "F_Importcarprob"
10. x1 = Shell("C:\25127\FORECAST\LsTopIP.BAT", 1)
11. DoCmd.RunMacro "F_DoTopYesnoIP"
12. x1 = Shell("C:\25127\FORECAST\PrTopIP.BAT", 1)
13. DoCmd.RunMacro "F_ImportTurtypeProbsCar012"
14. x1 = Shell("C:\25127\FORECAST\KmCarIP.bat", 1)
15. DoCmd.RunMacro "F_ImportKmTranspMiddel"
16. DoCmd.RunMacro "F_Forecasttable"
17. DoCmd.RunMacro " F_CalculateExpectedKM"
```

These Access commands must be executed in the shown order. Since Access must run programs in a shell, some code is necessary to ensure that the sequence shown is observed. The code is shown in Annex 1.

Figure 2-3 shows the flow of calculations.

Figure 2-3 Calculation flow



The numbers refer to the commands in Figure 2-2. The function of the commands is described briefly below.

1. Executes an ALOGIT program calculating logsums for the head of each of the households
2. Imports the calculated logsums to Access and updates data files to use in the TOP model
3. Executes an ALOGIT program calculating expected kilometre by car given chain choice for the head of each of the households
4. Executes an ALOGIT program calculating expected probability of choosing each of the 11 chain types
5. Imports expected kilometres by car calculated in submodels conditional on chain type.
6. Imports the calculated probabilities for each chain type from the TOP model.
7. Calculates expected kilometre by car as the conditional expected kilometre weighted by the probabilities from the TOP model and passes these data on to the data files for the CAR model.

8. Executes an ALOGIT program calculating the probability for no car, one car and two cars respectively available to the households.
9. Imports the calculated car availability probabilities into the result database.
10. Executes an ALOGIT program calculating logsums for the IP of each of the households
11. Imports the calculated IP logsums to Access and updates data files to use in the TOP model
12. Executes an ALOGIT program calculating expected probability of IP choosing each of the 11 chain types
13. Imports the calculated chain type probabilities into the result database.
14. Executes an ALOGIT program calculating expected kilometre by five transportation modes given chain choice and car availability for the IP of each of the households
15. Imports expected kilometres calculated in submodels conditional on chain type and car availability.
16. Collects the results in the table "Forecast" in the database.
17. Collects the results in the table "KMForecast" in the database and calculates expected distance with the five modes: Slow, Car, Car passenger, Bus and Train.

Short / long run

TDM can be used at two levels:

- The short term level includes effects from destination (distance) choice and transportation mode choice, but excludes effects from car availability and chain choice patterns.
- The long term level includes the total effect on distance, car fleet, chain choice, mode and destination choice

2.2 Basic scenarios

Presently, three basic scenarios have been elaborated for PETRA. More may follow if required.

- Base: 1995
- Forecast for 2020, only including demographic changes (age, sex, and, accordingly, licence holding)
- Forecast for 2020, including both demographic changes and income increase of 2 % per year from 1995 to 2020.

The scenarios consist of sets of data files with data for the forecast situation for all individuals in the sample. Each basic scenario set consist of 3 data files for IP and 3 data files for the Head of the household - 6 files in total. For both IP and HP there is a file for each of the three choice levels in the TDM: car availability, chain choice and mode/destination.

- LogsIP1.dat
- LogsIP2.dat
- LogsIP3.dat
- LogsHP1.dat
- LogsHP2.dat
- LogsHP3.dat

These data files have been created for the three basic scenarios mentioned above. To save space the files are compressed into zip files named after the basic scenario.

- Logs1995.zip
- Logs2020.zip
- Logs2020IG.zip

Other scenarios can be designed, but this would require more comprehensive changes in the databases.

To apply a specific scenario requires that the relevant 6 data files are uncompressed and copied to the working directory (C:\25127\forecast).

2.3 File structure

Working directory

The working directory is C:\25127\forecast, from which the TDM *must* run. All data and program files must be placed in this directory and subdirectories.

Figure 2-4 shows the file structure. Shaded boxes indicate directories.

Figure 2-4 Model structure - files

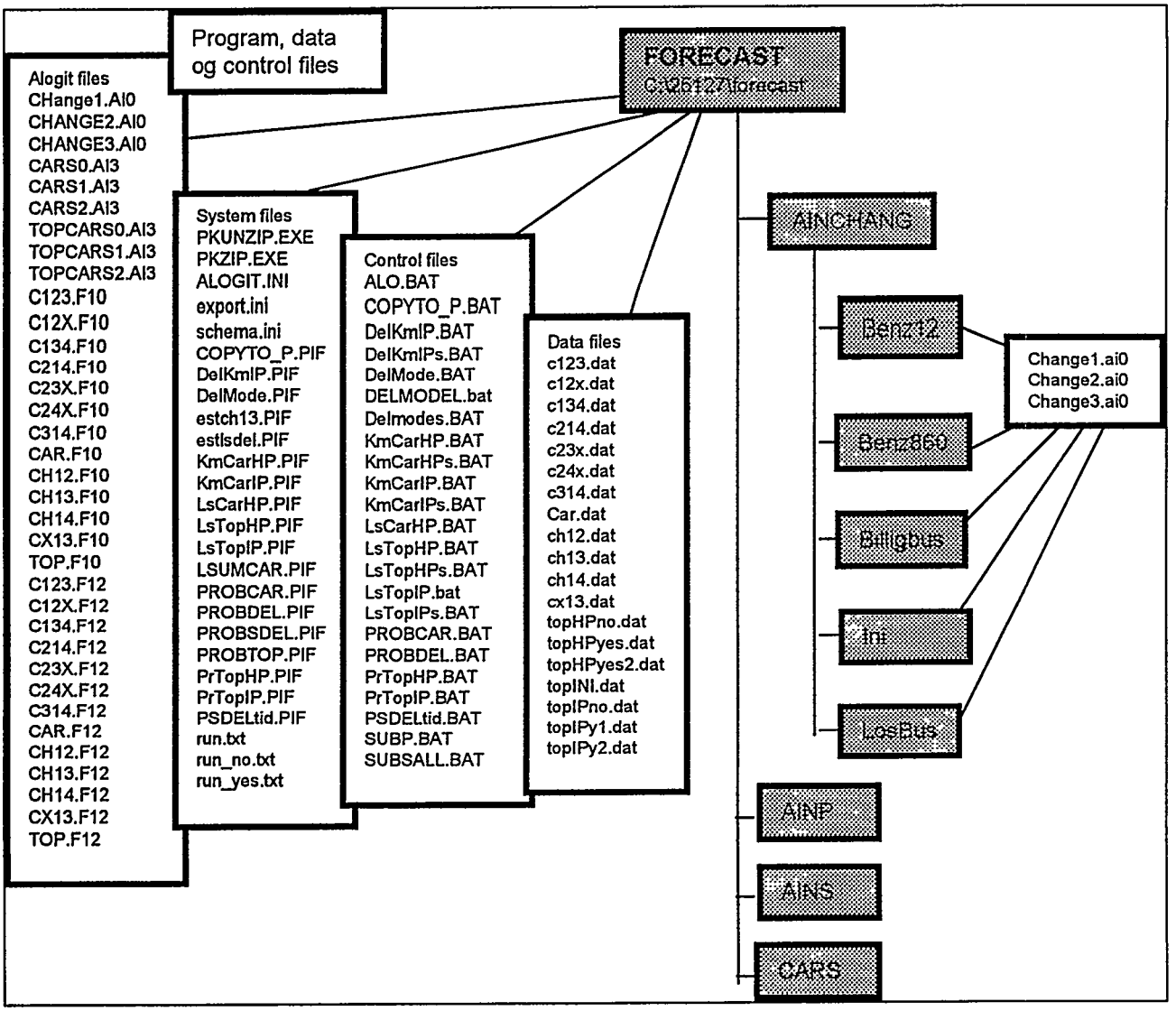


Table 2.1 describes the file types in the FORECAST directory.

Table 2.1 File types in the FORECAST directory

Extension	Description
AI0	ALOGIT program file component
AI3	ALOGIT program file component
BAT	Control file for execution of ALOGIT apply
DAT	Data file for execution of ALOGIT apply
EXE	DOS programs
F10	ALOGIT model information file
F12	ALOGIT parameter information file from estimations
INI	Ini file
PIF	Windows file controlling the DOS shell
TXT	Text file controlling conflicts between Access and ALOGIT

The most important files are the *.ai0 files. In these files changes to the chosen basic scenario are defined and transferred to the models according to the utility functions in the ALOGIT models. The program code uses these files to create complete apply files (*.AIN) applied in the ALOGIT runs. An example of these files is given in Figure 3-1.

Most of the ALOGIT apply file is placed in the subdirectories and when used these parts are copied to the working directory. Here the program combines these parts with more general parts of the ALOGIT apply files and controls the execution of these.

A number of transport policy measures have been analysed previously.

- Tax on petrol price;
- Price change on bus tickets;
- Reductions in transportation tax deductions (befordringsfradrag);
- Change in the level of service for busses.

Modified *.ai0 files for each analysis are placed in subdirectories to the AINCHANG directory.

A new analysis regarding these measures with a different level can be implemented by changing the three main part apply files "Change1.ai0", "Change2.ai0" and "Change3.ai0" in the relevant subdirectory.

3 Running PETRA

PETRA is used by translating the policy measures to analyse changes in the *.ai0 ALOGIT control files. It is recommended that a new subdirectory is created for each analysed policy change in such a way that e.g. petrol prices rising up to 10% would imply a new subdirectory named "PetrolUP10%" containing three files (Change1.ai0, Change2.ai0 and Change3.ai0).

What the ALOGIT *ai0 files should look like depends on the policy to analyse and how this variable enters in the utility function in the submodels of PETRA. Formulating new files not covered by the examples requires some knowledge about ALOGIT programming and how the submodels are specified.

Small changes for instance analysing the same policy variable as one of the examples is easy, since this only requires that the amount of change in the policy variable is changed.

When the necessary *.ai0 ALOGIT files are created, these three files are copied to the main working directory (C:\25127\FORECAST) and the model is started in the Access database (copy one of the basis scenario databases to a new name and execute the model from this new copied database).

The complete model is run by opening the form "TDM" in the copied database.

After the model has been run, the resulting predictions can be compared with the predictions in the basis scenario.

The PETRA model is applied in six steps.

1. Unzip the directory "Forecast" with content to the working directory "C:\2527\Forecast";

2. Unzip the 6 relevant logsum data files to the working directory;

3. Make the changes to the files (*.ai0) in the subdirectory. (It is advisable to make a copy of the apply files in a new subdirectory with the new policy measure to keep track of which measures has been carried out)

4. Copy these three policy files to the working directory;

5. Unzip the relevant scenario database;

6. Open the scenario database (create a copy of the base scenario database) and run the model by opening the form "TDM"

Three types of models make three different control files necessary to apply the model. This is the reason why there are three *.ai0 ALOGIT control files. Type one and three are relatively simple, while type two is more complicated since it refers to mode and destination choice for triangular tours.

The following example shows the changes necessary to introduce a reduction in the price of bus tickets.

*Figure 3-1 Example of *.ai0 file*

Type 1 and 3

```
END
EXCLUDE=0
busch=0.75
buscost1 = buscost1*busch
buscost2 = buscost2*busch
buscost3 = buscost3*busch
buscost4 = buscost4*busch
buscost5 = buscost5*busch
buscost6 = buscost6*busch
buscost7 = buscost7*busch
buscost8 = buscost8*busch
buscost9 = buscost9*busch
output= dummy
```

Type 2

```
END
EXCLUDE=0

busch=0.75

d050=d050*busch
d250=d250*busch
d450=d450*busch
d650=d650*busch
d850=d850*busch
d1050=d1050*busch
d1250=d1250*busch
d1450=d1450*busch
d1650=d1650*busch


d80=d80*busch
d110=d110*busch
d140=d140*busch
d170=d170*busch
d280=d280*busch
d310=d310*busch
d340=d340*busch
d370=d370*busch
d480=d480*busch
d510=d510*busch
d540=d540*busch
d570=d570*busch
d680=d680*busch
d710=d710*busch
d740=d740*busch
d770=d770*busch
d880=d880*busch
d910=d910*busch
d940=d940*busch
d970=d970*busch
d1080=d1080*busch
d1110=d1110*busch
d1140=d1140*busch
```

```
d1170=d1170*busch
d1280=d1280*busch
d1310=d1310*busch
d1340=d1340*busch
d1370=d1370*busch
d1480=d1480*busch
d1510=d1510*busch
d1540=d1540*busch
d1570=d1570*busch
d1680=d1680*busch
d1710=d1710*busch
d1740=d1740*busch
d1770=d1770*busch
```

As can be seen, the bus cost is reduced to 75% of the initial level in this example. Reduction by 10% to 90% would only require that 0.75 is replaced by 0.90 in the three files.

4 Results

The result from the TDM is a table containing expected travel demand which can be combined with individual socio-economic data for the interviewed persons[1]

The result table, named "KMforecast", from the comprehensive model is created in the database at the end of the procedure.

The result from the TDM consists of

- Expected kilometres
- Probabilities for chain types
- Probabilities for car availability

The expected kilometres are distributed on five modes.

- Slow
- Car
- Car passenger
- Bus
- Train

The expected kilometre is calculated conditional on

- No car
- One car and
- Two cars (only couples)

and conditional on chain type:

[1] To secure anonymity, all data which could identify single persons has been left out.

- 12: Home, Work, Home
- 12x: Home, Work, Home and thereafter another tour
- 13: Home, Shopping, Home
- 14: Home, Leisure, Home
- 214: Home, Work, Home, Leisure, Home
- 314: Home, Shopping, Home, Leisure, Home
- cx13: An unspecified main tour ending with Home, Shopping, Home
- 123: Home, Work, Shopping, Home
- 23x: Home, Work, Shopping, Home and a minor tour
- 24x: Home, Work, Leisure and a minor tour
- 134: Home, Shopping, Leisure, Home

The expected kilometre for a given mode, m, and a given individual, i, is calculated according to the formula:

$$K_{i,m} = \sum_{t,u} E(Km_{i,m} | Chain = t \ And \ Car = u) * P(chain = t) * P(car = u)$$

Examples of output generated with queries in the database are shown in the following tables.

Table 4.2 Predicted travel distance

Avg. Km per day	Slow	Car	Car passenger	Bus	Train	I alt
Base, 1995	2,03	15,55	4,40	1,84	0,39	24,20

Query: "F_AvgKmModel"

Table 4.3 Predicted number of tours segmented after purpose combinations

Number of tours in sample	95 ini
Home, work, home	2.226
Home, work, home, shop, home	759
Home, work, home, leisure, home	1.097
Home, work, shop, home	642
Home, work, shop, home, shop, home	193
Home, work, shop, home, leisure, home	269
Home, work, leisure, home	328
Home, work, leisure, home, shop, home	33
Home, work, leisure, home, leisure, home	73
Home, shop, home	2.147
Home, leisure, home	2.716
Home, shop, shop, home	449
Stay home	2.620

Query: "F_SumChainChoice"

Table 4.4 Predicted car availability per household

	1 car	2 cars
Year 1995	62,4%	14,1%

Query: "F_AvgCarAvailability"

The model predictions can be analysed with respect to socio-economic characteristics of the individuals by linking the result table to the IPvariable tables. This linking allows for analysis with respect to the following socio-economic characteristics of the individuals:

- Six
- Age
- Net income (25.000 intervals)
- Urbanisation
- Family type
- Employment Status

The forecast tables can be linked to the IPvariable table by means of the ID key which is present in both tables.

5 References

Alogit User's guide, Version 3.8 August 1995, Hague Consulting Group, the Hague, Netherlands.

Ben-Akiva, Moshe and Steven R. Lerman: Discrete Choice Analysis - Theory and Application to Travel Demand, MIT Press, London 1985.

PETRA Working Paper no. 1, Modelling Concept

PETRA Working Paper no. 2, Preliminary Analysis of Transportation Surveys

PETRA Working Paper no. 3, Weights, COWI June 1997.

PETRA Working Paper no. 4, The Licence Holding Model, COWI June 1997.

PETRA Working Paper No. 5, The Cohort Model, COWI, June 1997

PRTRA Working Paper no. 6, Car Fleet Model

PETRA Working Paper No. 7, The Travel Demand Module, COWI, June 1997.

PETRA Working Paper No. 8, Data, COWI June 1997.

PETRA Synthesis Report - The Forecast Model, COWI, June 1997.

6 Annex:

6.1 Annex 1: Code for main PETRA model

```
Public Function RunTDM()
Call AlogitRunning
If ARUN = 1 Then GoTo slut
If READY = 0 Then GoTo slut
Select Case STEP
   Case 1
      STEP = 2
      x1 = Shell("C:\25127\FORECAST\LsTopHP.BAT", 1)
      GoTo slut
   Case 2
      STEP = 3
      READY = 0
      DoCmd.RunMacro "F_DoTopYesnoHP"
      Call AccessFinished
      GoTo slut
   Case 3
      STEP = 4
      x1 = Shell("C:\25127\FORECAST\KmCarHP.bat", 1)
      GoTo slut
   Case 4
      STEP = 5
      x1 = Shell("C:\25127\FORECAST\PrTopHP.BAT", 1)
      GoTo slut
   Case 5
      STEP = 6
      READY = 0
      DoCmd.RunMacro "F_ImportKmTranspMiddel"
      Call AccessFinished
      GoTo slut
   Case 6
      STEP = 7
      READY = 0
      DoCmd.RunMacro "F_ImportTurtypeProbsCar012"
      Call AccessFinished
      GoTo slut
   Case 7
      STEP = 8
      READY = 0
      DoCmd.RunMacro "F_ForecastDelModelKmToYesNoDat"
      Call AccessFinished
```

```
    GoTo slut
  Case 8
    STEP = 9
    x1 = Shell("C:\25127\FORECAST\PROBCAR.BAT", 1)
    GoTo slut
  Case 9
    STEP = 10
    READY = 0
    DoCmd.RunMacro "F_Importcarprob"
    Call AccessFinished
    GoTo slut
  Case 10
    STEP = 11
    x1 = Shell("C:\25127\FORECAST\LsTopIP.BAT", 1)
    GoTo slut
  Case 11
    STEP = 12
    READY = 0
    DoCmd.RunMacro "F_DoTopYesnoIP"
    Call AccessFinished
    GoTo slut
  Case 12
    STEP = 13
    x1 = Shell("C:\25127\FORECAST\PrTopIP.BAT", 1)
    GoTo slut
  Case 13
    STEP = 14
    READY = 0
    DoCmd.RunMacro "F_ImportTurtypeProbsCar012"
    Call AccessFinished
    GoTo slut
  Case 14
    STEP = 15
    x1 = Shell("C:\25127\FORECAST\KmCarIP.bat", 1)
    GoTo slut
  Case 15
    STEP = 16
    READY = 0
    DoCmd.RunMacro "F_ImportKmTranspMiddel"
    Call AccessFinished
    GoTo slut
  Case 16
    STEP = 17
    READY = 0
    DoCmd.RunMacro "F_Forecasttable"
    Call AccessFinished
    GoTo slut
  Case Else
    Msg = "Så slap vi igennem"
End Select
slut:
Exit Function
End Function
```

6.2 Annex 2: Sub model formulation, (home, work home)

Work Tours (chain 12)

DATA 900

stats 1 0 0

print 4

- no slow if more than 5 km
nonav 1,1841 2,1842 3,1843 4,1844 5,1845 6,1846 7,1847 8,1848 9,1849

- no CYKEL if more than 40 km
nonav 10,1825 11,1826 12,1827 13,1828 14,1829 15,1830 16,1831 17,1832 18,1833

- no CAR if NO LICENCE or younger than 18
nonav 19,1802 20,1802 21,1802 22,1802 23,1802 24,1802 25,1802 26,1802 27,1802

- NO TRAIN IF BORNHOLM OR SHORT DISTANCE
nonav 46,1812 47,1812 48,1812 49,1812 50,1812 51,1812 52,1812 53,1812 54,1812
nonav 46,1803 47,1804 48,1805 49,1806 50,1807 51,1808 52,1809 53,1810 54,1811

- Ikke bus, hvis gaabus=5
nonav 37,1813 38,1813 39,1813 40,1813 41,1813 42,1813 43,1813 44,1813 45,1813

- Ikke bus, hvis afstanden er for lille
nonav 37,1815 38,1816 39,1817 40,1818 41,1819 42,1820 43,1821 44,1822 45,1823

STOP 20
END

001 Slow
002 Car
003 CarPas
004 Bus
005 Train
101 Kroner
102 Income t 0
103 KmInc t 0
203 KmInLF
111 WalkASC
112 Time
113 Enlig1
313 Enlig1F
114 SelvstM
116 HT
121 BicycASC
126 CarPrLic
326 CPLHjmF
127 Stud t 0
427 StudF
128 NoCarC
129 NoCarCP
131 CPASC
231 CPASCF
331 CPASCHM
431 CPASCLM
132 CPtime
133 XCPTime
135 CPgl
136 CarLicCP t 0
137 CPpartKK t 0
141 BusASC
241 BsASCYM
341 BsASCGF
441 BsASCYF
142 PBusLsDs t 0
143 SBusLsDs
144 PBusLsBo t 0
145 Enlig
146 HT
147 BusTime
148 BusGL
248 BunempF
149 GaaBus t 0
151 TrnASC
152 PTogtid
154 PTrnLsDs
153 PTrnLsBo t 0
156 GaaSta
159 GaaBus t 0
170 Beskft

172 PSelvbsk
173 PArbBsk
174 PStudH1
175 PStudH3
176 GymStudF
177 OfadmUd t 0
178 OFLArbL
179 PArbH1
180 SStudBsk
181 SArLoBsk
182 SSelvbsk
183 SArbeBsk
184 SStudH1
185 SStudH3
186 SStudGym
187 SSelvH1
188 SSelvHote
189 SArbH1
190 Befolkn
138 NoCar
200 PWorkArb
201 SWorkArb
210 OFFSHOs5
211 OFst1t3
212 H3exclHF
213 H1ArblsF
214 H1Fkt
215 PELandMM
216 PEFktLF
217 GymnStud
218 OFStud
219 OFFkt
220 ByAM356
221 ByAF356
222 ByAM4
223 H1M356
224 H1F346
225 H3alle
226 HRalle
227 PEM5
228 PEF8
229 Transp
230 Folkesk
301 OmkPrInc t 0
122 SelvstBC
125 ARBTID t 0

income=d38-5.65

M=ifeq(d14,1)
F=1-M

H=ifge(income,0)
L=1-H
Y=IFLE(d2,50)
licence=ifeq(d10,1)
GL=ifge(d2,50)
SelvstM=Ifin(d16:11,12)*m
HjemmeF=Ifin(d16:3,9)*f
bt=51
togkmt=57
by=17
CarPrLic=ifin(d9:1,3)*d9/max(d11,1)
CarPrLiF=CarPrLic*F
nolicenc=ifeq(d11,0)
HT=ifin(d27:33,34)
enlig=ifne(d3,2)/D38
enlig1=ifeq(d3,1)*F
ST1=ifin(d16:1,2)
ST2=ifeq(d16,3)+ifeq(d16,6)
ST3=ifin(d16:4,5)+ifeq(d16,7)+ifeq(d16,9)
ST4=ifeq(d16,11)
ST5=ifin(d16:12,13)
ST6=ifeq(d16,14)+ifin(d16:20,21)
ST7=ifin(d16:31,32)
ST8=ifeq(d16,30)
ST9=ifeq(d16:8,98,99)

Stud=st1
NotArb=st2+st3+st4+st5
Selvst=st4+st5
Arb=st6+st7+st8
ArboLav=st6+st7
StEjArb=st1+st2+st3+st4+st5

d1801=ifin(d9:1,3)
d1802=licence and ifge(d2,18)
d1803=ifge(d44,5)
d1804=ifge(d144,5)
d1805=ifge(d244,5)
d1806=ifge(d344,5)
d1807=ifge(d444,5)
d1808=ifge(d544,5)
d1809=ifge(d644,5)
d1810=ifge(d744,5)
d1811=ifge(d844,5)
d1812=ifne(d27,35)
d1813=iflt(d12,5)
d1814=recode(d12:0,0,0,3,6)
d1815= ifge(d44,d1814)
d1816= ifge(d144,d1814)
d1817= ifge(d244,d1814)

d1818= ifge(d344,d1814)
d1819= ifge(d444,d1814)
d1820= ifge(d544,d1814)
d1821= ifge(d644,d1814)
d1822= ifge(d744,d1814)
d1823= ifge(d844,d1814)
D1824 = ifge(d2,18)
d1825=ifle(d44,50)
d1826=ifle(d144,50)
d1827=ifle(d244,50)
d1828=ifle(d344,50)
d1829=ifle(d444,50)
d1830=ifle(d544,50)
d1831=ifle(d644,50)
d1832=ifle(d744,50)
d1833=ifle(d844,50)

d1841=ifle(d44,10)
d1842=ifle(d144,10)
d1843=ifle(d244,10)
d1844=ifle(d344,10)
d1845=ifle(d444,10)
d1846=ifle(d544,10)
d1847=ifle(d644,10)
d1848=ifle(d744,10)
d1849=ifle(d844,10)

Bustid1 = 60*d046/d048+10
Bustid2 = 60*d146/d148+10
Bustid3 = 60*d246/d248+10
Bustid4 = 60*d346/d348+10
Bustid5 = 60*d446/d448+10
Bustid6 = 60*d546/d548+10
Bustid7 = 60*d646/d648+10
Bustid8 = 60*d746/d748+10
Bustid9 = 60*d846/d848+10

Togtid1 = d052
Togtid2 = d152
Togtid3 = d252
Togtid4 = d352
Togtid5 = d452
Togtid6 = d552
Togtid7 = d652
Togtid8 = d752
Togtid9 = d852

- beregner valg Tog/bus PT=1 tog, PT=0 bus
PTog1 = IFgt(bustid1,togtid1)
PTog2 = IFgt(bustid2,togtid2)
PTog3 = IFgt(bustid3,togtid3)
PTog4 = IFgt(bustid4,togtid4)
PTog5 = IFgt(bustid5,togtid5)
PTog6 = IFgt(bustid6,togtid6)
PTog7 = IFgt(bustid7,togtid7)
PTog8 = IFgt(bustid8,togtid8)
PTog9 = IFgt(bustid9,togtid9)

Buskm1 = d046
Buskm2 = d146
Buskm3 = d246
Buskm4 = d346
Buskm5 = d446
Buskm6 = d546
Buskm7 = d646
Buskm8 = d746
Buskm9 = d846

Togkm1 = d042
Togkm2 = d142
Togkm3 = d242
Togkm4 = d342
Togkm5 = d442
Togkm6 = d542
Togkm7 = d642
Togkm8 = d742
Togkm9 = d842

BusLoS1 = d31

TogLoS11 = d054
TogLoS12 = d154
TogLoS13 = d254
TogLoS14 = d354
TogLoS15 = d454
TogLoS16 = d554
TogLoS17 = d654
TogLoS18 = d754
TogLoS19 = d854

```
togLoS2 =  d34

BusCost1 = d050
BusCost2 = d150
BusCost3 = d250
BusCost4 = d350
BusCost5 = d450
BusCost6 = d550
BusCost7 = d650
BusCost8 = d750
BusCost9 = d850

TogCost1 = d051
TogCost2 = d151
TogCost3 = d251
TogCost4 = d351
TogCost5 = d451
TogCost6 = d551
TogCost7 = d651
TogCost8 = d751
TogCost9 = d851

PslowTid1 = (ifle(d044,1)*d044*12 + ifGT(d044,1)*d044*4)
PslowTid2 = (ifle(d144,1)*d144*12 + ifGT(d144,1)*d144*4)
PslowTid3 = (ifle(d244,1)*d244*12 + ifGT(d244,1)*d244*4)
PslowTid4 = (ifle(d344,1)*d344*12 + ifGT(d344,1)*d344*4)
PslowTid5 = (ifle(d444,1)*d444*12 + ifGT(d444,1)*d444*4)
PslowTid6 = (ifle(d544,1)*d544*12 + ifGT(d544,1)*d544*4)
PslowTid7 = (ifle(d644,1)*d644*12 + ifGT(d644,1)*d644*4)
PslowTid8 = (ifle(d744,1)*d744*12 + ifGT(d744,1)*d744*4)
PslowTid9 = (ifle(d844,1)*d844*12 + ifGT(d844,1)*d844*4)

PS1 = -log(d041)
PS2 = -log(d141)
PS3 = -log(d241)
PS4 = -log(d341)
PS5 = -log(d441)
PS6 = -log(d541)
PS7 = -log(d641)
PS8 = -log(d741)
PS9 = -log(d841)

IBEF1 = log(d055)
IBEF2 = log(d155)
IBEF3 = log(d255)
IBEF4 = log(d355)
IBEF5 = log(d455)
IBEF6 = log(d555)
IBEF7 = log(d655)
IBEF8 = log(d755)
IBEF9 = log(d855)

SUM1 = d047 +d056 +d057 +d058 +d059 +d060 +d061 +d062
+d063 +d064 +d065 +d066
+d067 +d068 +d069 +d070
SUM2 = d147 +d156 +d157 +d158 +d159 +d160 +d161 +d162
+d163 +d164 +d165 +d166
+d167 +d168 +d169 +d170
SUM3 = d247 +d256 +d257 +d258 +d259 +d260 +d261 +d262
+d263 +d264 +d265 +d266
+d267 +d268 +d269 +d270
SUM4 = d347 +d356 +d357 +d358 +d359 +d360 +d361 +d362
+d363 +d364 +d365 +d366
+d367 +d368 +d369 +d370
SUM5 = d447 +d456 +d457 +d458 +d459 +d460 +d461 +d462
+d463 +d464 +d465 +d466
+d467 +d468 +d469 +d470
SUM6 = d547 +d556 +d557 +d558 +d559 +d560 +d561 +d562
+d563 +d564 +d565 +d566
+d567 +d568 +d569 +d570
SUM7 = d647 +d656 +d657 +d658 +d659 +d660 +d661 +d662
+d663 +d664 +d665 +d666
+d667 +d668 +d669 +d670
SUM8 = d747 +d756 +d757 +d758 +d759 +d760 +d761 +d762
+d763 +d764 +d765 +d766
+d767 +d768 +d769 +d770
SUM9 = d847 +d856 +d857 +d858 +d859 +d860 +d861 +d862
+d863 +d864 +d865 +d866
+d867 +d868 +d869 +d870

Lsum1 = log(sum1)
Lsum2 = log(sum2)
Lsum3 = log(sum3)
Lsum4 = log(sum4)
Lsum5 = log(sum5)
Lsum6 = log(sum6)
Lsum7 = log(sum7)
Lsum8 = log(sum8)
Lsum9 = log(sum9)

H11 = max(d059,1)
H12 = max(d159,1)
H13 = max(d259,1)
H14 = max(d359,1)
H15 = max(d459,1)
H16 = max(d559,1)
H17 = max(d659,1)
H18 = max(d759,1)
H19 = max(d859,1)
```

IH11 = log(H11/(sum1-H11))
IH12 = log(H12/(sum2-H12))
IH13 = log(H13/(sum3-H13))
IH14 = log(H14/(sum4-H14))
IH15 = log(H15/(sum5-H15))
IH16 = log(H16/(sum6-H16))
IH17 = log(H17/(sum7-H17))
IH18 = log(H18/(sum8-H18))
IH19 = log(H19/(sum9-H19))

H31 = max(d061,1)
H32 = max(d161,1)
H33 = max(d261,1)
H34 = max(d361,1)
H35 = max(d461,1)
H36 = max(d561,1)
H37 = max(d661,1)
H38 = max(d761,1)
H39 = max(d861,1)

IH31 = log(H31/(sum1-H31))
IH32 = log(H32/(sum2-H32))
IH33 = log(H33/(sum3-H33))
IH34 = log(H34/(sum4-H34))
IH35 = log(H35/(sum5-H35))
IH36 = log(H36/(sum6-H36))
IH37 = log(H37/(sum7-H37))
IH38 = log(H38/(sum8-H38))
IH39 = log(H39/(sum9-H39))

HR1 = max(d062,1)
HR2 = max(d162,1)
HR3 = max(d262,1)
HR4 = max(d362,1)
HR5 = max(d462,1)
HR6 = max(d562,1)
HR7 = max(d662,1)
HR8 = max(d762,1)
HR9 = max(d862,1)

IHR1 = log(HR1/(sum1-HR1))
IHR2 = log(HR2/(sum2-HR2))
IHR3 = log(HR3/(sum3-HR3))
IHR4 = log(HR4/(sum4-HR4))
IHR5 = log(HR5/(sum5-HR5))
IHR6 = log(HR6/(sum6-HR6))
IHR7 = log(HR7/(sum7-HR7))
IHR8 = log(HR8/(sum8-HR8))
IHR9 = log(HR9/(sum9-HR9))

GY1 = max(d069,1)
GY2 = max(d169,1)
GY3 = max(d269,1)
GY4 = max(d369,1)
GY5 = max(d469,1)
GY6 = max(d569,1)
GY7 = max(d669,1)
GY8 = max(d769,1)
GY9 = max(d869,1)

IGY1 = log(GY1/(sum1-GY1))
IGY2 = log(GY2/(sum2-GY2))
IGY3 = log(GY3/(sum3-GY3))
IGY4 = log(GY4/(sum4-GY4))
IGY5 = log(GY5/(sum5-GY5))
IGY6 = log(GY6/(sum6-GY6))
IGY7 = log(GY7/(sum7-GY7))
IGY8 = log(GY8/(sum8-GY8))
IGY9 = log(GY9/(sum9-GY9))

OF1 = max(d066,1)
OF2 = max(d166,1)
OF3 = max(d266,1)
OF4 = max(d366,1)
OF5 = max(d466,1)
OF6 = max(d566,1)
OF7 = max(d666,1)
OF8 = max(d766,1)
OF9 = max(d866,1)

IOF1 = log(OF1/(sum1-OF1))
IOF2 = log(OF2/(sum2-OF2))
IOF3 = log(OF3/(sum3-OF3))
IOF4 = log(OF4/(sum4-OF4))
IOF5 = log(OF5/(sum5-OF5))
IOF6 = log(OF6/(sum6-OF6))
IOF7 = log(OF7/(sum7-OF7))
IOF8 = log(OF8/(sum8-OF8))
IOF9 = log(OF9/(sum9-OF9))

TR1 = max(d063,1)
TR2 = max(d163,1)
TR3 = max(d263,1)
TR4 = max(d363,1)
TR5 = max(d463,1)
TR6 = max(d563,1)
TR7 = max(d663,1)
TR8 = max(d763,1)
TR9 = max(d863,1)

ITR1 = log(TR1/(sum1-TR1))
ITR2 = log(TR2/(sum2-TR2))
ITR3 = log(TR3/(sum3-TR3))
ITR4 = log(TR4/(sum4-TR4))
ITR5 = log(TR5/(sum5-TR5))
ITR6 = log(TR6/(sum6-TR6))
ITR7 = log(TR7/(sum7-TR7))
ITR8 = log(TR8/(sum8-TR8))
ITR9 = log(TR9/(sum9-TR9))

PE1 = max(d064,1)
PE2 = max(d164,1)
PE3 = max(d264,1)
PE4 = max(d364,1)
PE5 = max(d464,1)
PE6 = max(d564,1)
PE7 = max(d664,1)
PE8 = max(d764,1)
PE9 = max(d864,1)

IPE1 = log(PE1/(sum1-PE1))
IPE2 = log(PE2/(sum2-PE2))
IPE3 = log(PE3/(sum3-PE3))
IPE4 = log(PE4/(sum4-PE4))
IPE5 = log(PE5/(sum5-PE5))
IPE6 = log(PE6/(sum6-PE6))
IPE7 = log(PE7/(sum7-PE7))
IPE8 = log(PE8/(sum8-PE8))
IPE9 = log(PE9/(sum9-PE9))

BY1 = max(d058,1)
BY2 = max(d158,1)
BY3 = max(d258,1)
BY4 = max(d358,1)
BY5 = max(d458,1)
BY6 = max(d558,1)
BY7 = max(d658,1)
BY8 = max(d758,1)
BY9 = max(d858,1)

IBY1 = log(BY1/(sum1-BY1))
IBY2 = log(BY2/(sum2-BY2))
IBY3 = log(BY3/(sum3-BY3))
IBY4 = log(BY4/(sum4-BY4))
IBY5 = log(BY5/(sum5-BY5))
IBY6 = log(BY6/(sum6-BY6))
IBY7 = log(BY7/(sum7-BY7))
IBY8 = log(BY8/(sum8-BY8))
IBY9 = log(BY9/(sum9-BY9))

HO1 = max(d068,1)
HO2 = max(d168,1)
HO3 = max(d268,1)
HO4 = max(d368,1)
HO5 = max(d468,1)
HO6 = max(d568,1)
HO7 = max(d668,1)
HO8 = max(d768,1)
HO9 = max(d868,1)

IHO1 = log(HO1/(sum1-HO1))
IHO2 = log(HO2/(sum2-HO2))
IHO3 = log(HO3/(sum3-HO3))
IHO4 = log(HO4/(sum4-HO4))
IHO5 = log(HO5/(sum5-HO5))
IHO6 = log(HO6/(sum6-HO6))
IHO7 = log(HO7/(sum7-HO7))
IHO8 = log(HO8/(sum8-HO8))
IHO9 = log(HO9/(sum9-HO9))

FS1 = max(d067,1)
FS2 = max(d167,1)
FS3 = max(d267,1)
FS4 = max(d367,1)
FS5 = max(d467,1)
FS6 = max(d567,1)
FS7 = max(d667,1)
FS8 = max(d767,1)
FS9 = max(d867,1)

IFS1 = log(FS1/(sum1-FS1))
IFS2 = log(FS2/(sum2-FS2))
IFS3 = log(FS3/(sum3-FS3))
IFS4 = log(FS4/(sum4-FS4))
IFS5 = log(FS5/(sum5-FS5))
IFS6 = log(FS6/(sum6-FS6))
IFS7 = log(FS7/(sum7-FS7))
IFS8 = log(FS8/(sum8-FS8))
IFS9 = log(FS9/(sum9-FS9))

- befordringsfradrag
fradrag1=-0.46*(min(max(0, d44-12),38)*1.21 + max(0, d44-50)*0.605)

```
fradrag2=-0.46*( min(max(0,d144-12),38)*1.21 + max(0,d144-
50)*0.605 )
fradrag3=-0.46*( min(max(0,d244-12),38)*1.21 + max(0,d244-
50)*0.605 )
fradrag4=-0.46*( min(max(0,d344-12),38)*1.21 + max(0,d344-
50)*0.605 )
fradrag5=-0.46*( min(max(0,d444-12),38)*1.21 + max(0,d444-
50)*0.605 )
fradrag6=-0.46*( min(max(0,d544-12),38)*1.21 + max(0,d544-
50)*0.605 )
fradrag7=-0.46*( min(max(0,d644-12),38)*1.21 + max(0,d644-
50)*0.605 )
fradrag8=-0.46*( min(max(0,d744-12),38)*1.21 + max(0,d744-
50)*0.605 )
fradrag9=-0.46*( min(max(0,d844-12),38)*1.21 + max(0,d844-
50)*0.605 )
- traekprocent, kmgraenser/2, fradragssatser

- marginal km omkostning (gennemsnitspris paa benzin 1995)
kmomk=5.92/13.3



mode=recode(d26,1,2,3,4,5,5,6,7,7)
choseAlt=ifeq(d43,1)*1+ifeq(d143,1)*2+ifeq(d243,1)*3+ifeq(d343,
1)*4
    +ifeq(d443,1)*5+ifeq(d543,1)*6
    +ifeq(d643,1)*7 +ifeq(d743,1)*8 +ifeq(d843,1)*9

exclude = (1-ifin(mode:1,6)) or ifeq(chosealt,0)

WEIGHT=d37
ID=d1
CHOICE=(mode-1)*9 + chosealt


-slow
U1= p111 +p112*(12*d044*2) +p313*enlig1  +p170*lsum1
+p176*(Stud*F*IGy1)
+p178*(ArboLav*ITR1)  +ps1
U2= p111 +p112*(12*d144*2) +p313*enlig1  +p170*lsum2
+p176*(Stud*F*IGy2)
+p178*(ArboLav*ITR2)  +ps2
.
.
U9= p111 +p112*(12*d844*2) +p313*enlig1  +p170*lsum9
+p176*(Stud*F*IGy9)
+p178*(ArboLav*ITR9)  +ps9

-bicycle

U10= p121 +p112*(4*d044*2) +p101*(fradrag1*2) +p313*enlig1
+p170*lsum1
+p176*(Stud*F*IGy1) +p178*(ArboLav*ITR1)  +ps1
U11= p121 +p112*(4*d144*2) +p101*(fradrag2*2) +p313*enlig1
+p170*lsum2
+p176*(Stud*F*IGy2) +p178*(ArboLav*ITR2)  +ps2
.
.
U18= p121 +p112*(4*d844*2) +p101*(fradrag9*2) +p313*enlig1
+p170*lsum9
+p176*(Stud*F*IGy9) +p178*(ArboLav*ITR9)  +ps9

- car
U19=p101*(fradrag1*2) +p101*(kmomk*d044*2) +p112*(d045*2)
+p126*CarPrLic
+p128*lfeq(d9,4)  +p170*lsum1 +p176*(Stud*F*IGy1)
+p178*(ArboLav*ITR1)  +ps1
U20=p101*(fradrag2*2) +p101*(kmomk*d144*2) +p112*(d145*2)
+p126*CarPrLic
+p128*lfeq(d9,4)  +p170*lsum2 +p176*(Stud*F*IGy2)
+p178*(ArboLav*ITR2)  +ps2
.
.
U27=p101*(fradrag9*2) +p101*(kmomk*d844*2) +p112*(d845*2)
+p126*CarPrLic
+p128*lfeq(d9,4)  +p170*lsum9 +p176*(Stud*F*IGy9)
+p178*(ArboLav*ITR9)  +ps9

- CP
U28=p131 +p101*(fradrag1*2) +p101*((kmomk/2)*d044*2)
+p132*(d045*2) +p231*F
+p129*lfeq(d9,4)  +p170*lsum1 +p176*(Stud*F*IGy1)
+p178*(ArboLav*ITR1)  +ps1
U29=p131 +p101*(fradrag2*2) +p101*((kmomk/2)*d144*2)
+p132*(d145*2) +p231*F
+p129*lfeq(d9,4)  +p170*lsum2 +p176*(Stud*F*IGy2)
+p178*(ArboLav*ITR2)  +ps2
.
.

U36=p131 +p101*(fradrag9*2) +p101*((kmomk/2)*d844*2)
+p132*(d845*2) +p231*F
+p129*lfeq(d9,4)  +p170*lsum9 +p176*(Stud*F*IGy9)
+p178*(ArboLav*ITR9)  +ps9

- Bus
U37=p141 +p101*(fradrag1*2) +p101*(Buscost1*2*ifit(d2,67))
+p112*(bustid1*2)
+p146*HT +p142*d053 +p441*(Y*F) +p341*((1-Y)*F)
+p145*Enlig  +p170*lsum1
```

```
+p176*(Stud*F*IGy1) +p178*(ArboLav*ITR1)  +ps1
U38=p141 +p101*(fradrag2*2) +p101*(Buscost2*2*iflt(d2,67))
+p112*(bustid2*2)
+p146*HT +p142*d153 +p441*(Y*F) +p341*((1-Y)*F)
+p145*Enlig  +p170*Isum2
+p176*(Stud*F*IGy2) +p178*(ArboLav*ITR2)  +ps2
.
.
U45=p141 +p101*(fradrag9*2) +p101*(Buscost9*2*iflt(d2,67))
+p112*(bustid9*2)
+p146*HT +p142*d853 +p441*(Y*F) +p341*((1-Y)*F)
+p145*Enlig  +p170*Isum9
+p176*(Stud*F*IGy9) +p178*(ArboLav*ITR9)  +ps9

- Train
U46=p151 +p101*(fradrag1*2) +p101*(Togcost1*2*(1-
ifge(d2,67)*0.5))
+p112*(Togtid1*2) +p156*d13 +p154*d054 +p441*(Y*F)
+p341*((1-Y)*F) +p145*Enlig
+p170*Isum1 +p176*(Stud*F*IGy1) +p178*(ArboLav*ITR1)  +ps1
U47=p151 +p101*(fradrag2*2) +p101*(Togcost2*2*(1-
ifge(d2,67)*0.5))
+p112*(Togtid2*2) +p156*d13 +p154*d154 +p441*(Y*F)
+p341*((1-Y)*F) +p145*Enlig
+p170*Isum2 +p176*(Stud*F*IGy2) +p178*(ArboLav*ITR2)  +ps2
.
.

U54=p151 +p101*(fradrag9*2) +p101*(Togcost9*2*(1-
ifge(d2,67)*0.5))
+p112*(Togtid9*2) +p156*d13 +p154*d854 +p441*(Y*F)
+p341*((1-Y)*F) +p145*Enlig
+p170*Isum9 +p176*(Stud*F*IGy9) +p178*(ArboLav*ITR9)  +ps9
```

6.3 Annex 3: Sub model formulation, (home, work shopping, home)

Work and Shopping Tours (chain 123)
DATA 1800
STOP 30
stats 1 1 0
- PRINT 4
- slow car CarP Bus Train
- 1 36 37 72 73 108 109 144 145 180
-slow if over 5 km
Nonav 1,801 2,801 3,801 4,801
Nonav 5,802 6,802 7,802 8,802
Nonav 9,803 10,803 11,803 12,803
Nonav 13,804 14,804 15,804 16,804
Nonav 17,805 18,805 19,805 20,805
Nonav 21,806 22,806 23,806 24,806
Nonav 25,807 26,807 27,807 28,807
Nonav 29,808 30,808 31,808 32,808
Nonav 33,809 34,809 35,809 36,809
- no licence or younger than 18 NO CAR
Nonav 73,202 74,202 75,202 76,202 77,202 78,202
Nonav 79,202 80,202 81,202 82,202 83,202 84,202
Nonav 85,202 86,202 87,202 88,202 89,202 90,202
Nonav 91,202 92,202 93,202 94,202 95,202 96,202
Nonav 97,202 98,202 99,202 100,202 101,202 102,202
Nonav 103,202 104,202 105,202 106,202 107,202 108,202
- Bicycle if distance over 30 km
Nonav 37,1001 38,1002 39,1003 40,1004 41,1005 42,1006
Nonav 43,1007 44,1008 45,1009 46,1010 47,1011 48,1012
Nonav 49,1013 50,1014 51,1015 52,1016 53,1017 54,1018
Nonav 55,1019 56,1020 57,1021 58,1022 59,1023 60,1024
Nonav 61,1025 62,1026 63,1027 64,1028 65,1029 66,1030
Nonav 67,1031 68,1032 69,1033 70,1034 71,1035 72,1036
- Bornholm har ikke tog - Train on short distances

- Bus if gaabus=5
Nonav 145,203 146,203 147,203 148,203 149,203 150,203
Nonav 151,203 152,203 153,203 154,203 155,203 156,203
Nonav 157,203 158,203 159,203 160,203 161,203 162,203
Nonav 163,203 164,203 165,203 166,203 167,203 168,203
Nonav 169,203 170,203 171,203 172,203 173,203 174,203
Nonav 175,203 176,203 177,203 178,203 179,203 180,203
- Bornholm har ikke tog - Train on short distances
Nonav 181,401 182,402 183,403 184,404 185,405 186,406
Nonav 187,407 188,408 189,409 190,410 191,411 192,412
Nonav 193,413 194,414 195,415 196,416 197,417 198,418
Nonav 199,419 200,420 201,421 202,422 203,423 204,424
Nonav 205,425 206,426 207,427 208,428 209,429 210,430
Nonav 211,431 212,432 213,433 214,434 215,435 216,436
- Bus if distance is short
Nonav 145,601 146,602 147,603 148,604 149,605 150,606
Nonav 151,607 152,608 153,609 154,610 155,611 156,612
Nonav 157,613 158,614 159,615 160,616 161,617 162,618
Nonav 163,619 164,620 165,621 166,622 167,623 168,624
Nonav 169,625 170,626 171,627 172,628 173,629 174,630
Nonav 175,631 176,632 177,633 178,634 179,635 180,636
END

001 Slow
002 Car
003 CarPas
004 Bus
005 Train
101 Kroner
102 Income
103 KmInc
203 KmInLF
111 WalkASC
112 Time
113 Enlig1 t 0
313 Enlig1F
114 SelvstM
116 HT
126 CarPrLic
121 BicycASC
326 CPLHjmF
127 Stud
427 StudF
128 NoCarC
129 NoCarCP
131 CPASC
231 CPASCF
331 CPASCHM
431 CPASCLM
132 CPtime
133 XCPTime
135 CPgl

136 CarLicCP
137 CPpartKK
141 BusASC
241 BsASCYM
341 BsASCGF
441 BsASCYF
142 PBusLsDs t 0
143 SBusLsDs t 0
144 PBusLsBo t 0
145 Enlig t 0
146 HT t 0
147 BusTime
148 BusGL
248 BunempF
149 GaaBus
151 TrnASC
152 PTogtid
154 PTrnLsDs t 0
155 STrnLsDs t 0
156 GaaSta
159 GaaBus
170 PBeskft

172 PSelvbsk
173 PArbBsk
174 PStudH1
175 PStudH3
176 GymStudF
177 OfadmUd t 0
178 OFLArbL t 0
179 PArbH1
180 SBeskft

182 SSelvbsk
183 SArbeBsk
184 SStudH1
185 SStudH3
186 SStudGym
187 SSelvH1
188 SSelvHote
189 SArbH1
190 Befolkn
191 Sbefolkn
200 PWorkArb
201 SWorkArb
210 OFFSHOs5
211 OFst1t3
212 H3exclHF
213 H1ArblsF
214 H1Fkt
215 PELandMM
216 PEFktLF
217 GymnStud
218 OFStud
219 OFFkt
220 ByAM356
221 ByAF356
222 ByAM4
223 H1M356
224 H1F346
225 H3alle
226 HRalle
227 PEM5
228 PEF8

251 H1s356M
252 H1s346F
253 OFs78
254 BAFRs356
258 PEs5M
259 PEs8F
260 SPEs7F
261 SPEs5M
262 SH1s78F
263 SH1s2F
264 SH3sNot8
265 SOFs123M
266 SOFFSHs5 t 0

160 TrnLosBo t 0
153 TrnLoSbo t 0
122 Enlig1BC t 0
121 BicycASC

270 Beskft
271 Landbrug
272 Fremst
273 ByggAnl
274 Handel1
275 Handel2
276 Handel3
277 HotelRes
278 Transp
279 PengeFin
280 OffAdm
281 Folkesk
282 Hospital
283 Gymnasie
284 Uddann.

374 SHandel1
375 SHandel2
376 SHandel3
377 SHotelRes
378 STransp
379 SPengeFin
380 SOffAdm
381 SFolkesk
382 SHospital
383 SGymnasie
384 SUddann.

Income=d38-5.65

M=ifeq(d14,1)
F=1-M
H=ifge(income,0)
L=1-H
Y=IFLE(d2,50)
licence=ifeq(d10,1)
GL=ifge(d2,50)
SelvstM=ifin(d16:11,12)*m
HjemmeF=ifin(d16:3,9)*f
bt=51
togkmt=57
by=17
CarPrLic=ifin(d9:1,3)*d9/max(d11,1)
CarPrLiF=CarPrLic*F
nolicenc=ifeq(d11,0)
HT=ifin(d27:33,34)
enlig=ifne(d3,2)/D38
enlig1=ifeq(d3,1)*F
-stud
ST1=ifin(d16:1,2)
-arblos
ST2=ifeq(d16,3)+ifeq(d16,6)
-ikke arb. sog
ST3=ifin(d16:4,5)+ifeq(d16,7)+ifeq(d16,9)
-selvstændig
ST4=ifeq(d16,11)
-landmænd
ST5=ifin(d16:12,13)
-arbejdere
ST6=ifeq(d16,14)+ifin(d16:20,21)
-Lavere funktionær
ST7=ifin(d16:31,32)
-Højere funktionær
ST8=ifeq(d16,30)
-resten
ST9=ifeq(d16:8,98,99)

Stud=st1
NotArb=st2+st3+st4+st5
Selvst=st4+st5
Arb=st6+st7+st8
ArboLav=st6+st7
StEjArb=st1+st2+st3+st4+st5

d201=ifne(d27,35)
d202=ifeq(d10,1) and ifge(d2,18)
d203=iflt(d12,5)
d204=recode(d12:0,0,0,3,6)

D401 = ifge(d044,5) and ifne(d27,35)
D402 = ifge(d044,5) and ifne(d27,35)
D403 = ifge(d044,5) and ifne(d27,35)
D404 = ifge(d044,5) and ifne(d27,35)
D405 = ifge(d244,5) and ifne(d27,35)
D406 = ifge(d244,5) and ifne(d27,35)
D407 = ifge(d244,5) and ifne(d27,35)
D408 = ifge(d244,5) and ifne(d27,35)
D409 = ifge(d444,5) and ifne(d27,35)
D410 = ifge(d444,5) and ifne(d27,35)
D411 = ifge(d444,5) and ifne(d27,35)
D412 = ifge(d444,5) and ifne(d27,35)
D413 = ifge(d644,5) and ifne(d27,35)
D414 = ifge(d644,5) and ifne(d27,35)
D415 = ifge(d644,5) and ifne(d27,35)
D416 = ifge(d644,5) and ifne(d27,35)
D417 = ifge(d844,5) and ifne(d27,35)
D418 = ifge(d844,5) and ifne(d27,35)
D419 = ifge(d844,5) and ifne(d27,35)
D420 = ifge(d844,5) and ifne(d27,35)
D421 = ifge(d1044,5) and ifne(d27,35)
D422 = ifge(d1044,5) and ifne(d27,35)
D423 = ifge(d1044,5) and ifne(d27,35)
D424 = ifge(d1044,5) and ifne(d27,35)
D425 = ifge(d1244,5) and ifne(d27,35)
D426 = ifge(d1244,5) and ifne(d27,35)
D427 = ifge(d1244,5) and ifne(d27,35)
D428 = ifge(d1244,5) and ifne(d27,35)
D429 = ifge(d1444,5) and ifne(d27,35)
D430 = ifge(d1444,5) and ifne(d27,35)
D431 = ifge(d1444,5) and ifne(d27,35)
D432 = ifge(d1444,5) and ifne(d27,35)
D433 = ifge(d1644,5) and ifne(d27,35)
D434 = ifge(d1644,5) and ifne(d27,35)
D435 = ifge(d1644,5) and ifne(d27,35)
D436 = ifge(d1644,5) and ifne(d27,35)

D601 = ifge(d044,d204)
D602 = ifge(d044,d204)
D603 = ifge(d044,d204)
D604 = ifge(d044,d204)
D605 = ifge(d244,d204)
D606 = ifge(d244,d204)
D607 = ifge(d244,d204)
D608 = ifge(d244,d204)
D609 = ifge(d444,d204)
D610 = ifge(d444,d204)
D611 = ifge(d444,d204)
D612 = ifge(d444,d204)
D613 = ifge(d644,d204)
D614 = ifge(d644,d204)
D615 = ifge(d644,d204)
D616 = ifge(d644,d204)
D617 = ifge(d844,d204)
D618 = ifge(d844,d204)
D619 = ifge(d844,d204)
D620 = ifge(d844,d204)
D621 = ifge(d1044,d204)
D622 = ifge(d1044,d204)
D623 = ifge(d1044,d204)
D624 = ifge(d1044,d204)
D625 = ifge(d1244,d204)
D626 = ifge(d1244,d204)
D627 = ifge(d1244,d204)
D628 = ifge(d1244,d204)
D629 = ifge(d1444,d204)
D630 = ifge(d1444,d204)
D631 = ifge(d1444,d204)
D632 = ifge(d1444,d204)
D633 = ifge(d1644,d204)
D634 = ifge(d1644,d204)
D635 = ifge(d1644,d204)
D636 = ifge(d1644,d204)

D801 = ifle(d044,10)
D802 = ifle(d244,10)
D803 = ifle(d444,10)
D804 = ifle(d644,10)
D805 = ifle(d844,10)
D806 = ifle(d1044,10)
D807 = ifle(d1244,10)
D808 = ifle(d1444,10)
D809 = ifle(d1644,10)

D1001 = ifle(d44,50)
D1002 = ifle(d44,50)
D1003 = ifle(d44,50)
D1004 = ifle(d44,50)
D1005 = ifle(d244,50)
D1006 = ifle(d244,50)
D1007 = ifle(d244,50)
D1008 = ifle(d244,50)
D1009 = ifle(d444,50)
D1010 = ifle(d444,50)
D1011 = ifle(d444,50)
D1012 = ifle(d444,50)
D1013 = ifle(d644,50)
D1014 = ifle(d644,50)
D1015 = ifle(d644,50)
D1016 = ifle(d644,50)

D1019 = ifle(d844,50)
D1020 = ifle(d844,50)
D1021 = ifle(d1044,50)
D1022 = ifle(d1044,50)
D1023 = ifle(d1044,50)
D1024 = ifle(d1044,50)
D1025 = ifle(d1244,50)
D1026 = ifle(d1244,50)
D1027 = ifle(d1244,50)
D1028 = ifle(d1244,50)
D1029 = ifle(d1444,50)
D1030 = ifle(d1444,50)
D1031 = ifle(d1444,50)
D1032 = ifle(d1444,50)
D1033 = ifle(d1644,50)
D1034 = ifle(d1644,50)
D1035 = ifle(d1644,50)
D1036 = ifle(d1644,50)

Bustid1 = 60*d046/d048+10
Bustid2 = 60*d046/d048+10
Bustid3 = 60*d046/d048+10
Bustid4 = 60*d046/d048+10
Bustid5 = 60*d246/d248+10
Bustid6 = 60*d246/d248+10
Bustid7 = 60*d246/d248+10
Bustid8 = 60*d246/d248+10
Bustid9 = 60*d446/d448+10
Bustid10 = 60*d446/d448+10
Bustid11 = 60*d446/d448+10
Bustid12 = 60*d446/d448+10
Bustid13 = 60*d646/d648+10
Bustid14 = 60*d646/d648+10
Bustid15 = 60*d646/d648+10

Bustid16 = 60*d646/d648+10
Bustid17 = 60*d846/d848+10
Bustid18 = 60*d846/d848+10
Bustid19 = 60*d846/d848+10
Bustid20 = 60*d846/d848+10
Bustid21 = 60*d1046/d1048+10
Bustid22 = 60*d1046/d1048+10
Bustid23 = 60*d1046/d1048+10
Bustid24 = 60*d1046/d1048+10
Bustid25 = 60*d1246/d1248+10
Bustid26 = 60*d1246/d1248+10
Bustid27 = 60*d1246/d1248+10
Bustid28 = 60*d1246/d1248+10
Bustid29 = 60*d1446/d1448+10
Bustid30 = 60*d1446/d1448+10
Bustid31 = 60*d1446/d1448+10
Bustid32 = 60*d1446/d1448+10
Bustid33 = 60*d1646/d1648+10
Bustid34 = 60*d1646/d1648+10
Bustid35 = 60*d1646/d1648+10
Bustid36 = 60*d1646/d1648+10

PTogtid1 = d52
PTogtid2 = d52
PTogtid3 = d52
PTogtid4 = d52
PTogtid5 = d252
PTogtid6 = d252
PTogtid7 = d252
PTogtid8 = d252
PTogtid9 = d452
PTogtid10 = d452
PTogtid11 = d452
PTogtid12 = d452
PTogtid13 = d652
PTogtid14 = d652
PTogtid15 = d652
PTogtid16 = d652
PTogtid17 = d852
PTogtid18 = d852
PTogtid19 = d852
PTogtid20 = d852
PTogtid21 = d1052
PTogtid22 = d1052
PTogtid23 = d1052
PTogtid24 = d1052
PTogtid25 = d1252
PTogtid26 = d1252
PTogtid27 = d1252
PTogtid28 = d1252
PTogtid29 = d1452
PTogtid30 = d1452
PTogtid31 = d1452
PTogtid32 = d1452
PTogtid33 = d1652
PTogtid34 = d1652
PTogtid35 = d1652
PTogtid36 = d1652

- beregner valg Tog/bus PT=1 tog, PT=0 bus

BusLoS1 = d31

TogLoS11 = d054
TogLoS12 = d054
TogLoS13 = d054
TogLoS14 = d054
TogLoS15 = d254
TogLoS16 = d254
TogLoS17 = d254
TogLoS18 = d254
TogLoS19 = d454
TogLoS110 = d454
TogLoS111 = d454
TogLoS112 = d454
TogLoS113 = d654
TogLoS114 = d654
TogLoS115 = d654
TogLoS116 = d654
TogLoS117 = d854
TogLoS118 = d854
TogLoS119 = d854
TogLoS120 = d854
TogLoS121 = d1054
TogLoS122 = d1054
TogLoS123 = d1054
TogLoS124 = d1054
TogLoS125 = d1254
TogLoS126 = d1254
TogLoS127 = d1254
TogLoS128 = d1254
TogLoS129 = d1454
TogLoS130 = d1454
TogLoS131 = d1454
TogLoS132 = d1454
TogLoS133 = d1654
TogLoS134 = d1654
TogLoS135 = d1654
TogLoS136 = d1654

togLoS2 = d34

BusCost1 = d050
BusCost2 = d050
BusCost3 = d050
BusCost4 = d050
BusCost5 = d250
BusCost6 = d250
BusCost7 = d250
BusCost8 = d250
BusCost9 = d450
BusCost10 = d450
BusCost11 = d450
BusCost12 = d450
BusCost13 = d650
BusCost14 = d650
BusCost15 = d650
BusCost16 = d650
BusCost17 = d850
BusCost18 = d850
BusCost19 = d850
BusCost20 = d850
BusCost21 = d1050
BusCost22 = d1050
BusCost23 = d1050
BusCost24 = d1050
BusCost25 = d1250
BusCost26 = d1250
BusCost27 = d1250
BusCost28 = d1250
BusCost29 = d1450
BusCost30 = d1450
BusCost31 = d1450
BusCost32 = d1450
BusCost33 = d1650
BusCost34 = d1650
BusCost35 = d1650
BusCost36 = d1650

TogCost1 = d051
TogCost2 = d051
TogCost3 = d051
TogCost4 = d051
TogCost5 = d251
TogCost6 = d251
TogCost7 = d251
TogCost8 = d251
TogCost9 = d451
TogCost10 = d451
TogCost11 = d451
TogCost12 = d451
TogCost13 = d651
TogCost14 = d651
TogCost15 = d651
TogCost16 = d651
TogCost17 = d851
TogCost18 = d851
TogCost19 = d851
TogCost20 = d851
TogCost21 = d1051
TogCost22 = d1051
TogCost23 = d1051
TogCost24 = d1051

TogCost25 = d1251
TogCost26 = d1251
TogCost27 = d1251
TogCost28 = d1251
TogCost29 = d1451
TogCost30 = d1451
TogCost31 = d1451
TogCost32 = d1451
TogCost33 = d1651
TogCost34 = d1651
TogCost35 = d1651
TogCost36 = d1651

IBEF1 = log(d055)
IBEF2 = log(d055)
IBEF3 = log(d055)
IBEF4 = log(d055)
IBEF5 = log(d255)
IBEF6 = log(d255)
IBEF7 = log(d255)
IBEF8 = log(d255)
IBEF9 = log(d455)
IBEF10 = log(d455)
IBEF11 = log(d455)
IBEF12 = log(d455)
IBEF13 = log(d655)
IBEF14 = log(d655)
IBEF15 = log(d655)
IBEF16 = log(d655)
IBEF17 = log(d855)
IBEF18 = log(d855)
IBEF19 = log(d855)
IBEF20 = log(d855)
IBEF21 = log(d1055)
IBEF22 = log(d1055)
IBEF23 = log(d1055)
IBEF24 = log(d1055)
IBEF25 = log(d1255)
IBEF26 = log(d1255)
IBEF27 = log(d1255)
IBEF28 = log(d1255)
IBEF29 = log(d1455)
IBEF30 = log(d1455)
IBEF31 = log(d1455)
IBEF32 = log(d1455)
IBEF33 = log(d1655)
IBEF34 = log(d1655)
IBEF35 = log(d1655)
IBEF36 = log(d1655)

Psum1 = d047 +d056 +d057 +d058 +d059 +d060 +d061 +d062
+d063 +d064 +d065 +d066
+d067 +d068 +d069 +d070
Psum2 = d047 +d056 +d057 +d058 +d059 +d060 +d061 +d062
+d063 +d064 +d065 +d066
+d067 +d068 +d069 +d070
Psum3 = d047 +d056 +d057 +d058 +d059 +d060 +d061 +d062
+d063 +d064 +d065 +d066
+d067 +d068 +d069 +d070
Psum4 = d047 +d056 +d057 +d058 +d059 +d060 +d061 +d062
+d063 +d064 +d065 +d066
+d067 +d068 +d069 +d070
Psum5 = d247 +d256 +d257 +d258 +d259 +d260 +d261 +d262
+d263 +d264 +d265 +d266
+d267 +d268 +d269 +d270
Psum6 = d247 +d256 +d257 +d258 +d259 +d260 +d261 +d262
+d263 +d264 +d265 +d266
+d267 +d268 +d269 +d270
Psum7 = d247 +d256 +d257 +d258 +d259 +d260 +d261 +d262
+d263 +d264 +d265 +d266
+d267 +d268 +d269 +d270
Psum8 = d247 +d256 +d257 +d258 +d259 +d260 +d261 +d262
+d263 +d264 +d265 +d266
+d267 +d268 +d269 +d270
Psum9 = d447 +d456 +d457 +d458 +d459 +d460 +d461 +d462
+d463 +d464 +d465 +d466
+d467 +d468 +d469 +d470
Psum10 = d447 +d456 +d457 +d458 +d459 +d460 +d461 +d462
+d463 +d464 +d465 +d466
+d467 +d468 +d469 +d470
Psum11 = d447 +d456 +d457 +d458 +d459 +d460 +d461 +d462
+d463 +d464 +d465 +d466
+d467 +d468 +d469 +d470
Psum12 = d447 +d456 +d457 +d458 +d459 +d460 +d461 +d462
+d463 +d464 +d465 +d466
+d467 +d468 +d469 +d470
Psum13 = d647 +d656 +d657 +d658 +d659 +d660 +d661 +d662
+d663 +d664 +d665 +d666
+d667 +d668 +d669 +d670
Psum14 = d647 +d656 +d657 +d658 +d659 +d660 +d661 +d662
+d663 +d664 +d665 +d666
+d667 +d668 +d669 +d670
Psum15 = d647 +d656 +d657 +d658 +d659 +d660 +d661 +d662
+d663 +d664 +d665 +d666
+d667 +d668 +d669 +d670
Psum16 = d647 +d656 +d657 +d658 +d659 +d660 +d661 +d662
+d663 +d664 +d665 +d666
+d667 +d668 +d669 +d670
Psum17 = d847 +d856 +d857 +d858 +d859 +d860 +d861 +d862
+d863 +d864 +d865 +d866

```
+d867 +d868 +d869 +d870
Psum18 = d847 +d856 +d857 +d858 +d859 +d860 +d861 +d862
+d863 +d864 +d865 +d866
+d867 +d868 +d869 +d870
Psum19 = d847 +d856 +d857 +d858 +d859 +d860 +d861 +d862
+d863 +d864 +d865 +d866
+d867 +d868 +d869 +d870
Psum20 = d847 +d856 +d857 +d858 +d859 +d860 +d861 +d862
+d863 +d864 +d865 +d866
+d867 +d868 +d869 +d870
Psum21 = d1047 +d1056 +d1057 +d1058 +d1059 +d1060
+d1061 +d1062 +d1063 +d1064
+d1065 +d1066 +d1067 +d1068 +d1069 +d1070
Psum22 = d1047 +d1056 +d1057 +d1058 +d1059 +d1060
+d1061 +d1062 +d1063 +d1064
+d1065 +d1066 +d1067 +d1068 +d1069 +d1070
Psum23 = d1047 +d1056 +d1057 +d1058 +d1059 +d1060
+d1061 +d1062 +d1063 +d1064
+d1065 +d1066 +d1067 +d1068 +d1069 +d1070
Psum24 = d1047 +d1056 +d1057 +d1058 +d1059 +d1060
+d1061 +d1062 +d1063 +d1064
+d1065 +d1066 +d1067 +d1068 +d1069 +d1070
Psum25 = d1247 +d1256 +d1257 +d1258 +d1259 +d1260
+d1261 +d1262 +d1263 +d1264
+d1265 +d1266 +d1267 +d1268 +d1269 +d1270
Psum26 = d1247 +d1256 +d1257 +d1258 +d1259 +d1260
+d1261 +d1262 +d1263 +d1264
+d1265 +d1266 +d1267 +d1268 +d1269 +d1270
Psum27 = d1247 +d1256 +d1257 +d1258 +d1259 +d1260
+d1261 +d1262 +d1263 +d1264
+d1265 +d1266 +d1267 +d1268 +d1269 +d1270
Psum28 = d1247 +d1256 +d1257 +d1258 +d1259 +d1260
+d1261 +d1262 +d1263 +d1264
+d1265 +d1266 +d1267 +d1268 +d1269 +d1270
Psum29 = d1447 +d1456 +d1457 +d1458 +d1459 +d1460
+d1461 +d1462 +d1463 +d1464
+d1465 +d1466 +d1467 +d1468 +d1469 +d1470
Psum30 = d1447 +d1456 +d1457 +d1458 +d1459 +d1460
+d1461 +d1462 +d1463 +d1464
+d1465 +d1466 +d1467 +d1468 +d1469 +d1470
Psum31 = d1447 +d1456 +d1457 +d1458 +d1459 +d1460
+d1461 +d1462 +d1463 +d1464
+d1465 +d1466 +d1467 +d1468 +d1469 +d1470
Psum32 = d1447 +d1456 +d1457 +d1458 +d1459 +d1460
+d1461 +d1462 +d1463 +d1464
+d1465 +d1466 +d1467 +d1468 +d1469 +d1470
Psum33 = d1647 +d1656 +d1657 +d1658 +d1659 +d1660
+d1661 +d1662 +d1663 +d1664
+d1665 +d1666 +d1667 +d1668 +d1669 +d1670
Psum34 = d1647 +d1656 +d1657 +d1658 +d1659 +d1660
+d1661 +d1662 +d1663 +d1664
+d1665 +d1666 +d1667 +d1668 +d1669 +d1670
Psum35 = d1647 +d1656 +d1657 +d1658 +d1659 +d1660
+d1661 +d1662 +d1663 +d1664
+d1665 +d1666 +d1667 +d1668 +d1669 +d1670
Psum36 = d1647 +d1656 +d1657 +d1658 +d1659 +d1660
+d1661 +d1662 +d1663 +d1664
+d1665 +d1666 +d1667 +d1668 +d1669 +d1670

LPS1 = log(Psum1)
LPS2 = log(Psum2)
LPS3 = log(Psum3)
LPS4 = log(Psum4)
LPS5 = log(Psum5)
LPS6 = log(Psum6)
LPS7 = log(Psum7)
LPS8 = log(Psum8)
LPS9 = log(Psum9)
LPS10 = log(Psum10)
LPS11 = log(Psum11)
LPS12 = log(Psum12)
LPS13 = log(Psum13)
LPS14 = log(Psum14)
LPS15 = log(Psum15)
LPS16 = log(Psum16)
LPS17 = log(Psum17)
LPS18 = log(Psum18)
LPS19 = log(Psum19)
LPS20 = log(Psum20)
LPS21 = log(Psum21)
LPS22 = log(Psum22)
LPS23 = log(Psum23)
LPS24 = log(Psum24)
LPS25 = log(Psum25)
LPS26 = log(Psum26)
LPS27 = log(Psum27)
LPS28 = log(Psum28)
LPS29 = log(Psum29)
LPS30 = log(Psum30)
LPS31 = log(Psum31)
LPS32 = log(Psum32)
LPS33 = log(Psum33)
LPS34 = log(Psum34)
LPS35 = log(Psum35)
LPS36 = log(Psum36)

IPH11 = log(max(d59,1)/(psum1-max(d59,1)))
IPH12 = log(max(d59,1)/(psum2-max(d59,1)))
IPH13 = log(max(d59,1)/(psum3-max(d59,1)))
```

IPH14 = log(max(d59,1)/(psum4-max(d59,1)))
IPH15 = log(max(d259,1)/(psum5-max(d259,1)))
IPH16 = log(max(d259,1)/(psum6-max(d259,1)))
IPH17 = log(max(d259,1)/(psum7-max(d259,1)))
IPH18 = log(max(d259,1)/(psum8-max(d259,1)))
IPH19 = log(max(d459,1)/(psum9-max(d459,1)))
IPH110 = log(max(d459,1)/(psum10-max(d459,1)))
IPH111 = log(max(d459,1)/(psum11-max(d459,1)))
IPH112 = log(max(d459,1)/(psum12-max(d459,1)))
IPH113 = log(max(d659,1)/(psum13-max(d659,1)))
IPH114 = log(max(d659,1)/(psum14-max(d659,1)))
IPH115 = log(max(d659,1)/(psum15-max(d659,1)))
IPH116 = log(max(d659,1)/(psum16-max(d659,1)))
IPH117 = log(max(d859,1)/(psum17-max(d859,1)))
IPH118 = log(max(d859,1)/(psum18-max(d859,1)))
IPH119 = log(max(d859,1)/(psum19-max(d859,1)))
IPH120 = log(max(d859,1)/(psum20-max(d859,1)))
IPH121 = log(max(d1059,1)/(psum21-max(d1059,1)))
IPH122 = log(max(d1059,1)/(psum22-max(d1059,1)))
IPH123 = log(max(d1059,1)/(psum23-max(d1059,1)))
IPH124 = log(max(d1059,1)/(psum24-max(d1059,1)))
IPH125 = log(max(d1259,1)/(psum25-max(d1259,1)))
IPH126 = log(max(d1259,1)/(psum26-max(d1259,1)))
IPH127 = log(max(d1259,1)/(psum27-max(d1259,1)))
IPH128 = log(max(d1259,1)/(psum28-max(d1259,1)))
IPH129 = log(max(d1459,1)/(psum29-max(d1459,1)))
IPH130 = log(max(d1459,1)/(psum30-max(d1459,1)))
IPH131 = log(max(d1459,1)/(psum31-max(d1459,1)))
IPH132 = log(max(d1459,1)/(psum32-max(d1459,1)))
IPH133 = log(max(d1659,1)/(psum33-max(d1659,1)))
IPH134 = log(max(d1659,1)/(psum34-max(d1659,1)))
IPH135 = log(max(d1659,1)/(psum35-max(d1659,1)))
IPH136 = log(max(d1659,1)/(psum36-max(d1659,1)))

IPH31 = log(max(d61,1)/(psum1-max(d61,1)))
IPH32 = log(max(d61,1)/(psum2-max(d61,1)))
IPH33 = log(max(d61,1)/(psum3-max(d61,1)))
IPH34 = log(max(d61,1)/(psum4-max(d61,1)))
IPH35 = log(max(d261,1)/(psum5-max(d261,1)))
IPH36 = log(max(d261,1)/(psum6-max(d261,1)))
IPH37 = log(max(d261,1)/(psum7-max(d261,1)))
IPH38 = log(max(d261,1)/(psum8-max(d261,1)))
IPH39 = log(max(d461,1)/(psum9-max(d461,1)))
IPH310 = log(max(d461,1)/(psum10-max(d461,1)))
IPH311 = log(max(d461,1)/(psum11-max(d461,1)))
IPH312 = log(max(d461,1)/(psum12-max(d461,1)))
IPH313 = log(max(d661,1)/(psum13-max(d661,1)))
IPH314 = log(max(d661,1)/(psum14-max(d661,1)))
IPH315 = log(max(d661,1)/(psum15-max(d661,1)))
IPH316 = log(max(d661,1)/(psum16-max(d661,1)))
IPH317 = log(max(d861,1)/(psum17-max(d861,1)))
IPH318 = log(max(d861,1)/(psum18-max(d861,1)))
IPH319 = log(max(d861,1)/(psum19-max(d861,1)))
IPH320 = log(max(d861,1)/(psum20-max(d861,1)))
IPH321 = log(max(d1061,1)/(psum21-max(d1061,1)))
IPH322 = log(max(d1061,1)/(psum22-max(d1061,1)))
IPH323 = log(max(d1061,1)/(psum23-max(d1061,1)))
IPH324 = log(max(d1061,1)/(psum24-max(d1061,1)))
IPH325 = log(max(d1261,1)/(psum25-max(d1261,1)))
IPH326 = log(max(d1261,1)/(psum26-max(d1261,1)))
IPH327 = log(max(d1261,1)/(psum27-max(d1261,1)))
IPH328 = log(max(d1261,1)/(psum28-max(d1261,1)))
IPH329 = log(max(d1461,1)/(psum29-max(d1461,1)))
IPH330 = log(max(d1461,1)/(psum30-max(d1461,1)))
IPH331 = log(max(d1461,1)/(psum31-max(d1461,1)))
IPH332 = log(max(d1461,1)/(psum32-max(d1461,1)))
IPH333 = log(max(d1661,1)/(psum33-max(d1661,1)))
IPH334 = log(max(d1661,1)/(psum34-max(d1661,1)))
IPH335 = log(max(d1661,1)/(psum35-max(d1661,1)))
IPH336 = log(max(d1661,1)/(psum36-max(d1661,1)))

IPHR1 = log(max(d62,1)/(psum1-max(d62,1)))
IPHR2 = log(max(d62,1)/(psum2-max(d62,1)))
IPHR3 = log(max(d62,1)/(psum3-max(d62,1)))
IPHR4 = log(max(d62,1)/(psum4-max(d62,1)))
IPHR5 = log(max(d262,1)/(psum5-max(d262,1)))
IPHR6 = log(max(d262,1)/(psum6-max(d262,1)))
IPHR7 = log(max(d262,1)/(psum7-max(d262,1)))
IPHR8 = log(max(d262,1)/(psum8-max(d262,1)))
IPHR9 = log(max(d462,1)/(psum9-max(d462,1)))
IPHR10 = log(max(d462,1)/(psum10-max(d462,1)))
IPHR11 = log(max(d462,1)/(psum11-max(d462,1)))
IPHR12 = log(max(d462,1)/(psum12-max(d462,1)))
IPHR13 = log(max(d662,1)/(psum13-max(d662,1)))
IPHR14 = log(max(d662,1)/(psum14-max(d662,1)))
IPHR15 = log(max(d662,1)/(psum15-max(d662,1)))
IPHR16 = log(max(d662,1)/(psum16-max(d662,1)))
IPHR17 = log(max(d862,1)/(psum17-max(d862,1)))
IPHR18 = log(max(d862,1)/(psum18-max(d862,1)))
IPHR19 = log(max(d862,1)/(psum19-max(d862,1)))
IPHR20 = log(max(d862,1)/(psum20-max(d862,1)))
IPHR21 = log(max(d1062,1)/(psum21-max(d1062,1)))
IPHR22 = log(max(d1062,1)/(psum22-max(d1062,1)))
IPHR23 = log(max(d1062,1)/(psum23-max(d1062,1)))
IPHR24 = log(max(d1062,1)/(psum24-max(d1062,1)))
IPHR25 = log(max(d1262,1)/(psum25-max(d1262,1)))
IPHR26 = log(max(d1262,1)/(psum26-max(d1262,1)))
IPHR27 = log(max(d1262,1)/(psum27-max(d1262,1)))
IPHR28 = log(max(d1262,1)/(psum28-max(d1262,1)))
IPHR29 = log(max(d1462,1)/(psum29-max(d1462,1)))

IPHR30 = log(max(d1462,1)/(psum30-max(d1462,1)))
IPHR31 = log(max(d1462,1)/(psum31-max(d1462,1)))
IPHR32 = log(max(d1462,1)/(psum32-max(d1462,1)))
IPHR33 = log(max(d1662,1)/(psum33-max(d1662,1)))
IPHR34 = log(max(d1662,1)/(psum34-max(d1662,1)))
IPHR35 = log(max(d1662,1)/(psum35-max(d1662,1)))
IPHR36 = log(max(d1662,1)/(psum36-max(d1662,1)))

IPGY1 = log(max(d69,1)/(psum1-max(d69,1)))
IPGY2 = log(max(d69,1)/(psum2-max(d69,1)))
IPGY3 = log(max(d69,1)/(psum3-max(d69,1)))
IPGY4 = log(max(d69,1)/(psum4-max(d69,1)))
IPGY5 = log(max(d269,1)/(psum5-max(d269,1)))
IPGY6 = log(max(d269,1)/(psum6-max(d269,1)))
IPGY7 = log(max(d269,1)/(psum7-max(d269,1)))
IPGY8 = log(max(d269,1)/(psum8-max(d269,1)))
IPGY9 = log(max(d469,1)/(psum9-max(d469,1)))
IPGY10 = log(max(d469,1)/(psum10-max(d469,1)))
IPGY11 = log(max(d469,1)/(psum11-max(d469,1)))
IPGY12 = log(max(d469,1)/(psum12-max(d469,1)))
IPGY13 = log(max(d669,1)/(psum13-max(d669,1)))
IPGY14 = log(max(d669,1)/(psum14-max(d669,1)))
IPGY15 = log(max(d669,1)/(psum15-max(d669,1)))
IPGY16 = log(max(d669,1)/(psum16-max(d669,1)))
IPGY17 = log(max(d869,1)/(psum17-max(d869,1)))
IPGY18 = log(max(d869,1)/(psum18-max(d869,1)))
IPGY19 = log(max(d869,1)/(psum19-max(d869,1)))
IPGY20 = log(max(d869,1)/(psum20-max(d869,1)))
IPGY21 = log(max(d1069,1)/(psum21-max(d1069,1)))
IPGY22 = log(max(d1069,1)/(psum22-max(d1069,1)))
IPGY23 = log(max(d1069,1)/(psum23-max(d1069,1)))
IPGY24 = log(max(d1069,1)/(psum24-max(d1069,1)))
IPGY25 = log(max(d1269,1)/(psum25-max(d1269,1)))
IPGY26 = log(max(d1269,1)/(psum26-max(d1269,1)))
IPGY27 = log(max(d1269,1)/(psum27-max(d1269,1)))
IPGY28 = log(max(d1269,1)/(psum28-max(d1269,1)))
IPGY29 = log(max(d1469,1)/(psum29-max(d1469,1)))
IPGY30 = log(max(d1469,1)/(psum30-max(d1469,1)))
IPGY31 = log(max(d1469,1)/(psum31-max(d1469,1)))
IPGY32 = log(max(d1469,1)/(psum32-max(d1469,1)))
IPGY33 = log(max(d1669,1)/(psum33-max(d1669,1)))
IPGY34 = log(max(d1669,1)/(psum34-max(d1669,1)))
IPGY35 = log(max(d1669,1)/(psum35-max(d1669,1)))
IPGY36 = log(max(d1669,1)/(psum36-max(d1669,1)))

IPOF1 = log(max(d66,1)/(psum1-max(d66,1)))
IPOF2 = log(max(d66,1)/(psum2-max(d66,1)))
IPOF3 = log(max(d66,1)/(psum3-max(d66,1)))
IPOF4 = log(max(d66,1)/(psum4-max(d66,1)))
IPOF5 = log(max(d266,1)/(psum5-max(d266,1)))
IPOF6 = log(max(d266,1)/(psum6-max(d266,1)))
IPOF7 = log(max(d266,1)/(psum7-max(d266,1)))
IPOF8 = log(max(d266,1)/(psum8-max(d266,1)))
IPOF9 = log(max(d466,1)/(psum9-max(d466,1)))
IPOF10 = log(max(d466,1)/(psum10-max(d466,1)))
IPOF11 = log(max(d466,1)/(psum11-max(d466,1)))
IPOF12 = log(max(d466,1)/(psum12-max(d466,1)))
IPOF13 = log(max(d666,1)/(psum13-max(d666,1)))
IPOF14 = log(max(d666,1)/(psum14-max(d666,1)))
IPOF15 = log(max(d666,1)/(psum15-max(d666,1)))
IPOF16 = log(max(d666,1)/(psum16-max(d666,1)))
IPOF17 = log(max(d866,1)/(psum17-max(d866,1)))
IPOF18 = log(max(d866,1)/(psum18-max(d866,1)))
IPOF19 = log(max(d866,1)/(psum19-max(d866,1)))
IPOF20 = log(max(d866,1)/(psum20-max(d866,1)))
IPOF21 = log(max(d1066,1)/(psum21-max(d1066,1)))
IPOF22 = log(max(d1066,1)/(psum22-max(d1066,1)))
IPOF23 = log(max(d1066,1)/(psum23-max(d1066,1)))
IPOF24 = log(max(d1066,1)/(psum24-max(d1066,1)))
IPOF25 = log(max(d1266,1)/(psum25-max(d1266,1)))
IPOF26 = log(max(d1266,1)/(psum26-max(d1266,1)))
IPOF27 = log(max(d1266,1)/(psum27-max(d1266,1)))
IPOF28 = log(max(d1266,1)/(psum28-max(d1266,1)))
IPOF29 = log(max(d1466,1)/(psum29-max(d1466,1)))
IPOF30 = log(max(d1466,1)/(psum30-max(d1466,1)))
IPOF31 = log(max(d1466,1)/(psum31-max(d1466,1)))
IPOF32 = log(max(d1466,1)/(psum32-max(d1466,1)))
IPOF33 = log(max(d1666,1)/(psum33-max(d1666,1)))
IPOF34 = log(max(d1666,1)/(psum34-max(d1666,1)))
IPOF35 = log(max(d1666,1)/(psum35-max(d1666,1)))
IPOF36 = log(max(d1666,1)/(psum36-max(d1666,1)))

IPTR1 = log(max(d63,1)/(psum1-max(d63,1)))
IPTR2 = log(max(d63,1)/(psum2-max(d63,1)))
IPTR3 = log(max(d63,1)/(psum3-max(d63,1)))
IPTR4 = log(max(d63,1)/(psum4-max(d63,1)))
IPTR5 = log(max(d263,1)/(psum5-max(d263,1)))
IPTR6 = log(max(d263,1)/(psum6-max(d263,1)))
IPTR7 = log(max(d263,1)/(psum7-max(d263,1)))
IPTR8 = log(max(d263,1)/(psum8-max(d263,1)))
IPTR9 = log(max(d463,1)/(psum9-max(d463,1)))
IPTR10 = log(max(d463,1)/(psum10-max(d463,1)))
IPTR11 = log(max(d463,1)/(psum11-max(d463,1)))
IPTR12 = log(max(d463,1)/(psum12-max(d463,1)))
IPTR13 = log(max(d663,1)/(psum13-max(d663,1)))
IPTR14 = log(max(d663,1)/(psum14-max(d663,1)))
IPTR15 = log(max(d663,1)/(psum15-max(d663,1)))
IPTR16 = log(max(d663,1)/(psum16-max(d663,1)))
IPTR17 = log(max(d863,1)/(psum17-max(d863,1)))
IPTR18 = log(max(d863,1)/(psum18-max(d863,1)))

IPTR19 = log(max(d863,1)/(psum19-max(d863,1)))
IPTR20 = log(max(d863,1)/(psum20-max(d863,1)))
IPTR21 = log(max(d1063,1)/(psum21-max(d1063,1)))
IPTR22 = log(max(d1063,1)/(psum22-max(d1063,1)))
IPTR23 = log(max(d1063,1)/(psum23-max(d1063,1)))
IPTR24 = log(max(d1063,1)/(psum24-max(d1063,1)))
IPTR25 = log(max(d1263,1)/(psum25-max(d1263,1)))
IPTR26 = log(max(d1263,1)/(psum26-max(d1263,1)))
IPTR27 = log(max(d1263,1)/(psum27-max(d1263,1)))
IPTR28 = log(max(d1263,1)/(psum28-max(d1263,1)))
IPTR29 = log(max(d1463,1)/(psum29-max(d1463,1)))
IPTR30 = log(max(d1463,1)/(psum30-max(d1463,1)))
IPTR31 = log(max(d1463,1)/(psum31-max(d1463,1)))
IPTR32 = log(max(d1463,1)/(psum32-max(d1463,1)))
IPTR33 = log(max(d1663,1)/(psum33-max(d1663,1)))
IPTR34 = log(max(d1663,1)/(psum34-max(d1663,1)))
IPTR35 = log(max(d1663,1)/(psum35-max(d1663,1)))
IPTR36 = log(max(d1663,1)/(psum36-max(d1663,1)))

IPPE1 = log(max(d64,1)/(psum1-max(d64,1)))
IPPE2 = log(max(d64,1)/(psum2-max(d64,1)))
IPPE3 = log(max(d64,1)/(psum3-max(d64,1)))
IPPE4 = log(max(d64,1)/(psum4-max(d64,1)))
IPPE5 = log(max(d264,1)/(psum5-max(d264,1)))
IPPE6 = log(max(d264,1)/(psum6-max(d264,1)))
IPPE7 = log(max(d264,1)/(psum7-max(d264,1)))
IPPE8 = log(max(d264,1)/(psum8-max(d264,1)))
IPPE9 = log(max(d464,1)/(psum9-max(d464,1)))
IPPE10 = log(max(d464,1)/(psum10-max(d464,1)))
IPPE11 = log(max(d464,1)/(psum11-max(d464,1)))
IPPE12 = log(max(d464,1)/(psum12-max(d464,1)))
IPPE13 = log(max(d664,1)/(psum13-max(d664,1)))
IPPE14 = log(max(d664,1)/(psum14-max(d664,1)))
IPPE15 = log(max(d664,1)/(psum15-max(d664,1)))
IPPE16 = log(max(d664,1)/(psum16-max(d664,1)))
IPPE17 = log(max(d864,1)/(psum17-max(d864,1)))
IPPE18 = log(max(d864,1)/(psum18-max(d864,1)))
IPPE19 = log(max(d864,1)/(psum19-max(d864,1)))
IPPE20 = log(max(d864,1)/(psum20-max(d864,1)))
IPPE21 = log(max(d1064,1)/(psum21-max(d1064,1)))
IPPE22 = log(max(d1064,1)/(psum22-max(d1064,1)))
IPPE23 = log(max(d1064,1)/(psum23-max(d1064,1)))
IPPE24 = log(max(d1064,1)/(psum24-max(d1064,1)))
IPPE25 = log(max(d1264,1)/(psum25-max(d1264,1)))
IPPE26 = log(max(d1264,1)/(psum26-max(d1264,1)))
IPPE27 = log(max(d1264,1)/(psum27-max(d1264,1)))
IPPE28 = log(max(d1264,1)/(psum28-max(d1264,1)))
IPPE29 = log(max(d1464,1)/(psum29-max(d1464,1)))
IPPE30 = log(max(d1464,1)/(psum30-max(d1464,1)))
IPPE31 = log(max(d1464,1)/(psum31-max(d1464,1)))
IPPE32 = log(max(d1464,1)/(psum32-max(d1464,1)))
IPPE33 = log(max(d1664,1)/(psum33-max(d1664,1)))
IPPE34 = log(max(d1664,1)/(psum34-max(d1664,1)))
IPPE35 = log(max(d1664,1)/(psum35-max(d1664,1)))
IPPE36 = log(max(d1664,1)/(psum36-max(d1664,1)))

IPH21 = log(max(d58,1)/(psum1-max(d58,1)))
IPH22 = log(max(d58,1)/(psum2-max(d58,1)))
IPH23 = log(max(d58,1)/(psum3-max(d58,1)))
IPH24 = log(max(d58,1)/(psum4-max(d58,1)))
IPH25 = log(max(d258,1)/(psum5-max(d258,1)))
IPH26 = log(max(d258,1)/(psum6-max(d258,1)))
IPH27 = log(max(d258,1)/(psum7-max(d258,1)))
IPH28 = log(max(d258,1)/(psum8-max(d258,1)))
IPH29 = log(max(d458,1)/(psum9-max(d458,1)))
IPH210 = log(max(d458,1)/(psum10-max(d458,1)))
IPH211 = log(max(d458,1)/(psum11-max(d458,1)))
IPH212 = log(max(d458,1)/(psum12-max(d458,1)))
IPH213 = log(max(d658,1)/(psum13-max(d658,1)))
IPH214 = log(max(d658,1)/(psum14-max(d658,1)))
IPH215 = log(max(d658,1)/(psum15-max(d658,1)))
IPH216 = log(max(d658,1)/(psum16-max(d658,1)))
IPH217 = log(max(d858,1)/(psum17-max(d858,1)))
IPH218 = log(max(d858,1)/(psum18-max(d858,1)))
IPH219 = log(max(d858,1)/(psum19-max(d858,1)))
IPH220 = log(max(d858,1)/(psum20-max(d858,1)))
IPH221 = log(max(d1058,1)/(psum21-max(d1058,1)))
IPH222 = log(max(d1058,1)/(psum22-max(d1058,1)))
IPH223 = log(max(d1058,1)/(psum23-max(d1058,1)))
IPH224 = log(max(d1058,1)/(psum24-max(d1058,1)))
IPH225 = log(max(d1258,1)/(psum25-max(d1258,1)))
IPH226 = log(max(d1258,1)/(psum26-max(d1258,1)))
IPH227 = log(max(d1258,1)/(psum27-max(d1258,1)))
IPH228 = log(max(d1258,1)/(psum28-max(d1258,1)))
IPH229 = log(max(d1458,1)/(psum29-max(d1458,1)))
IPH230 = log(max(d1458,1)/(psum30-max(d1458,1)))
IPH231 = log(max(d1458,1)/(psum31-max(d1458,1)))
IPH232 = log(max(d1458,1)/(psum32-max(d1458,1)))
IPH233 = log(max(d1658,1)/(psum33-max(d1658,1)))
IPH234 = log(max(d1658,1)/(psum34-max(d1658,1)))
IPH235 = log(max(d1658,1)/(psum35-max(d1658,1)))
IPH236 = log(max(d1658,1)/(psum36-max(d1658,1)))

IPHO1 = log(max(d68,1)/(psum1-max(d68,1)))
IPHO2 = log(max(d68,1)/(psum2-max(d68,1)))
IPHO3 = log(max(d68,1)/(psum3-max(d68,1)))
IPHO4 = log(max(d68,1)/(psum4-max(d68,1)))
IPHO5 = log(max(d268,1)/(psum5-max(d268,1)))
IPHO6 = log(max(d268,1)/(psum6-max(d268,1)))
IPHO7 = log(max(d268,1)/(psum7-max(d268,1)))

```
IPHO8 = log(max(d268,1)/(psum8-max(d268,1)))
IPHO9 = log(max(d468,1)/(psum9-max(d468,1)))
IPHO10 = log(max(d468,1)/(psum10-max(d468,1)))
IPHO11 = log(max(d468,1)/(psum11-max(d468,1)))
IPHO12 = log(max(d468,1)/(psum12-max(d468,1)))
IPHO13 = log(max(d668,1)/(psum13-max(d668,1)))
IPHO14 = log(max(d668,1)/(psum14-max(d668,1)))
IPHO15 = log(max(d668,1)/(psum15-max(d668,1)))
IPHO16 = log(max(d668,1)/(psum16-max(d668,1)))
IPHO17 = log(max(d868,1)/(psum17-max(d868,1)))
IPHO18 = log(max(d868,1)/(psum18-max(d868,1)))
IPHO19 = log(max(d868,1)/(psum19-max(d868,1)))
IPHO20 = log(max(d868,1)/(psum20-max(d868,1)))
IPHO21 = log(max(d1068,1)/(psum21-max(d1068,1)))
IPHO22 = log(max(d1068,1)/(psum22-max(d1068,1)))
IPHO23 = log(max(d1068,1)/(psum23-max(d1068,1)))
IPHO24 = log(max(d1068,1)/(psum24-max(d1068,1)))
IPHO25 = log(max(d1268,1)/(psum25-max(d1268,1)))
IPHO26 = log(max(d1268,1)/(psum26-max(d1268,1)))
IPHO27 = log(max(d1268,1)/(psum27-max(d1268,1)))
IPHO28 = log(max(d1268,1)/(psum28-max(d1268,1)))
IPHO29 = log(max(d1468,1)/(psum29-max(d1468,1)))
IPHO30 = log(max(d1468,1)/(psum30-max(d1468,1)))
IPHO31 = log(max(d1468,1)/(psum31-max(d1468,1)))
IPHO32 = log(max(d1468,1)/(psum32-max(d1468,1)))
IPHO33 = log(max(d1668,1)/(psum33-max(d1668,1)))
IPHO34 = log(max(d1668,1)/(psum34-max(d1668,1)))
IPHO35 = log(max(d1668,1)/(psum35-max(d1668,1)))
IPHO36 = log(max(d1668,1)/(psum36-max(d1668,1)))

IPFS1 = log(max(d67,1)/(psum1-max(d67,1)))
IPFS2 = log(max(d67,1)/(psum2-max(d67,1)))
IPFS3 = log(max(d67,1)/(psum3-max(d67,1)))
IPFS4 = log(max(d67,1)/(psum4-max(d67,1)))
IPFS5 = log(max(d267,1)/(psum5-max(d267,1)))
IPFS6 = log(max(d267,1)/(psum6-max(d267,1)))
IPFS7 = log(max(d267,1)/(psum7-max(d267,1)))
IPFS8 = log(max(d267,1)/(psum8-max(d267,1)))
IPFS9 = log(max(d467,1)/(psum9-max(d467,1)))
IPFS10 = log(max(d467,1)/(psum10-max(d467,1)))
IPFS11 = log(max(d467,1)/(psum11-max(d467,1)))
IPFS12 = log(max(d467,1)/(psum12-max(d467,1)))
IPFS13 = log(max(d667,1)/(psum13-max(d667,1)))
IPFS14 = log(max(d667,1)/(psum14-max(d667,1)))
IPFS15 = log(max(d667,1)/(psum15-max(d667,1)))
IPFS16 = log(max(d667,1)/(psum16-max(d667,1)))
IPFS17 = log(max(d867,1)/(psum17-max(d867,1)))
IPFS18 = log(max(d867,1)/(psum18-max(d867,1)))
IPFS19 = log(max(d867,1)/(psum19-max(d867,1)))
IPFS20 = log(max(d867,1)/(psum20-max(d867,1)))
IPFS21 = log(max(d1067,1)/(psum21-max(d1067,1)))
IPFS22 = log(max(d1067,1)/(psum22-max(d1067,1)))
IPFS23 = log(max(d1067,1)/(psum23-max(d1067,1)))
IPFS24 = log(max(d1067,1)/(psum24-max(d1067,1)))
IPFS25 = log(max(d1267,1)/(psum25-max(d1267,1)))
IPFS26 = log(max(d1267,1)/(psum26-max(d1267,1)))
IPFS27 = log(max(d1267,1)/(psum27-max(d1267,1)))
IPFS28 = log(max(d1267,1)/(psum28-max(d1267,1)))
IPFS29 = log(max(d1467,1)/(psum29-max(d1467,1)))
IPFS30 = log(max(d1467,1)/(psum30-max(d1467,1)))
IPFS31 = log(max(d1467,1)/(psum31-max(d1467,1)))
IPFS32 = log(max(d1467,1)/(psum32-max(d1467,1)))
IPFS33 = log(max(d1667,1)/(psum33-max(d1667,1)))
IPFS34 = log(max(d1667,1)/(psum34-max(d1667,1)))
IPFS35 = log(max(d1667,1)/(psum35-max(d1667,1)))
IPFS36 = log(max(d1667,1)/(psum36-max(d1667,1)))

IPUD1 = log(max(d57,1)/(psum1-max(d57,1)))
IPUD2 = log(max(d57,1)/(psum2-max(d57,1)))
IPUD3 = log(max(d57,1)/(psum3-max(d57,1)))
IPUD4 = log(max(d57,1)/(psum4-max(d57,1)))
IPUD5 = log(max(d257,1)/(psum5-max(d257,1)))
IPUD6 = log(max(d257,1)/(psum6-max(d257,1)))
IPUD7 = log(max(d257,1)/(psum7-max(d257,1)))
IPUD8 = log(max(d257,1)/(psum8-max(d257,1)))
IPUD9 = log(max(d457,1)/(psum9-max(d457,1)))
IPUD10 = log(max(d457,1)/(psum10-max(d457,1)))
IPUD11 = log(max(d457,1)/(psum11-max(d457,1)))
IPUD12 = log(max(d457,1)/(psum12-max(d457,1)))
IPUD13 = log(max(d657,1)/(psum13-max(d657,1)))
IPUD14 = log(max(d657,1)/(psum14-max(d657,1)))
IPUD15 = log(max(d657,1)/(psum15-max(d657,1)))
IPUD16 = log(max(d657,1)/(psum16-max(d657,1)))
IPUD17 = log(max(d857,1)/(psum17-max(d857,1)))
IPUD18 = log(max(d857,1)/(psum18-max(d857,1)))
IPUD19 = log(max(d857,1)/(psum19-max(d857,1)))
IPUD20 = log(max(d857,1)/(psum20-max(d857,1)))
IPUD21 = log(max(d1057,1)/(psum21-max(d1057,1)))
IPUD22 = log(max(d1057,1)/(psum22-max(d1057,1)))
IPUD23 = log(max(d1057,1)/(psum23-max(d1057,1)))
IPUD24 = log(max(d1057,1)/(psum24-max(d1057,1)))
IPUD25 = log(max(d1257,1)/(psum25-max(d1257,1)))
IPUD26 = log(max(d1257,1)/(psum26-max(d1257,1)))
IPUD27 = log(max(d1257,1)/(psum27-max(d1257,1)))
IPUD28 = log(max(d1257,1)/(psum28-max(d1257,1)))
IPUD29 = log(max(d1457,1)/(psum29-max(d1457,1)))
IPUD30 = log(max(d1457,1)/(psum30-max(d1457,1)))
IPUD31 = log(max(d1457,1)/(psum31-max(d1457,1)))
IPUD32 = log(max(d1457,1)/(psum32-max(d1457,1)))
IPUD33 = log(max(d1657,1)/(psum33-max(d1657,1)))
```

IPUD34 = log(max(d1657,1)/(psum34-max(d1657,1)))
IPUD35 = log(max(d1657,1)/(psum35-max(d1657,1)))
IPUD36 = log(max(d1657,1)/(psum36-max(d1657,1)))
-Detour
BTx1 = 60*d76/d78+10
BTx2 = 60*d106/d108+10
BTx3 = 60*d136/d138+10
BTx4 = 60*d166/d168+10
BTx5 = 60*d276/d278+10
BTx6 = 60*d306/d308+10
BTx7 = 60*d336/d338+10
BTx8 = 60*d366/d368+10
BTx9 = 60*d476/d478+10
BTx10 = 60*d506/d508+10
BTx11 = 60*d536/d538+10
BTx12 = 60*d566/d568+10
BTx13 = 60*d676/d678+10
BTx14 = 60*d706/d708+10
BTx15 = 60*d736/d738+10
BTx16 = 60*d766/d768+10
BTx17 = 60*d876/d878+10
BTx18 = 60*d906/d908+10
BTx19 = 60*d936/d938+10
BTx20 = 60*d966/d968+10
BTx21 = 60*d1076/d1078+10
BTx22 = 60*d1106/d1108+10
BTx23 = 60*d1136/d1138+10
BTx24 = 60*d1166/d1168+10
BTx25 = 60*d1276/d1278+10
BTx26 = 60*d1306/d1308+10
BTx27 = 60*d1336/d1338+10
BTx28 = 60*d1366/d1368+10
BTx29 = 60*d1476/d1478+10
BTx30 = 60*d1506/d1508+10
BTx31 = 60*d1536/d1538+10
BTx32 = 60*d1566/d1568+10
BTx33 = 60*d1676/d1678+10
BTx34 = 60*d1706/d1708+10
BTx35 = 60*d1736/d1738+10
BTx36 = 60*d1766/d1768+10

BusLoS1 = d31

STogLoS11 = d84
STogLoS12 = d114
STogLoS13 = d144
STogLoS14 = d174
STogLoS15 = d284
STogLoS16 = d314
STogLoS17 = d344
STogLoS18 = d374
STogLoS19 = d484
STogLoS110 = d514
STogLoS111 = d544
STogLoS112 = d574
STogLoS113 = d684
STogLoS114 = d714
STogLoS115 = d744
STogLoS116 = d774
STogLoS117 = d884
STogLoS118 = d914
STogLoS119 = d944
STogLoS120 = d974
STogLoS121 = d1084
STogLoS122 = d1114
STogLoS123 = d1144
STogLoS124 = d1174
STogLoS125 = d1284
STogLoS126 = d1314
STogLoS127 = d1344
STogLoS128 = d1374
STogLoS129 = d1484
STogLoS130 = d1514
STogLoS131 = d1544
STogLoS132 = d1574
STogLoS133 = d1684
STogLoS134 = d1714
STogLoS135 = d1744
STogLoS136 = d1774

togLoS2 = d34

SBusCost1 = d80
SBusCost2 = d110
SBusCost3 = d140
SBusCost4 = d170
SBusCost5 = d280
SBusCost6 = d310
SBusCost7 = d340
SBusCost8 = d370
SBusCost9 = d480
SBusCost10 = d510
SBusCost11 = d540
SBusCost12 = d570
SBusCost13 = d680
SBusCost14 = d710
SBusCost15 = d740
SBusCost16 = d770
SBusCost17 = d880
SBusCost18 = d910

SBusCost19 = d940
SBusCost20 = d970
SBusCost21 = d1080
SBusCost22 = d1110
SBusCost23 = d1140
SBusCost24 = d1170
SBusCost25 = d1280
SBusCost26 = d1310
SBusCost27 = d1340
SBusCost28 = d1370
SBusCost29 = d1480
SBusCost30 = d1510
SBusCost31 = d1540
SBusCost32 = d1570
SBusCost33 = d1680
SBusCost34 = d1710
SBusCost35 = d1740
SBusCost36 = d1770

STogCost1 = d81
STogCost2 = d111
STogCost3 = d141
STogCost4 = d171
STogCost5 = d281
STogCost6 = d311
STogCost7 = d341
STogCost8 = d371
STogCost9 = d481
STogCost10 = d511
STogCost11 = d541
STogCost12 = d571
STogCost13 = d681
STogCost14 = d711
STogCost15 = d741
STogCost16 = d771
STogCost17 = d881
STogCost18 = d911
STogCost19 = d941
STogCost20 = d971
STogCost21 = d1081
STogCost22 = d1111
STogCost23 = d1141
STogCost24 = d1171
STogCost25 = d1281
STogCost26 = d1311
STogCost27 = d1341
STogCost28 = d1371
STogCost29 = d1481
STogCost30 = d1511
STogCost31 = d1541
STogCost32 = d1571
STogCost33 = d1681
STogCost34 = d1711
STogCost35 = d1741
STogCost36 = d1771

SIBEF1 = log(d85)
SIBEF2 = log(d115)
SIBEF3 = log(d145)
SIBEF4 = log(d175)
SIBEF5 = log(d285)
SIBEF6 = log(d315)
SIBEF7 = log(d345)
SIBEF8 = log(d375)
SIBEF9 = log(d485)
SIBEF10 = log(d515)
SIBEF11 = log(d545)
SIBEF12 = log(d575)
SIBEF13 = log(d685)
SIBEF14 = log(d715)
SIBEF15 = log(d745)
SIBEF16 = log(d775)
SIBEF17 = log(d885)
SIBEF18 = log(d915)
SIBEF19 = log(d945)
SIBEF20 = log(d975)
SIBEF21 = log(d1085)
SIBEF22 = log(d1115)
SIBEF23 = log(d1145)
SIBEF24 = log(d1175)
SIBEF25 = log(d1285)
SIBEF26 = log(d1315)
SIBEF27 = log(d1345)
SIBEF28 = log(d1375)
SIBEF29 = log(d1485)
SIBEF30 = log(d1515)
SIBEF31 = log(d1545)
SIBEF32 = log(d1575)
SIBEF33 = log(d1685)
SIBEF34 = log(d1715)
SIBEF35 = log(d1745)
SIBEF36 = log(d1775)

SSUM1 = d77 +d86 +d87 +d88 +d89 +d90 +d91 +d92 +d93
+d94 +d95 +d96 +d97 +d98
+d99 +d100
SSUM2 = d107 +d116 +d117 +d118 +d119 +d120 +d121 +d122
+d123 +d124 +d125 +d126
+d127 +d128 +d129 +d130

SSUM3 = d137 +d146 +d147 +d148 +d149 +d150 +d151 +d152
+d153 +d154 +d155 +d156
+d157 +d158 +d159 +d160
SSUM4 = d167 +d176 +d177 +d178 +d179 +d180 +d181 +d182
+d183 +d184 +d185 +d186
+d187 +d188 +d189 +d190
SSUM5 = d277 +d286 +d287 +d288 +d289 +d290 +d291 +d292
+d293 +d294 +d295 +d296
+d297 +d298 +d299 +d300
SSUM6 = d307 +d316 +d317 +d318 +d319 +d320 +d321 +d322
+d323 +d324 +d325 +d326
+d327 +d328 +d329 +d330
SSUM7 = d337 +d346 +d347 +d348 +d349 +d350 +d351 +d352
+d353 +d354 +d355 +d356
+d357 +d358 +d359 +d360
SSUM8 = d367 +d376 +d377 +d378 +d379 +d380 +d381 +d382
+d383 +d384 +d385 +d386
+d387 +d388 +d389 +d390
SSUM9 = d477 +d486 +d487 +d488 +d489 +d490 +d491 +d492
+d493 +d494 +d495 +d496
+d497 +d498 +d499 +d500
SSUM10 = d507 +d516 +d517 +d518 +d519 +d520 +d521
+d522 +d523 +d524 +d525 +d526
+d527 +d528 +d529 +d530
SSUM11 = d537 +d546 +d547 +d548 +d549 +d550 +d551
+d552 +d553 +d554 +d555 +d556
+d557 +d558 +d559 +d560
SSUM12 = d567 +d576 +d577 +d578 +d579 +d580 +d581
+d582 +d583 +d584 +d585 +d586
+d587 +d588 +d589 +d590
SSUM13 = d677 +d686 +d687 +d688 +d689 +d690 +d691
+d692 +d693 +d694 +d695 +d696
+d697 +d698 +d699 +d700
SSUM14 = d707 +d716 +d717 +d718 +d719 +d720 +d721
+d722 +d723 +d724 +d725 +d726
+d727 +d728 +d729 +d730
SSUM15 = d737 +d746 +d747 +d748 +d749 +d750 +d751
+d752 +d753 +d754 +d755 +d756
+d757 +d758 +d759 +d760
SSUM16 = d767 +d776 +d777 +d778 +d779 +d780 +d781
+d782 +d783 +d784 +d785 +d786
+d787 +d788 +d789 +d790
SSUM17 = d877 +d886 +d887 +d888 +d889 +d890 +d891
+d892 +d893 +d894 +d895 +d896
+d897 +d898 +d899 +d900
SSUM18 = d907 +d916 +d917 +d918 +d919 +d920 +d921
+d922 +d923 +d924 +d925 +d926
+d927 +d928 +d929 +d930
SSUM19 = d937 +d946 +d947 +d948 +d949 +d950 +d951
+d952 +d953 +d954 +d955 +d956
+d957 +d958 +d959 +d960
SSUM20 = d967 +d976 +d977 +d978 +d979 +d980 +d981
+d982 +d983 +d984 +d985 +d986
+d987 +d988 +d989 +d990
SSUM21 = d1077 +d1086 +d1087 +d1088 +d1089 +d1090
+d1091 +d1092 +d1093 +d1094
+d1095 +d1096 +d1097 +d1098 +d1099 +d1100
SSUM22 = d1107 +d1116 +d1117 +d1118 +d1119 +d1120
+d1121 +d1122 +d1123 +d1124
+d1125 +d1126 +d1127 +d1128 +d1129 +d1130
SSUM23 = d1137 +d1146 +d1147 +d1148 +d1149 +d1150
+d1151 +d1152 +d1153 +d1154
+d1155 +d1156 +d1157 +d1158 +d1159 +d1160
SSUM24 = d1167 +d1176 +d1177 +d1178 +d1179 +d1180
+d1181 +d1182 +d1183 +d1184
+d1185 +d1186 +d1187 +d1188 +d1189 +d1190
SSUM25 = d1277 +d1286 +d1287 +d1288 +d1289 +d1290
+d1291 +d1292 +d1293 +d1294
+d1295 +d1296 +d1297 +d1298 +d1299 +d1300
SSUM26 = d1307 +d1316 +d1317 +d1318 +d1319 +d1320
+d1321 +d1322 +d1323 +d1324
+d1325 +d1326 +d1327 +d1328 +d1329 +d1330
SSUM27 = d1337 +d1346 +d1347 +d1348 +d1349 +d1350
+d1351 +d1352 +d1353 +d1354
+d1355 +d1356 +d1357 +d1358 +d1359 +d1360
SSUM28 = d1367 +d1376 +d1377 +d1378 +d1379 +d1380
+d1381 +d1382 +d1383 +d1384
+d1385 +d1386 +d1387 +d1388 +d1389 +d1390
SSUM29 = d1477 +d1486 +d1487 +d1488 +d1489 +d1490
+d1491 +d1492 +d1493 +d1494
+d1495 +d1496 +d1497 +d1498 +d1499 +d1500
SSUM30 = d1507 +d1516 +d1517 +d1518 +d1519 +d1520
+d1521 +d1522 +d1523 +d1524
+d1525 +d1526 +d1527 +d1528 +d1529 +d1530
SSUM31 = d1537 +d1546 +d1547 +d1548 +d1549 +d1550
+d1551 +d1552 +d1553 +d1554
+d1555 +d1556 +d1557 +d1558 +d1559 +d1560
SSUM32 = d1567 +d1576 +d1577 +d1578 +d1579 +d1580
+d1581 +d1582 +d1583 +d1584
+d1585 +d1586 +d1587 +d1588 +d1589 +d1590
SSUM33 = d1677 +d1686 +d1687 +d1688 +d1689 +d1690
+d1691 +d1692 +d1693 +d1694
+d1695 +d1696 +d1697 +d1698 +d1699 +d1700
SSUM34 = d1707 +d1716 +d1717 +d1718 +d1719 +d1720
+d1721 +d1722 +d1723 +d1724
+d1725 +d1726 +d1727 +d1728 +d1729 +d1730
SSUM35 = d1737 +d1746 +d1747 +d1748 +d1749 +d1750
+d1751 +d1752 +d1753 +d1754
+d1755 +d1756 +d1757 +d1758 +d1759 +d1760

```
SSUM36 = d1767 +d1776 +d1777 +d1778 +d1779 +d1780
+d1781 +d1782 +d1783 +d1784
+d1785 +d1786 +d1787 +d1788 +d1789 +d1790

LSs1 = log(Ssum1)
LSs2 = log(Ssum2)
LSs3 = log(Ssum3)
LSs4 = log(Ssum4)
LSs5 = log(Ssum5)
LSs6 = log(Ssum6)
LSs7 = log(Ssum7)
LSs8 = log(Ssum8)
LSs9 = log(Ssum9)
LSs10 = log(Ssum10)
LSs11 = log(Ssum11)
LSs12 = log(Ssum12)
LSs13 = log(Ssum13)
LSs14 = log(Ssum14)
LSs15 = log(Ssum15)
LSs16 = log(Ssum16)
LSs17 = log(Ssum17)
LSs18 = log(Ssum18)
LSs19 = log(Ssum19)
LSs20 = log(Ssum20)
LSs21 = log(Ssum21)
LSs22 = log(Ssum22)
LSs23 = log(Ssum23)
LSs24 = log(Ssum24)
LSs25 = log(Ssum25)
LSs26 = log(Ssum26)
LSs27 = log(Ssum27)
LSs28 = log(Ssum28)
LSs29 = log(Ssum29)
LSs30 = log(Ssum30)
LSs31 = log(Ssum31)
LSs32 = log(Ssum32)
LSs33 = log(Ssum33)
LSs34 = log(Ssum34)
LSs35 = log(Ssum35)
LSs36 = log(Ssum36)

ISH11 = log(max(d89,1))
ISH12 = log(max(d119,1))
ISH13 = log(max(d149,1))
ISH14 = log(max(d179,1))
ISH15 = log(max(d289,1))
ISH16 = log(max(d319,1))
ISH17 = log(max(d349,1))
ISH18 = log(max(d379,1))
ISH19 = log(max(d489,1))
ISH110 = log(max(d519,1))
ISH111 = log(max(d549,1))
ISH112 = log(max(d579,1))
ISH113 = log(max(d689,1))
ISH114 = log(max(d719,1))
ISH115 = log(max(d749,1))
ISH116 = log(max(d779,1))
ISH117 = log(max(d889,1))
ISH118 = log(max(d919,1))
ISH119 = log(max(d949,1))
ISH120 = log(max(d979,1))
ISH121 = log(max(d1089,1))
ISH122 = log(max(d1119,1))
ISH123 = log(max(d1149,1))
ISH124 = log(max(d1179,1))
ISH125 = log(max(d1289,1))
ISH126 = log(max(d1319,1))
ISH127 = log(max(d1349,1))
ISH128 = log(max(d1379,1))
ISH129 = log(max(d1489,1))
ISH130 = log(max(d1519,1))
ISH131 = log(max(d1549,1))
ISH132 = log(max(d1579,1))
ISH133 = log(max(d1689,1))
ISH134 = log(max(d1719,1))
ISH135 = log(max(d1749,1))
ISH136 = log(max(d1779,1))

ISH31 = log(max(d91,1))
ISH32 = log(max(d121,1))
ISH33 = log(max(d151,1))
ISH34 = log(max(d181,1))
ISH35 = log(max(d291,1))
ISH36 = log(max(d321,1))
ISH37 = log(max(d351,1))
ISH38 = log(max(d381,1))
ISH39 = log(max(d491,1))
ISH310 = log(max(d521,1))
ISH311 = log(max(d551,1))
ISH312 = log(max(d581,1))
ISH313 = log(max(d691,1))
ISH314 = log(max(d721,1))
ISH315 = log(max(d751,1))
ISH316 = log(max(d781,1))
ISH317 = log(max(d891,1))
ISH318 = log(max(d921,1))
ISH319 = log(max(d951,1))
ISH320 = log(max(d981,1))
ISH321 = log(max(d1091,1))
ISH322 = log(max(d1121,1))
```

ISH323 = log(max(d1151,1))
ISH324 = log(max(d1181,1))
ISH325 = log(max(d1291,1))
ISH326 = log(max(d1321,1))
ISH327 = log(max(d1351,1))
ISH328 = log(max(d1381,1))
ISH329 = log(max(d1491,1))
ISH330 = log(max(d1521,1))
ISH331 = log(max(d1551,1))
ISH332 = log(max(d1581,1))
ISH333 = log(max(d1691,1))
ISH334 = log(max(d1721,1))
ISH335 = log(max(d1751,1))
ISH336 = log(max(d1781,1))

ISHR1 = log(max(d92,1))
ISHR2 = log(max(d122,1))
ISHR3 = log(max(d152,1))
ISHR4 = log(max(d182,1))
ISHR5 = log(max(d292,1))
ISHR6 = log(max(d322,1))
ISHR7 = log(max(d352,1))
ISHR8 = log(max(d382,1))
ISHR9 = log(max(d492,1))
ISHR10 = log(max(d522,1))
ISHR11 = log(max(d552,1))
ISHR12 = log(max(d582,1))
ISHR13 = log(max(d692,1))
ISHR14 = log(max(d722,1))
ISHR15 = log(max(d752,1))
ISHR16 = log(max(d782,1))
ISHR17 = log(max(d892,1))
ISHR18 = log(max(d922,1))
ISHR19 = log(max(d952,1))
ISHR20 = log(max(d982,1))
ISHR21 = log(max(d1092,1))
ISHR22 = log(max(d1122,1))
ISHR23 = log(max(d1152,1))
ISHR24 = log(max(d1182,1))
ISHR25 = log(max(d1292,1))
ISHR26 = log(max(d1322,1))
ISHR27 = log(max(d1352,1))
ISHR28 = log(max(d1382,1))
ISHR29 = log(max(d1492,1))
ISHR30 = log(max(d1522,1))
ISHR31 = log(max(d1552,1))
ISHR32 = log(max(d1582,1))
ISHR33 = log(max(d1692,1))
ISHR34 = log(max(d1722,1))
ISHR35 = log(max(d1752,1))
ISHR36 = log(max(d1782,1))

ISGY1 = log(max(d99,1))
ISGY2 = log(max(d129,1))
ISGY3 = log(max(d159,1))
ISGY4 = log(max(d189,1))
ISGY5 = log(max(d299,1))
ISGY6 = log(max(d329,1))
ISGY7 = log(max(d359,1))
ISGY8 = log(max(d389,1))
ISGY9 = log(max(d499,1))
ISGY10 = log(max(d529,1))
ISGY11 = log(max(d559,1))
ISGY12 = log(max(d589,1))
ISGY13 = log(max(d699,1))
ISGY14 = log(max(d729,1))
ISGY15 = log(max(d759,1))
ISGY16 = log(max(d789,1))
ISGY17 = log(max(d899,1))
ISGY18 = log(max(d929,1))
ISGY19 = log(max(d959,1))
ISGY20 = log(max(d989,1))
ISGY21 = log(max(d1099,1))
ISGY22 = log(max(d1129,1))
ISGY23 = log(max(d1159,1))
ISGY24 = log(max(d1189,1))
ISGY25 = log(max(d1299,1))
ISGY26 = log(max(d1329,1))
ISGY27 = log(max(d1359,1))
ISGY28 = log(max(d1389,1))
ISGY29 = log(max(d1499,1))
ISGY30 = log(max(d1529,1))
ISGY31 = log(max(d1559,1))
ISGY32 = log(max(d1589,1))
ISGY33 = log(max(d1699,1))
ISGY34 = log(max(d1729,1))
ISGY35 = log(max(d1759,1))
ISGY36 = log(max(d1789,1))

ISOF1 = log(max(d96,1))
ISOF2 = log(max(d126,1))
ISOF3 = log(max(d156,1))
ISOF4 = log(max(d186,1))
ISOF5 = log(max(d296,1))
ISOF6 = log(max(d326,1))
ISOF7 = log(max(d356,1))
ISOF8 = log(max(d386,1))
ISOF9 = log(max(d496,1))
ISOF10 = log(max(d526,1))
ISOF11 = log(max(d556,1))

ISOF12 = log(max(d586,1))
ISOF13 = log(max(d696,1))
ISOF14 = log(max(d726,1))
ISOF15 = log(max(d756,1))
ISOF16 = log(max(d786,1))
ISOF17 = log(max(d896,1))
ISOF18 = log(max(d926,1))
ISOF19 = log(max(d956,1))
ISOF20 = log(max(d986,1))
ISOF21 = log(max(d1096,1))
ISOF22 = log(max(d1126,1))
ISOF23 = log(max(d1156,1))
ISOF24 = log(max(d1186,1))
ISOF25 = log(max(d1296,1))
ISOF26 = log(max(d1326,1))
ISOF27 = log(max(d1356,1))
ISOF28 = log(max(d1386,1))
ISOF29 = log(max(d1496,1))
ISOF30 = log(max(d1526,1))
ISOF31 = log(max(d1556,1))
ISOF32 = log(max(d1586,1))
ISOF33 = log(max(d1696,1))
ISOF34 = log(max(d1726,1))
ISOF35 = log(max(d1756,1))
ISOF36 = log(max(d1786,1))

ISTR1 = log(max(d93,1))
ISTR2 = log(max(d123,1))
ISTR3 = log(max(d153,1))
ISTR4 = log(max(d183,1))
ISTR5 = log(max(d293,1))
ISTR6 = log(max(d323,1))
ISTR7 = log(max(d353,1))
ISTR8 = log(max(d383,1))
ISTR9 = log(max(d493,1))
ISTR10 = log(max(d523,1))
ISTR11 = log(max(d553,1))
ISTR12 = log(max(d583,1))
ISTR13 = log(max(d693,1))
ISTR14 = log(max(d723,1))
ISTR15 = log(max(d753,1))
ISTR16 = log(max(d783,1))
ISTR17 = log(max(d893,1))
ISTR18 = log(max(d923,1))
ISTR19 = log(max(d953,1))
ISTR20 = log(max(d983,1))
ISTR21 = log(max(d1093,1))
ISTR22 = log(max(d1123,1))
ISTR23 = log(max(d1153,1))
ISTR24 = log(max(d1183,1))
ISTR25 = log(max(d1293,1))
ISTR26 = log(max(d1323,1))
ISTR27 = log(max(d1353,1))
ISTR28 = log(max(d1383,1))
ISTR29 = log(max(d1493,1))
ISTR30 = log(max(d1523,1))
ISTR31 = log(max(d1553,1))
ISTR32 = log(max(d1583,1))
ISTR33 = log(max(d1693,1))
ISTR34 = log(max(d1723,1))
ISTR35 = log(max(d1753,1))
ISTR36 = log(max(d1783,1))

ISPE1 = log(max(d94,1))
ISPE2 = log(max(d124,1))
ISPE3 = log(max(d154,1))
ISPE4 = log(max(d184,1))
ISPE5 = log(max(d294,1))
ISPE6 = log(max(d324,1))
ISPE7 = log(max(d354,1))
ISPE8 = log(max(d384,1))
ISPE9 = log(max(d494,1))
ISPE10 = log(max(d524,1))
ISPE11 = log(max(d554,1))
ISPE12 = log(max(d584,1))
ISPE13 = log(max(d694,1))
ISPE14 = log(max(d724,1))
ISPE15 = log(max(d754,1))
ISPE16 = log(max(d784,1))
ISPE17 = log(max(d894,1))
ISPE18 = log(max(d924,1))
ISPE19 = log(max(d954,1))
ISPE20 = log(max(d984,1))
ISPE21 = log(max(d1094,1))
ISPE22 = log(max(d1124,1))
ISPE23 = log(max(d1154,1))
ISPE24 = log(max(d1184,1))
ISPE25 = log(max(d1294,1))
ISPE26 = log(max(d1324,1))
ISPE27 = log(max(d1354,1))
ISPE28 = log(max(d1384,1))
ISPE29 = log(max(d1494,1))
ISPE30 = log(max(d1524,1))
ISPE31 = log(max(d1554,1))
ISPE32 = log(max(d1584,1))
ISPE33 = log(max(d1694,1))
ISPE34 = log(max(d1724,1))
ISPE35 = log(max(d1754,1))
ISPE36 = log(max(d1784,1))

ISH21 = log(max(d90,1))
ISH22 = log(max(d120,1))
ISH23 = log(max(d150,1))
ISH24 = log(max(d180,1))
ISH25 = log(max(d290,1))
ISH26 = log(max(d320,1))
ISH27 = log(max(d350,1))
ISH28 = log(max(d380,1))
ISH29 = log(max(d490,1))
ISH210 = log(max(d520,1))
ISH211 = log(max(d550,1))
ISH212 = log(max(d580,1))
ISH213 = log(max(d690,1))
ISH214 = log(max(d720,1))
ISH215 = log(max(d750,1))
ISH216 = log(max(d780,1))
ISH217 = log(max(d890,1))
ISH218 = log(max(d920,1))
ISH219 = log(max(d950,1))
ISH220 = log(max(d980,1))
ISH221 = log(max(d1090,1))
ISH222 = log(max(d1120,1))
ISH223 = log(max(d1150,1))
ISH224 = log(max(d1180,1))
ISH225 = log(max(d1290,1))
ISH226 = log(max(d1320,1))
ISH227 = log(max(d1350,1))
ISH228 = log(max(d1380,1))
ISH229 = log(max(d1490,1))
ISH230 = log(max(d1520,1))
ISH231 = log(max(d1550,1))
ISH232 = log(max(d1580,1))
ISH233 = log(max(d1690,1))
ISH234 = log(max(d1720,1))
ISH235 = log(max(d1750,1))
ISH236 = log(max(d1780,1))

ISHO1 = log(max(d98,1))
ISHO2 = log(max(d128,1))
ISHO3 = log(max(d158,1))
ISHO4 = log(max(d188,1))
ISHO5 = log(max(d298,1))
ISHO6 = log(max(d328,1))
ISHO7 = log(max(d358,1))
ISHO8 = log(max(d388,1))
ISHO9 = log(max(d498,1))
ISHO10 = log(max(d528,1))
ISHO11 = log(max(d558,1))
ISHO12 = log(max(d588,1))
ISHO13 = log(max(d698,1))
ISHO14 = log(max(d728,1))
ISHO15 = log(max(d758,1))
ISHO16 = log(max(d788,1))
ISHO17 = log(max(d898,1))
ISHO18 = log(max(d928,1))
ISHO19 = log(max(d958,1))
ISHO20 = log(max(d988,1))
ISHO21 = log(max(d1098,1))
ISHO22 = log(max(d1128,1))
ISHO23 = log(max(d1158,1))
ISHO24 = log(max(d1188,1))
ISHO25 = log(max(d1298,1))
ISHO26 = log(max(d1328,1))
ISHO27 = log(max(d1358,1))
ISHO28 = log(max(d1388,1))
ISHO29 = log(max(d1498,1))
ISHO30 = log(max(d1528,1))
ISHO31 = log(max(d1558,1))
ISHO32 = log(max(d1588,1))
ISHO33 = log(max(d1698,1))
ISHO34 = log(max(d1728,1))
ISHO35 = log(max(d1758,1))
ISHO36 = log(max(d1788,1))

ISFS1 = log(max(d97,1))
ISFS2 = log(max(d127,1))
ISFS3 = log(max(d157,1))
ISFS4 = log(max(d187,1))
ISFS5 = log(max(d297,1))
ISFS6 = log(max(d327,1))
ISFS7 = log(max(d357,1))
ISFS8 = log(max(d387,1))
ISFS9 = log(max(d497,1))
ISFS10 = log(max(d527,1))
ISFS11 = log(max(d557,1))
ISFS12 = log(max(d587,1))
ISFS13 = log(max(d697,1))
ISFS14 = log(max(d727,1))
ISFS15 = log(max(d757,1))
ISFS16 = log(max(d787,1))
ISFS17 = log(max(d897,1))
ISFS18 = log(max(d927,1))
ISFS19 = log(max(d957,1))
ISFS20 = log(max(d987,1))
ISFS21 = log(max(d1097,1))
ISFS22 = log(max(d1127,1))
ISFS23 = log(max(d1157,1))
ISFS24 = log(max(d1187,1))
ISFS25 = log(max(d1297,1))
ISFS26 = log(max(d1327,1))

ISFS27 = log(max(d1357,1))
ISFS28 = log(max(d1387,1))
ISFS29 = log(max(d1497,1))
ISFS30 = log(max(d1527,1))
ISFS31 = log(max(d1557,1))
ISFS32 = log(max(d1587,1))
ISFS33 = log(max(d1697,1))
ISFS34 = log(max(d1727,1))
ISFS35 = log(max(d1757,1))
ISFS36 = log(max(d1787,1))

ISFS1 = log(max(d87,1))
ISFS2 = log(max(d117,1))
ISFS3 = log(max(d147,1))
ISFS4 = log(max(d177,1))
ISFS5 = log(max(d287,1))
ISFS6 = log(max(d317,1))
ISFS7 = log(max(d347,1))
ISFS8 = log(max(d377,1))
ISFS9 = log(max(d487,1))
ISFS10 = log(max(d517,1))
ISFS11 = log(max(d547,1))
ISFS12 = log(max(d577,1))
ISFS13 = log(max(d687,1))
ISFS14 = log(max(d717,1))
ISFS15 = log(max(d747,1))
ISFS16 = log(max(d777,1))
ISFS17 = log(max(d887,1))
ISFS18 = log(max(d917,1))
ISFS19 = log(max(d947,1))
ISFS20 = log(max(d977,1))
ISFS21 = log(max(d1087,1))
ISFS22 = log(max(d1117,1))
ISFS23 = log(max(d1147,1))
ISFS24 = log(max(d1177,1))
ISFS25 = log(max(d1287,1))
ISFS26 = log(max(d1317,1))
ISFS27 = log(max(d1347,1))
ISFS28 = log(max(d1377,1))
ISFS29 = log(max(d1487,1))
ISFS30 = log(max(d1517,1))
ISFS31 = log(max(d1547,1))
ISFS32 = log(max(d1577,1))
ISFS33 = log(max(d1687,1))
ISFS34 = log(max(d1717,1))
ISFS35 = log(max(d1747,1))
ISFS36 = log(max(d1777,1))

ISUD1 = log(max(d70,1))
ISUD2 = log(max(d100,1))
ISUD3 = log(max(d130,1))
ISUD4 = log(max(d160,1))
ISUD5 = log(max(d270,1))
ISUD6 = log(max(d300,1))
ISUD7 = log(max(d330,1))
ISUD8 = log(max(d360,1))
ISUD9 = log(max(d470,1))
ISUD10 = log(max(d500,1))
ISUD11 = log(max(d530,1))
ISUD12 = log(max(d560,1))
ISUD13 = log(max(d670,1))
ISUD14 = log(max(d700,1))
ISUD15 = log(max(d730,1))
ISUD16 = log(max(d760,1))
ISUD17 = log(max(d870,1))
ISUD18 = log(max(d900,1))
ISUD19 = log(max(d930,1))
ISUD20 = log(max(d960,1))
ISUD21 = log(max(d1070,1))
ISUD22 = log(max(d1100,1))
ISUD23 = log(max(d1130,1))
ISUD24 = log(max(d1160,1))
ISUD25 = log(max(d1270,1))
ISUD26 = log(max(d1300,1))
ISUD27 = log(max(d1330,1))
ISUD28 = log(max(d1360,1))
ISUD29 = log(max(d1470,1))
ISUD30 = log(max(d1500,1))
ISUD31 = log(max(d1530,1))
ISUD32 = log(max(d1560,1))
ISUD33 = log(max(d1670,1))
ISUD34 = log(max(d1700,1))
ISUD35 = log(max(d1730,1))
ISUD36 = log(max(d1760,1))

- befordringsfradrag

fradrag1=-0.46*(min(max(0,d44-12),38)*1.21+max(0,d44-50)*0.605)
fradrag2=-0.46*(min(max(0,d44-12),38)*1.21+max(0,d44-50)*0.605)
fradrag3=-0.46*(min(max(0,d44-12),38)*1.21+max(0,d44-50)*0.605)
fradrag4=-0.46*(min(max(0,d44-12),38)*1.21+max(0,d44-50)*0.605)
fradrag5=-0.46*(min(max(0,d244-12),38)*1.21+max(0,d244-50)*0.605)
fradrag6=-0.46*(min(max(0,d244-12),38)*1.21+max(0,d244-50)*0.605)

```
fradrag7=-0.46*(min(max(0,d244-12),38)*1.21+max(0,d244-
50)*0.605)
fradrag8=-0.46*(min(max(0,d244-12),38)*1.21+max(0,d244-
50)*0.605)
fradrag9=-0.46*(min(max(0,d444-12),38)*1.21+max(0,d444-
50)*0.605)
fradrag10=-0.46*(min(max(0,d444-12),38)*1.21+max(0,d444-
50)*0.605)
fradrag11=-0.46*(min(max(0,d444-12),38)*1.21+max(0,d444-
50)*0.605)
fradrag12=-0.46*(min(max(0,d444-12),38)*1.21+max(0,d444-
50)*0.605)
fradrag13=-0.46*(min(max(0,d644-12),38)*1.21+max(0,d644-
50)*0.605)
fradrag14=-0.46*(min(max(0,d644-12),38)*1.21+max(0,d644-
50)*0.605)
fradrag15=-0.46*(min(max(0,d644-12),38)*1.21+max(0,d644-
50)*0.605)
fradrag16=-0.46*(min(max(0,d644-12),38)*1.21+max(0,d644-
50)*0.605)
fradrag17=-0.46*(min(max(0,d844-12),38)*1.21+max(0,d844-
50)*0.605)
fradrag18=-0.46*(min(max(0,d844-12),38)*1.21+max(0,d844-
50)*0.605)
fradrag19=-0.46*(min(max(0,d844-12),38)*1.21+max(0,d844-
50)*0.605)
fradrag20=-0.46*(min(max(0,d844-12),38)*1.21+max(0,d844-
50)*0.605)
fradrag21=-0.46*(min(max(0,d1044-12),38)*1.21+max(0,d1044-
50)*0.605)
fradrag22=-0.46*(min(max(0,d1044-12),38)*1.21+max(0,d1044-
50)*0.605)
fradrag23=-0.46*(min(max(0,d1044-12),38)*1.21+max(0,d1044-
50)*0.605)
fradrag24=-0.46*(min(max(0,d1044-12),38)*1.21+max(0,d1044-
50)*0.605)
fradrag25=-0.46*(min(max(0,d1244-12),38)*1.21+max(0,d1244-
50)*0.605)
fradrag26=-0.46*(min(max(0,d1244-12),38)*1.21+max(0,d1244-
50)*0.605)
fradrag27=-0.46*(min(max(0,d1244-12),38)*1.21+max(0,d1244-
50)*0.605)
fradrag28=-0.46*(min(max(0,d1244-12),38)*1.21+max(0,d1244-
50)*0.605)
fradrag29=-0.46*(min(max(0,d1444-12),38)*1.21+max(0,d1444-
50)*0.605)
fradrag30=-0.46*(min(max(0,d1444-12),38)*1.21+max(0,d1444-
50)*0.605)
fradrag31=-0.46*(min(max(0,d1444-12),38)*1.21+max(0,d1444-
50)*0.605)
fradrag32=-0.46*(min(max(0,d1444-12),38)*1.21+max(0,d1444-
50)*0.605)
fradrag33=-0.46*(min(max(0,d1644-12),38)*1.21+max(0,d1644-
50)*0.605)
fradrag34=-0.46*(min(max(0,d1644-12),38)*1.21+max(0,d1644-
50)*0.605)
fradrag35=-0.46*(min(max(0,d1644-12),38)*1.21+max(0,d1644-
50)*0.605)
fradrag36=-0.46*(min(max(0,d1644-12),38)*1.21+max(0,d1644-
50)*0.605)

- car variables
- marginal km omkostning (gennemsnitspris paa benzin 1995)
kmomk=5.92/13.3
CarPrLic= ifin(d9:1,3)*d9/max(d11,1)

- CP variables
CarOgPKK= ifne(d9,4)*ifeq(d22,1)

- bus variables
HT=ifin(d27:33,34)
BusLoSBo=d31*ifle(d40,5)+d32*ifeq(d40,6)+d33*ifeq(d40,7)

mode=recode(d26,1,2,3,4,5,5,6,7,7)
exclude = (1-ifin(mode:1,6))

ChooseD= d73+2*d103+  3*d133+ 4 *d163
 + 5* d273+ 6 *d303+  7*d333+ 8 *d363
 + 9* d473+10 *d503+ 11*d533+12 *d563
 +13* d673+14 *d703+ 15*d733+16 *d763
 +17* d873+18 *d903+ 19*d933+20 *d963
 +21*d1073+22*d1103+23*d1133+24*d1163
 +25*d1273+26*d1303+27*d1333+28*d1363
 +29*d1473+30*d1503+31*d1533+32*d1563
 +33*d1673+34*d1703+35*d1733+36*d1763
Choice=(mode-1)*36+ChooseD

weight=d37

ID=d1

-slow
U1= p111 +p112*(12*d74) +p112*(12*d044*2)  +p170*LPS1
+p252*(lpH11*F*(st3+st5+st6)) +p176*(Stud*F*lpGy1)
+p253*((st7+st8)*lpOF1)
+p274*LsH11 +p276*LsH31 +p279*LsPE1 +p281*LsFS1
+p284*LsUD1 +(log(1/d041)
+log(d71))
U2= p111 +p112*(12*d104) +p112*(12*d044*2)  +p170*LPS2
```

```
+p252*(lpH12*F*(st3+st5+st6)) +p176*(Stud*F*lpGy2)
+p253*((st7+st8)*lpOF2)
+p274*LsH12 +p276*LsH32 +p279*LsPE2 +p281*LsFS2
+p284*LsUD2 +(log(1/d041)
+log(d101))
.
.
U36= p111 +p112*(12*d1764) +p112*(12*d1644*2)
+p170*LPS36
+p252*(lpH136*F*(st3+st5+st6)) +p176*(Stud*F*lpGy36)
+p253*((st7+st8)*lpOF36)
+p274*LsH136 +p276*LsH336 +p279*LsPE36 +p281*LsFS36
+p284*LsUD36 +(log(1/d1641)
+log(d1761))

- bicycle
U37= p121 +p112*(4*d74) +p112*(4*d044*2)
+p101*(fradrag1*2) +p170*LPS1
+p252*(lpH11*F*(st3+st5+st6)) +p176*(Stud*F*lpGy1)
+p253*((st7+st8)*lpOF1)
+p274*LsH11 +p276*LsH31 +p279*LsPE1 +p281*LsFS1
+p284*LsUD1 +(log(1/d041)
+log(d71))
U38= p121 +p112*(4*d104) +p112*(4*d044*2)
+p101*(fradrag2*2) +p170*LPS2
+p252*(lpH12*F*(st3+st5+st6)) +p176*(Stud*F*lpGy2)
+p253*((st7+st8)*lpOF2)
+p274*LsH12 +p276*LsH32 +p279*LsPE2 +p281*LsFS2
+p284*LsUD2 +(log(1/d041)
+log(d101))
.
.
U72= p121 +p112*(4*d1764) +p112*(4*d1644*2)
+p101*(fradrag36*2) +p170*LPS36
+p252*(lpH136*F*(st3+st5+st6)) +p176*(Stud*F*lpGy36)
+p253*((st7+st8)*lpOF36)
+p274*LsH136 +p276*LsH336 +p279*LsPE36 +p281*LsFS36
+p284*LsUD36 +(log(1/d1641)
+log(d1761))


- car
U73=p101*(fradrag1*2) +p101*(kmomk*d044*2) +p112*(d045*2)
+p101*(kmomk*d74)
+p112*d75 +p126*CarPrLic +p128*(1-lfin(d9:1,3)) +p170*LPS1
+p252*(lpH11*F*(st3+st5+st6)) +p176*(Stud*F*lpGy1)
+p253*((st7+st8)*lpOF1)
+p274*LsH11 +p276*LsH31 +p279*LsPE1 +p281*LsFS1
+p284*LsUD1 +(log(1/d041)
+log(d71))
U74=p101*(fradrag2*2) +p101*(kmomk*d044*2) +p112*(d045*2)
+p101*(kmomk*d104)
+p112*d105 +p126*CarPrLic +p128*(1-lfin(d9:1,3))
+p170*LPS2
+p252*(lpH12*F*(st3+st5+st6)) +p176*(Stud*F*lpGy2)
+p253*((st7+st8)*lpOF2)
+p274*LsH12 +p276*LsH32 +p279*LsPE2 +p281*LsFS2
+p284*LsUD2 +(log(1/d041)
+log(d101))
.
.
U108=p101*(fradrag36*2) +p101*(kmomk*d1644*2)
+p112*(d1645*2)
+p101*(kmomk*d1764) +p112*d1765 +p126*CarPrLic +p128*(1-
lfin(d9:1,3))

- CP
U109=p131 +p101*(fradrag1*2) +p101*(kmomk/2*d044*2)
+p132*(d045*2)
+p101*(kmomk/2*d74) +p132*d75 +p129*(1-lfin(d9:1,3))
+p137*ifeq(d22,1)
+p170*LPS1 +p252*(lpH11*F*(st3+st5+st6))
+p176*(Stud*F*lpGy1)
+p253*((st7+st8)*lpOF1) +p274*LsH11 +p276*LsH31
+p279*LsPE1 +p281*LsFS1
+p284*LsUD1 +(log(1/d041) +log(d71))
U110=p131 +p101*(fradrag2*2) +p101*(kmomk/2*d044*2)
+p132*(d045*2)
+p101*(kmomk/2*d104) +p132*d105 +p129*(1-lfin(d9:1,3))
+p137*ifeq(d22,1)
+p170*LPS2 +p252*(lpH12*F*(st3+st5+st6))
+p176*(Stud*F*lpGy2)
+p253*((st7+st8)*lpOF2) +p274*LsH12 +p276*LsH32
+p279*LsPE2 +p281*LsFS2
+p284*LsUD2 +(log(1/d041) +log(d101))
.
.
U144=p131 +p101*(fradrag36*2) +p101*(kmomk/2*d1644*2)
+p132*(d1645*2)
+p101*(kmomk/2*d1764) +p132*d1765 +p129*(1-lfin(d9:1,3))
+p137*ifeq(d22,1)
+p170*LPS36 +p252*(lpH136*F*(st3+st5+st6))
+p176*(Stud*F*lpGy36)
+p253*((st7+st8)*lpOF36) +p274*LsH136 +p276*LsH336
+p279*LsPE36 +p281*LsFS36
+p284*LsUD36 +(log(1/d1641) +log(d1761))

- Bus
U145=p141 +p101*(fradrag1*2) +p101*(d050*2*iflt(d2,67))
+p112*(bustid1*2)
```

+p101*(d80*iflt(d2,67)) +p112*BTx1 +p143*d83 +p170*LPS1
+p252*(lpH11*F*(st3+st5+st6)) +p176*(Stud*F*lpGy1)
+p253*((st7+st8)*lpOF1)
+p274*LsH11 +p276*LsH31 +p279*LsPE1 +p281*LsFS1
+p284*LsUD1 +(log(1/d041)
+log(d71))
U146=p141 +p101*(fradrag2*2) +p101*(d050*2*iflt(d2,67))
+p112*(bustid2*2)
+p101*(d110*iflt(d2,67)) +p112*BTx2 +p143*d113 +p170*LPS2
+p252*(lpH12*F*(st3+st5+st6)) +p176*(Stud*F*lpGy2)
+p253*((st7+st8)*lpOF2)
+p274*LsH12 +p276*LsH32 +p279*LsPE2 +p281*LsFS2
+p284*LsUD2 +(log(1/d041)
+log(d101))
.
.
U180=p141 +p101*(fradrag36*2) +p101*(d1650*2*iflt(d2,67))
+p112*(bustid36*2)
+p101*(d1770*iflt(d2,67)) +p112*BTx36 +p143*d1773
+p170*LPS36
+p252*(lpH136*F*(st3+st5+st6)) +p176*(Stud*F*lpGy36)
+p253*((st7+st8)*lpOF36)
+p274*LsH136 +p276*LsH336 +p279*LsPE36 +p281*LsFS36
+p284*LsUD36 +(log(1/d1641)
+log(d1761))

- Train
U181=p151 +p101*(fradrag1*2) +p101*(d051*2*(1-
ifge(d2,67)*0.5))
+p112*(PTogtid1*2) +p101*(d80*ifgt(d74,1)*(1-ifge(d2,67)*0.5))
+p112*(BTx1*ifgt(d74,1)) +p112*(12*d74*ifle(d74,1))
+p170*LPS1
+p252*(lpH11*F*(st3+st5+st6)) +p176*(Stud*F*lpGy1)
+p253*((st7+st8)*lpOF1)
+p274*LsH11 +p276*LsH31 +p279*LsPE1 +p281*LsFS1
+p284*LsUD1 +(log(1/d041)
+log(d71))
U182=p151 +p101*(fradrag2*2) +p101*(d051*2*(1-
ifge(d2,67)*0.5))
+p112*(PTogtid2*2) +p101*(d110*ifgt(d104,1)*(1-
ifge(d2,67)*0.5))
+p112*(BTx2*ifgt(d104,1)) +p112*(12*d104*ifle(d104,1))
+p170*LPS2
+p252*(lpH12*F*(st3+st5+st6)) +p176*(Stud*F*lpGy2)
+p253*((st7+st8)*lpOF2)
+p274*LsH12 +p276*LsH32 +p279*LsPE2 +p281*LsFS2
+p284*LsUD2 +(log(1/d041)
+log(d101))
.
.

U216=p151 +p101*(fradrag36*2) +p101*(d1651*2*(1-
ifge(d2,67)*0.5))
+p112*(PTogtid36*2) +p101*(d1770*ifgt(d1764,1)*(1-
ifge(d2,67)*0.5))
+p112*(BTx36*ifgt(d1764,1)) +p112*(12*d1764*ifle(d1764,1))
+p170*LPS36
+p252*(lpH136*F*(st3+st5+st6)) +p176*(Stud*F*lpGy36)
+p253*((st7+st8)*lpOF36)
+p274*LsH136 +p276*LsH336 +p279*LsPE36 +p281*LsFS36
+p284*LsUD36 +(log(1/d1641)
+log(d1761))