Conceptualizing and building Markov models

(with emphasis on rates and probabilities)

david.naimark@sunnybrook.ca

Steps to conceptualizing and building a Markov model

- Enumerate possible health states
- From each state consider possible stochastic events that could lead to transitions
  - Construct the cycle trees
- Populate model parameters
  - Incremental utilities
  - Costs
  - Transition probabilities

Steps to conceptualizing and building a Markov model

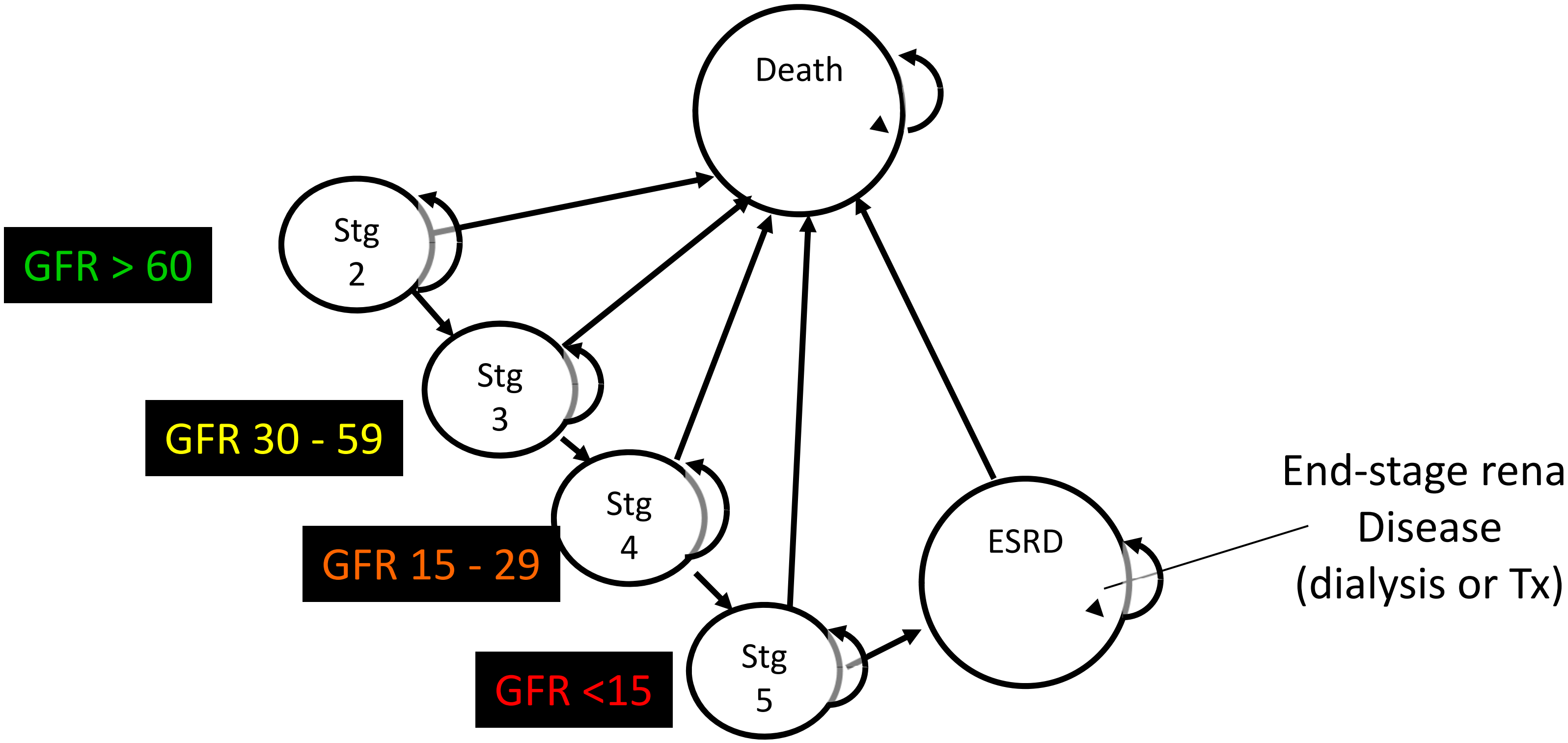
- Enumerate possible health states
- From each state consider possible stochastic events that could lead to transitions
  - Construct the cycle trees
- Populate model parameters
  - Incremental utilities
  - Costs
  - Transition probabilities

Steps to conceptualizing and building a Markov model

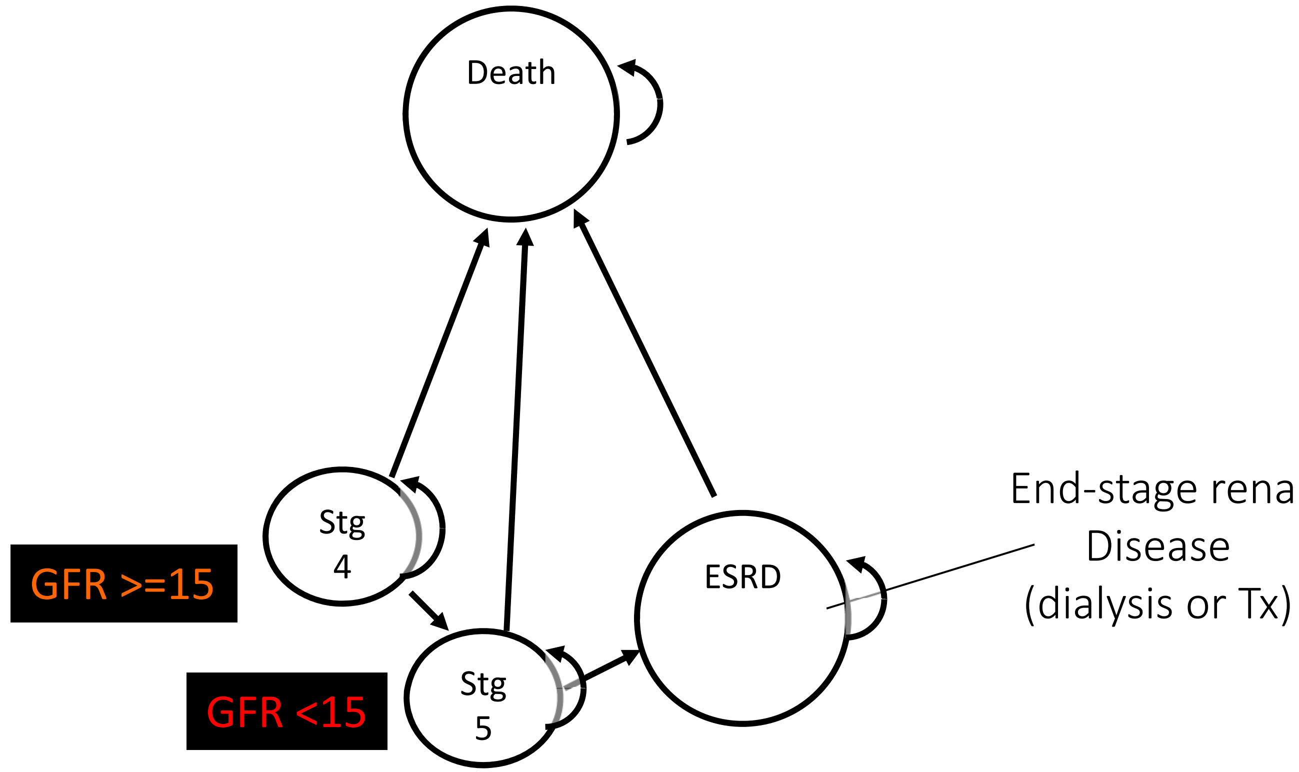
- Enumerate possible health states
- From each state consider possible stochastic events that could lead to transitions
  - Construct the cycle trees
- Populate model parameters
- Debug and validate

Example Markov conceptualization

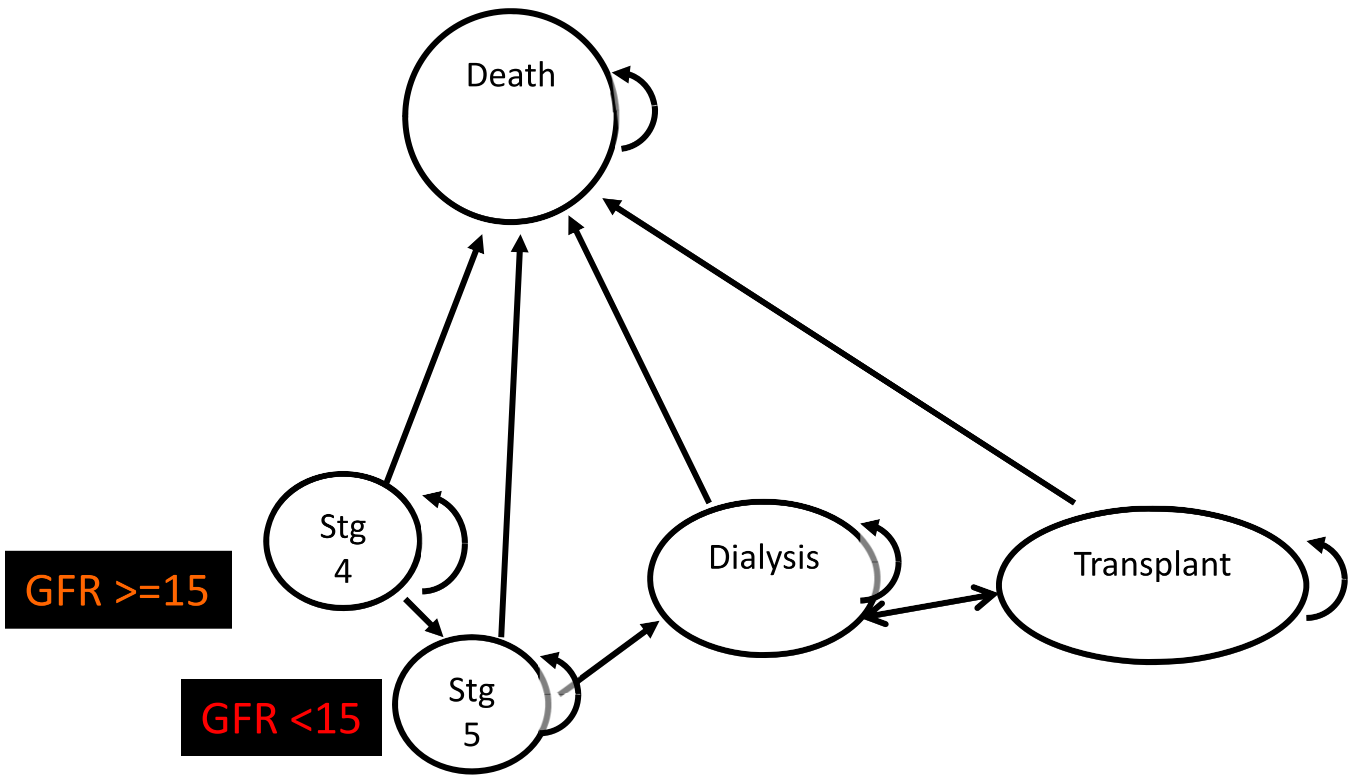
- The objective is to model a cohort with chronic kidney disease (CKD)
- Severity of CKD is defined by the value of the glomerular filtration rate (GFR)
- The lower the GFR, the worse the kidney function
- Normal ~ 100 ml/min, at < 10 we consider dialysis
- CKD progresses through stages:



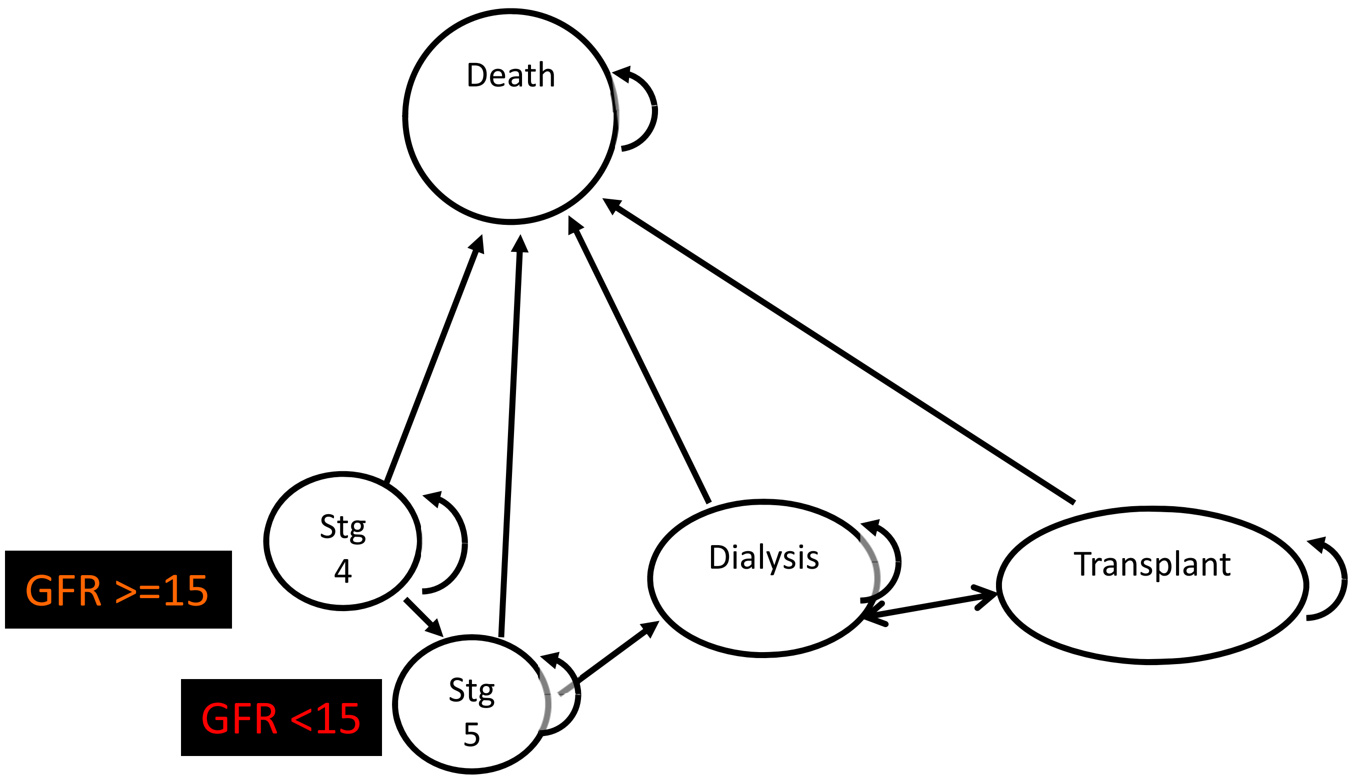
Death
GFR > 60
Stg 2
Stg 3
GFR 30 - 59
Stg 4
GFR 15 - 29
Stg 5
GFR <15
ESRD
End-stage renal Disease (dialysis or Tx)



For now,
let's consider only late stage CKD:
Stg4, stg5, ESRD and death

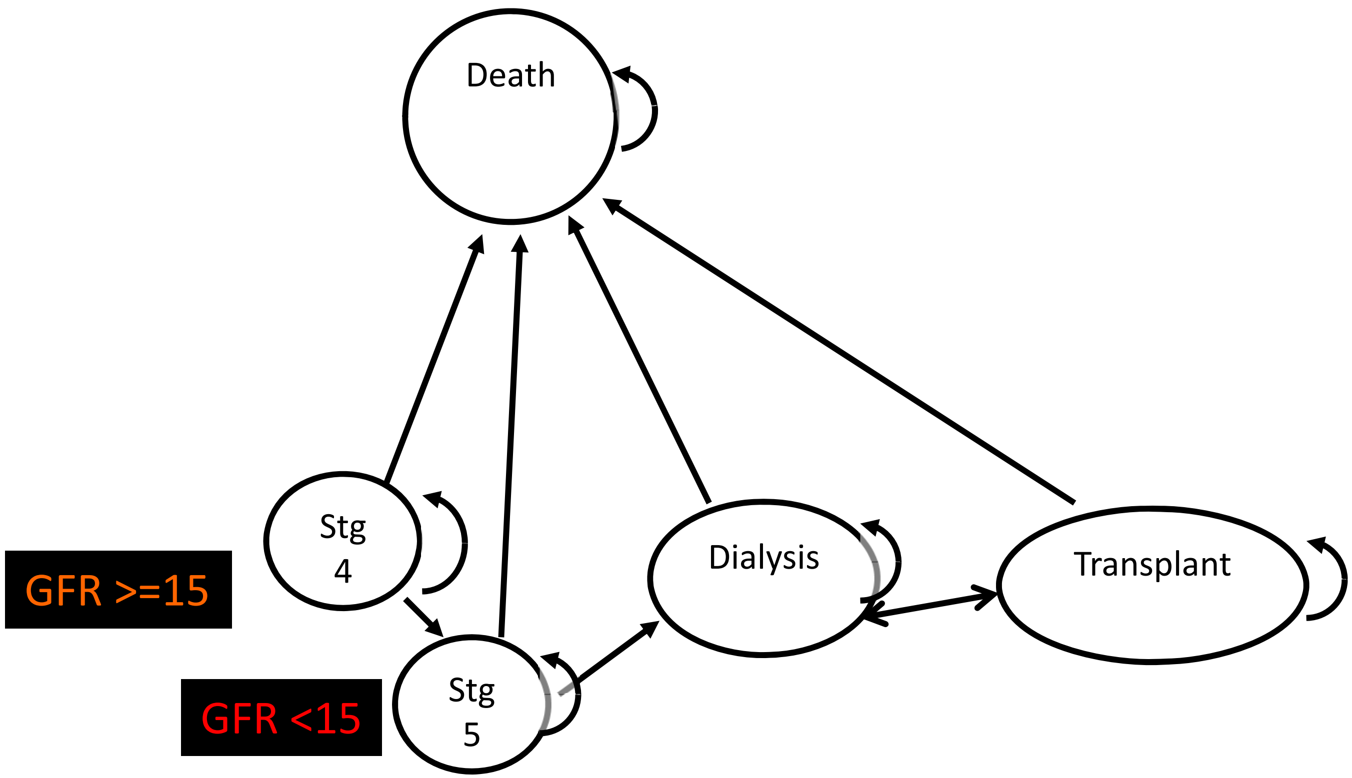


Let's split ESRD into dialysis and renal transplant states.
What does this state structure and allowed transitions imply?

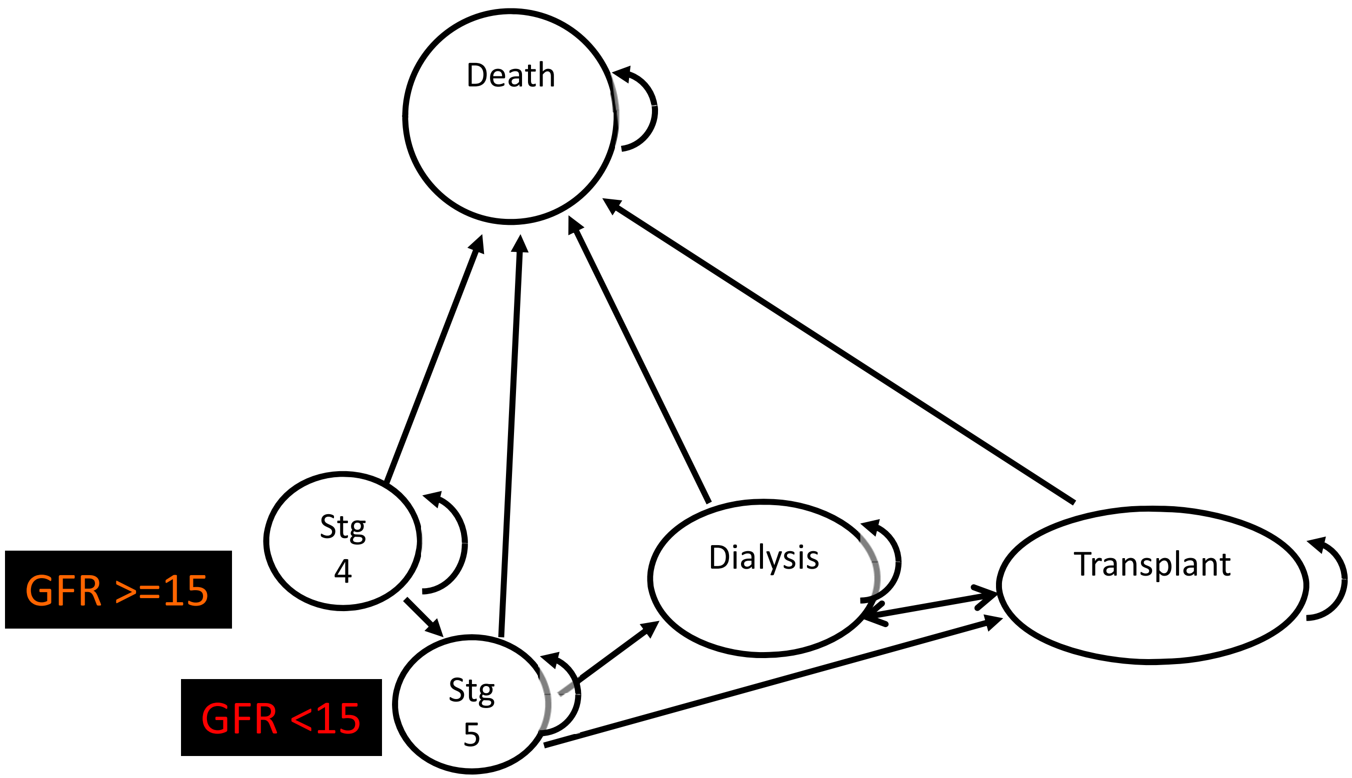


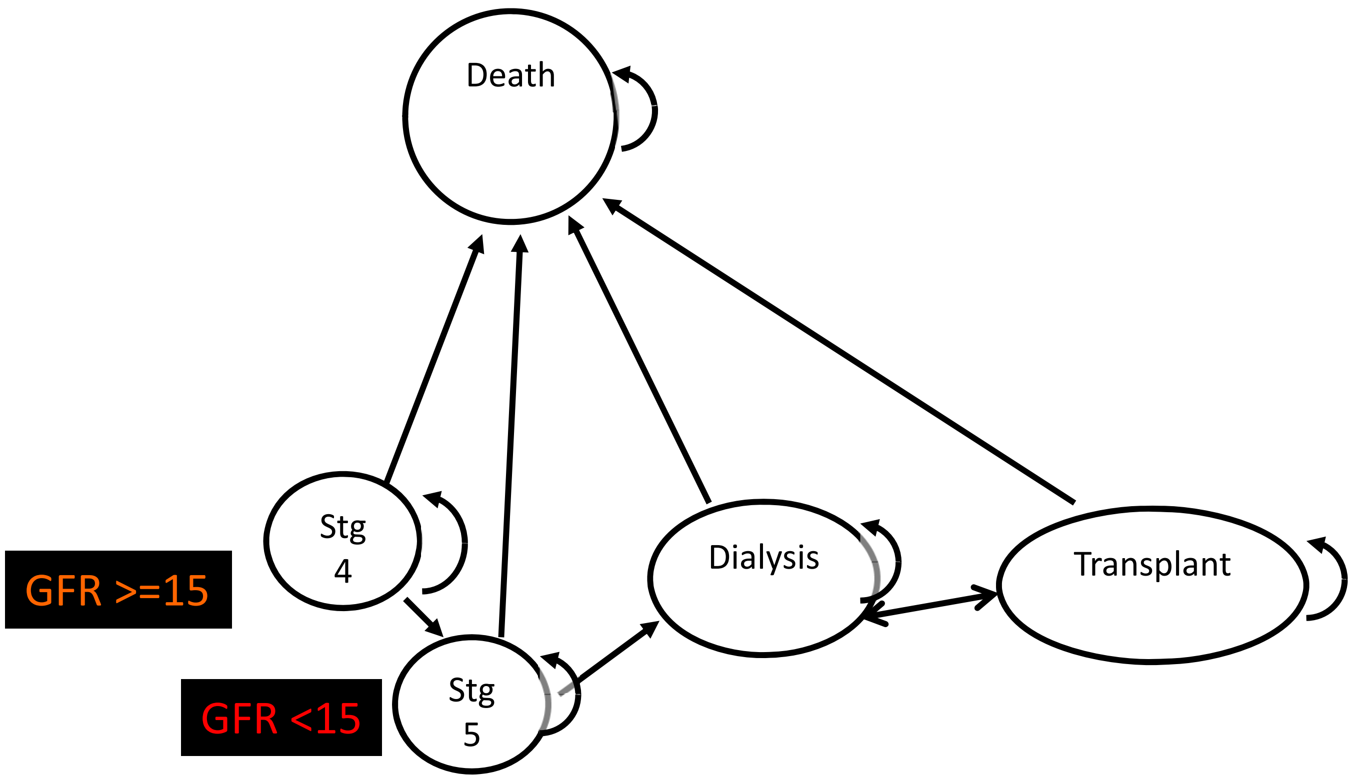
Let’s split ESRD into dialysis and renal transplant states.
What does this state structure and allowed transitions imply?

This is an unfair question for non-nephrologists

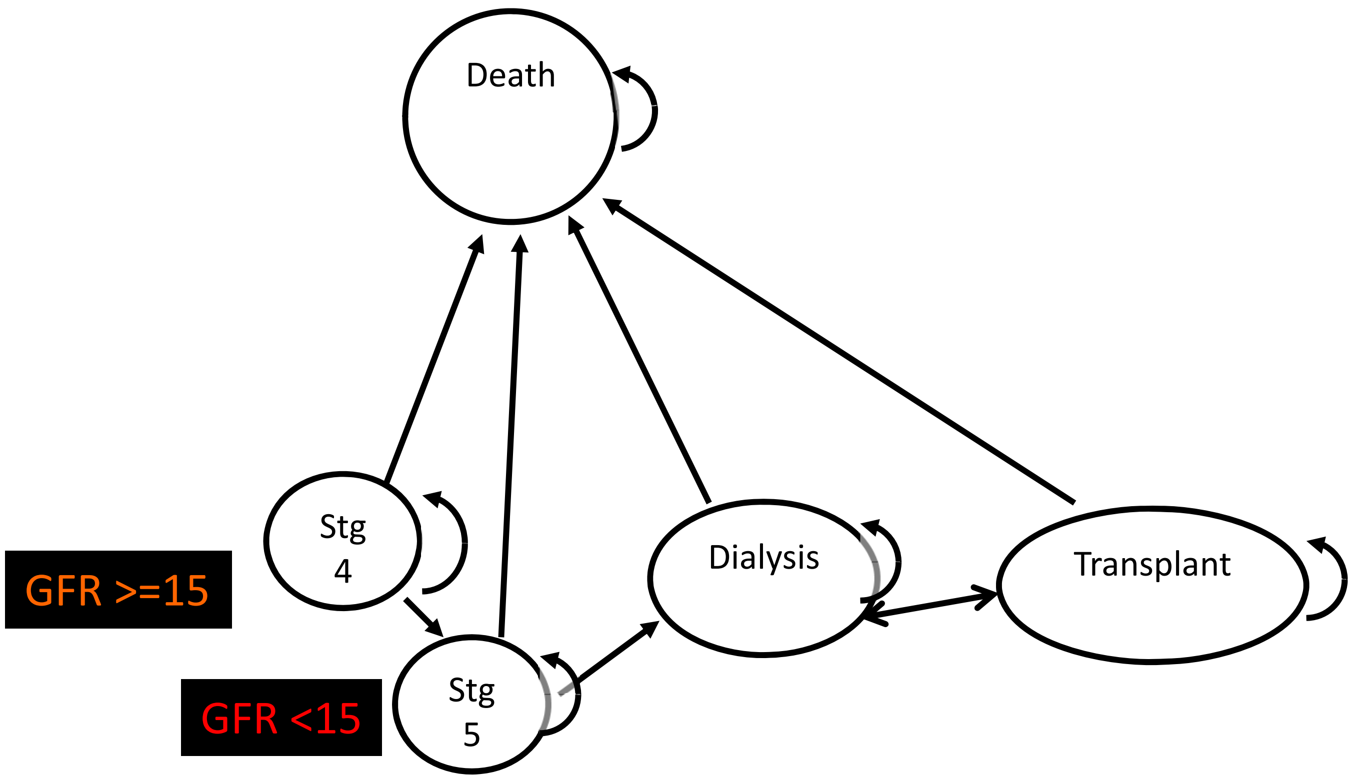


This structure does not allow for pre-emptive transplantation.
How could it be adjusted to do so?

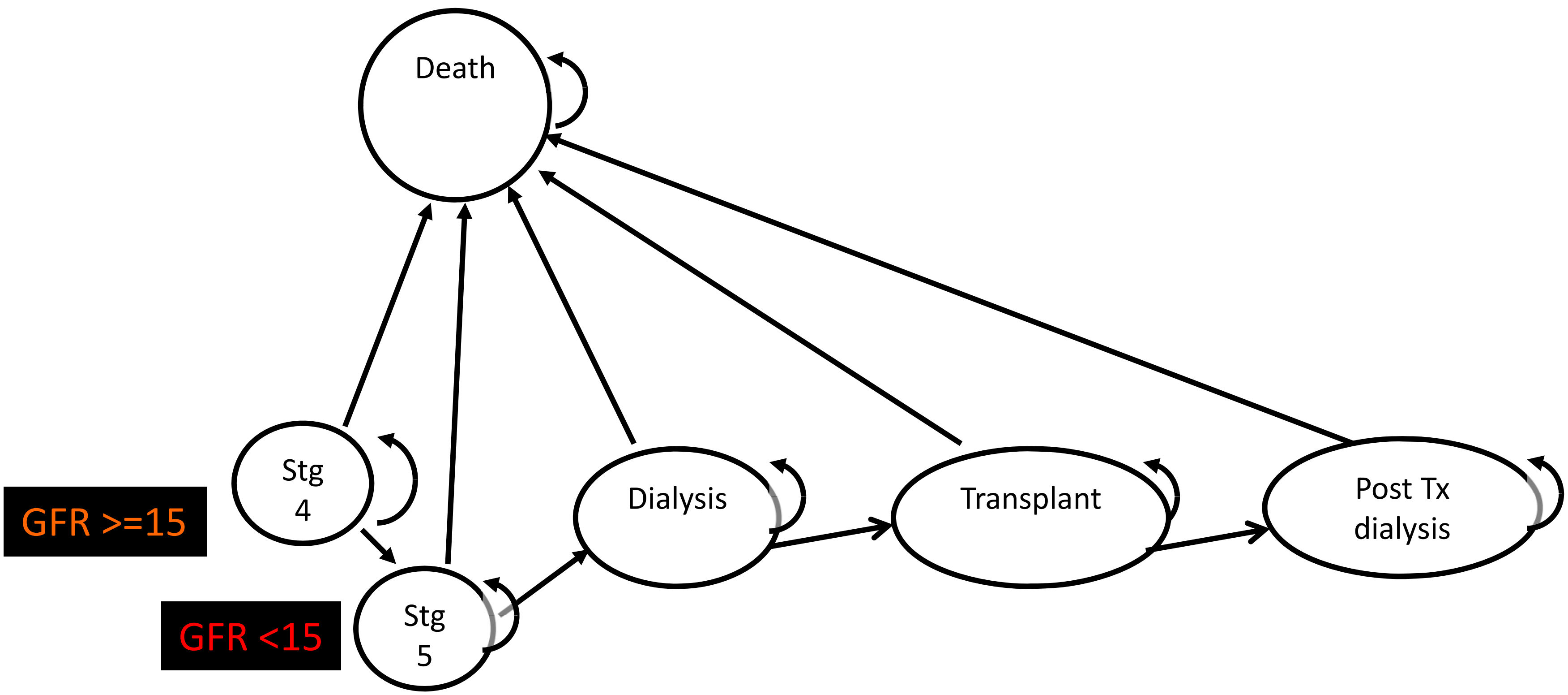

Death
Stg 4
GFR >=15
Stg 5
GFR <15
Dialysis
Transplant



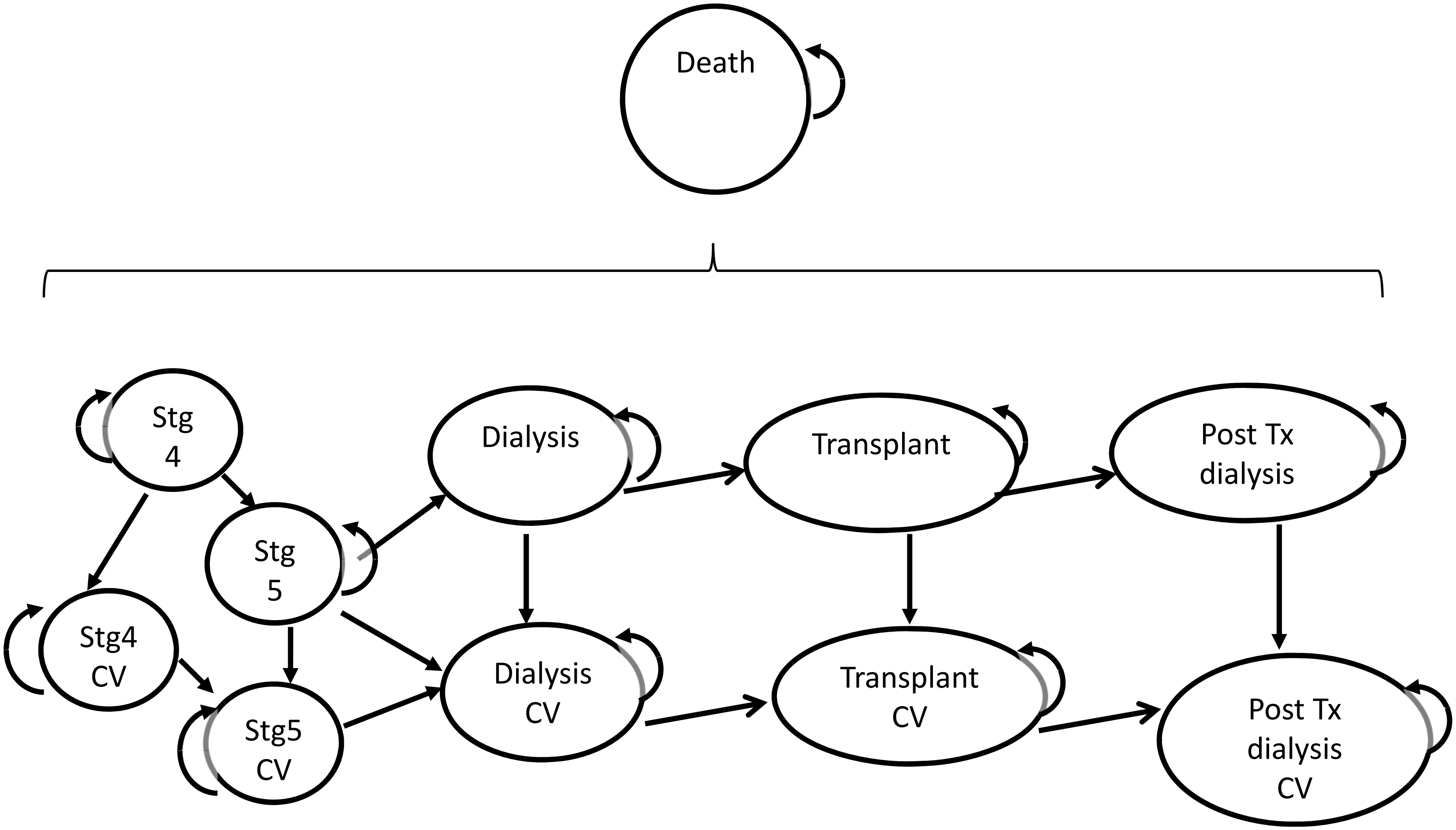
For now, let's NOT consider pre-emptive transplantation.
But……, how many transplants could happen?



How should this be structured so that only ONE transplant can happen?
(and respects the Markovian assumption)



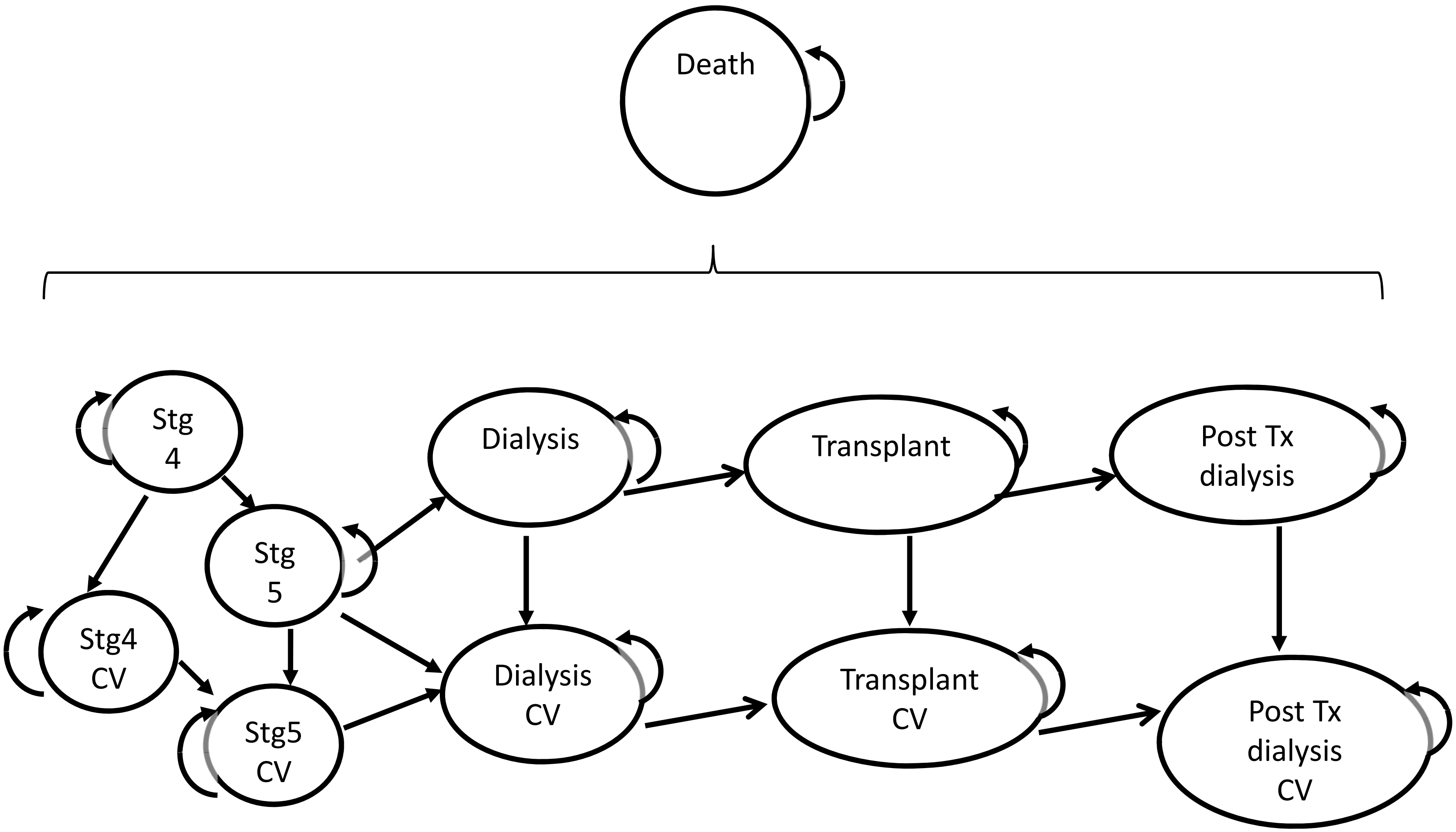
How should this be structured so that cardiovascular events are considered?
(and respects the Markovian assumption)



Structured so that cardiovascular events are considered
(respecting the Markovian assumption)

In general, when are additional states required?

- When costs or incremental utilities differ
- When subsequent disease evolution differs
- In order to ‘remember’ prior history



Structured so that cardiovascular events are considered
(respecting the Markovian assumption)

Are additional states required for CV events?

- Do costs or incremental utilities differ?
  - Yes
- Does subsequent disease evolution differ?
  - Yes
- Do we ‘remember’ prior history?
  - Yes, being in a ‘X CV’ state implies that there must have been a prior CV event

Distinguishing between states and short-term, transient events

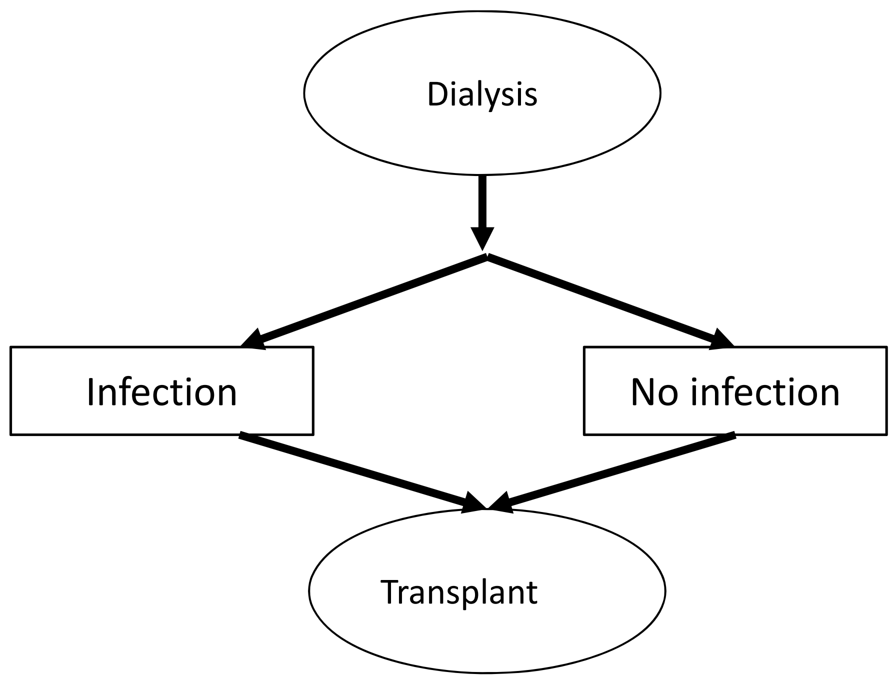
- Renal transplant recipients may have an infection immediately after the transplant procedure lasting 1 – 2 days (cycle length = 1 month)
- Imagine that the latter does not effect the ultimate function of the transplant
- Should infection be modeled as a state?

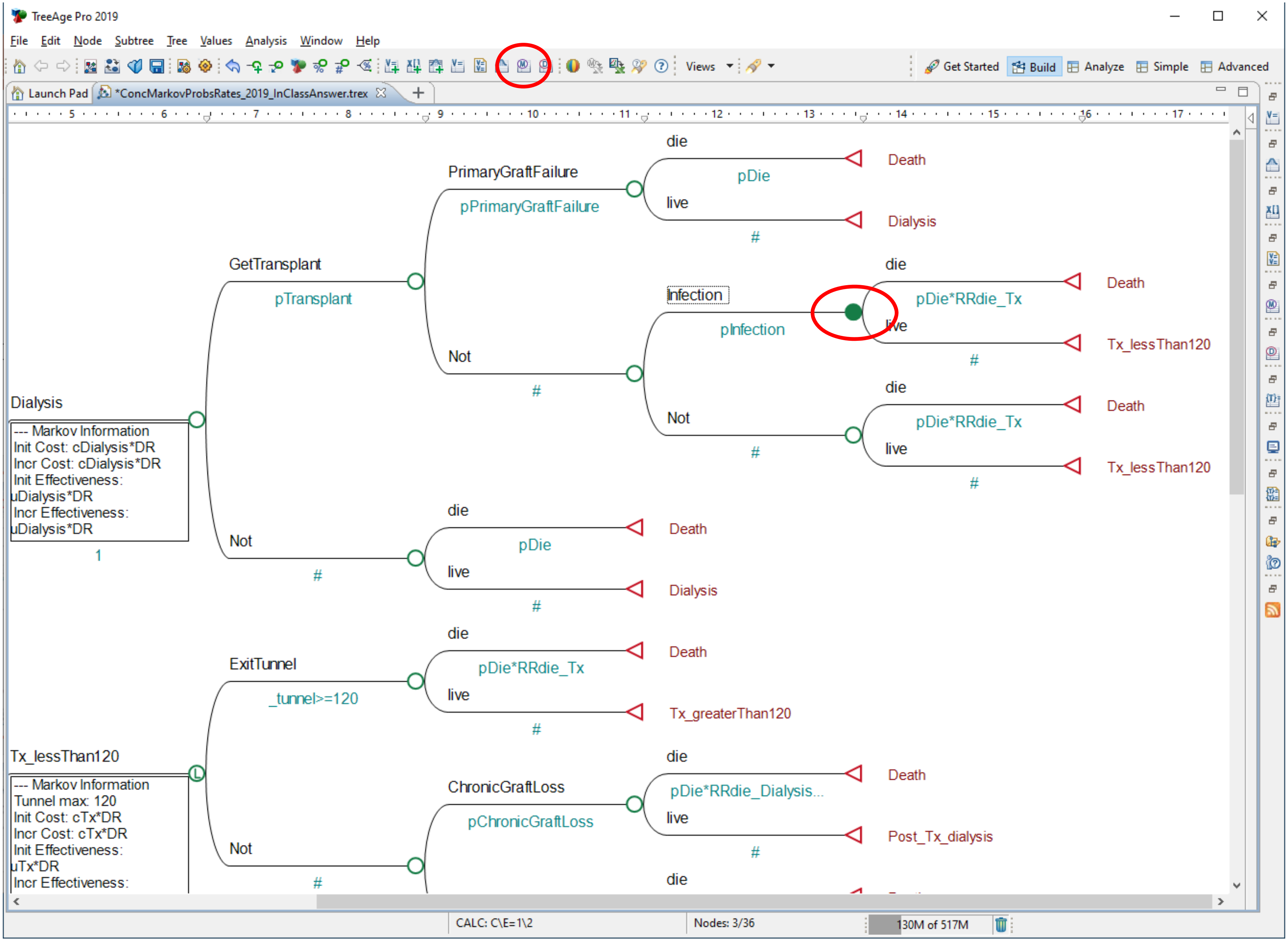
Are additional states required for post Tx infection?

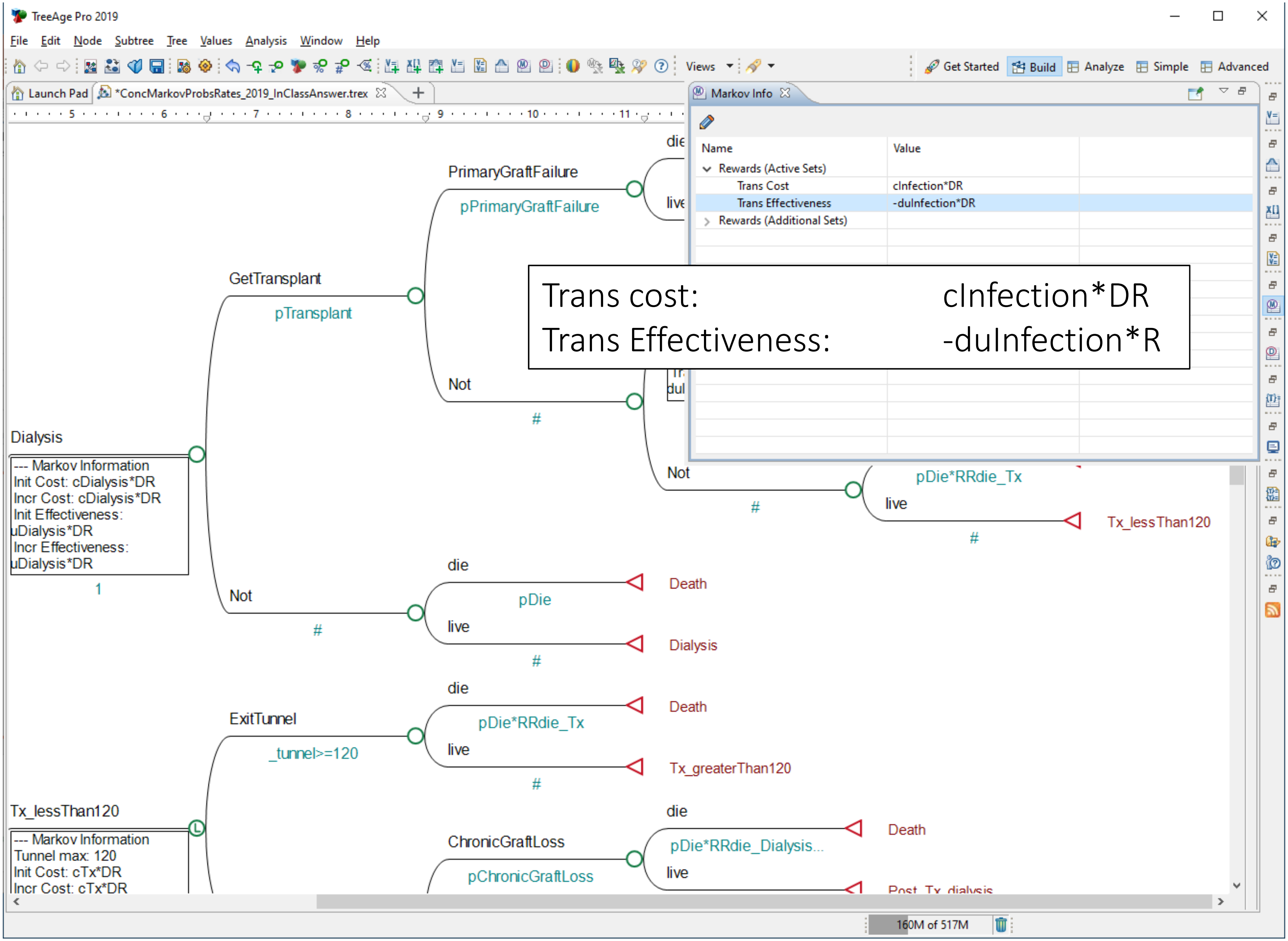
- Do costs or incremental utilities differ?
  - Yes, but only *transiently*, not permanently
- Does subsequent disease evolution differ?
  - No
- Do we need to 'remember' prior history?
  - No

Transient events

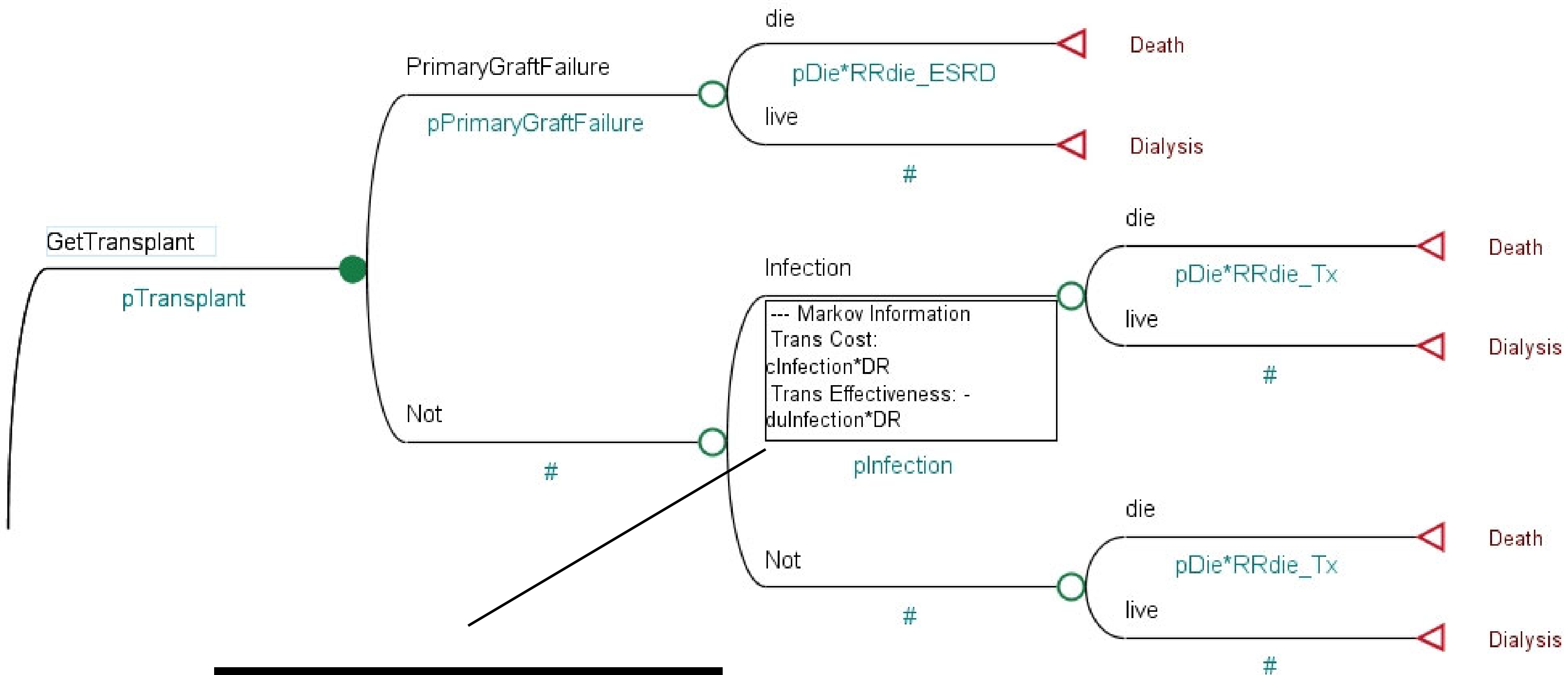
- If
  - Final and end states identical
  - Duration < 1 cycle length
- Then
  - Capture effects as transition rewards (tolls)



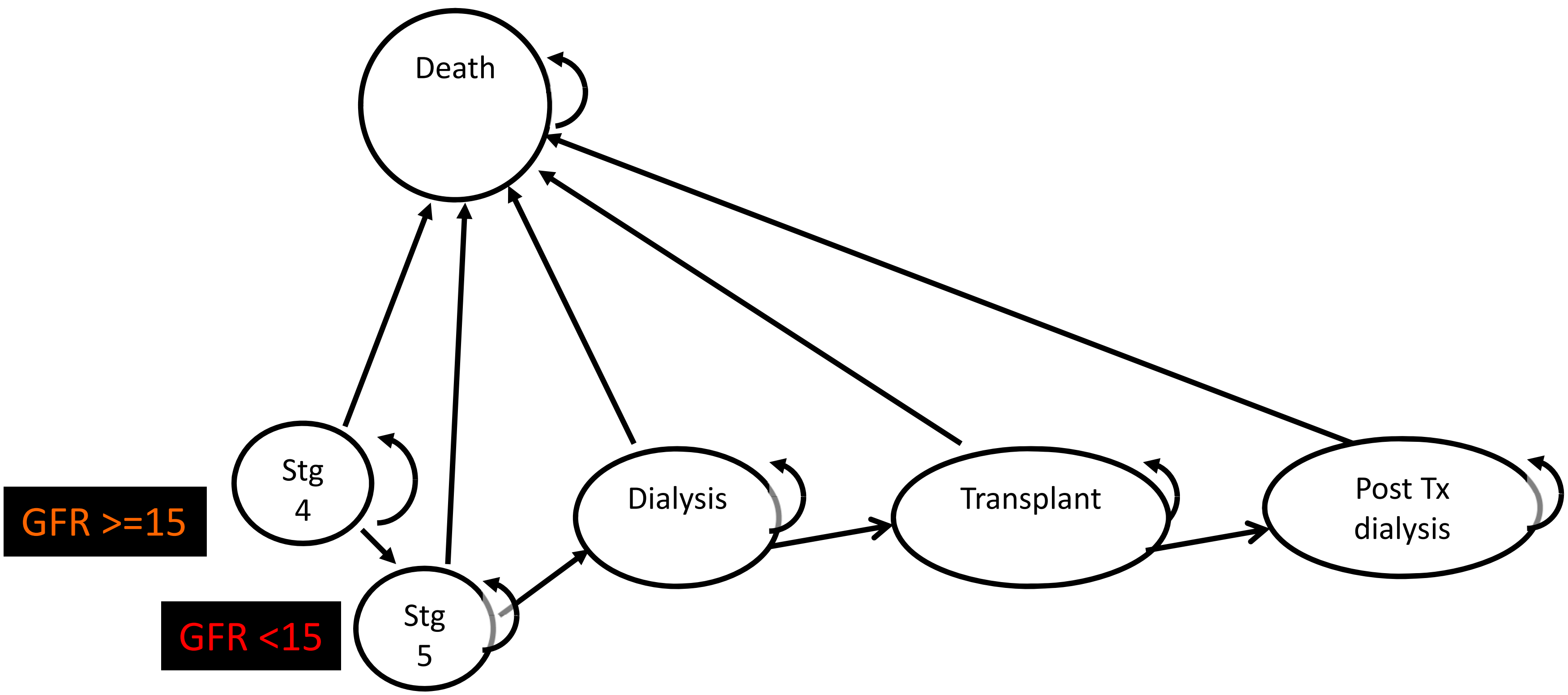

TreeAge Pro 2019
File Edit Node Subtree Tree Values Analysis Window Help
Views
Get Started Build Analyze Simple Advanced
Launch Pad
*ConcMarkovProbsRates_2019_InClassAnswer.trex
Dialysis
--- Markov Information
Init Cost: cDialysis*DR
Incr Cost: cDialysis*DR
Init Effectiveness: uDialysis*DR
Incr Effectiveness: uDialysis*DR
1
GetTransplant
pTransplant
PrimaryGraftFailure
pPrimaryGraftFailure
die
pDie
live
#
Death
Dialysis
Not
#
Infection
pInfection
die
pDie*RRdie_Tx
live
#
Death
Tx_lessThan120
Not
#
die
pDie*RRdie_Tx
live
#
Death
Tx_lessThan120
Not
#
die
pDie
live
#
Death
Dialysis
Tx_lessThan120
--- Markov Information
Tunnel max: 120
Init Cost: cTx*DR
Incr Cost: cTx*DR
Init Effectiveness: uTx*DR
Incr Effectiveness:
ExitTunnel
_tunnel>=120
die
pDie*RRdie_Tx
live
#
Death
Tx_greaterThan120
Not
#
ChronicGraftLoss
pChronicGraftLoss
die
pDie*RRdie_Dialysis...
live
#
Death
Post_Tx_dialysis
die
CALC: C\E=1\2
Nodes: 3/36
130M of 517M



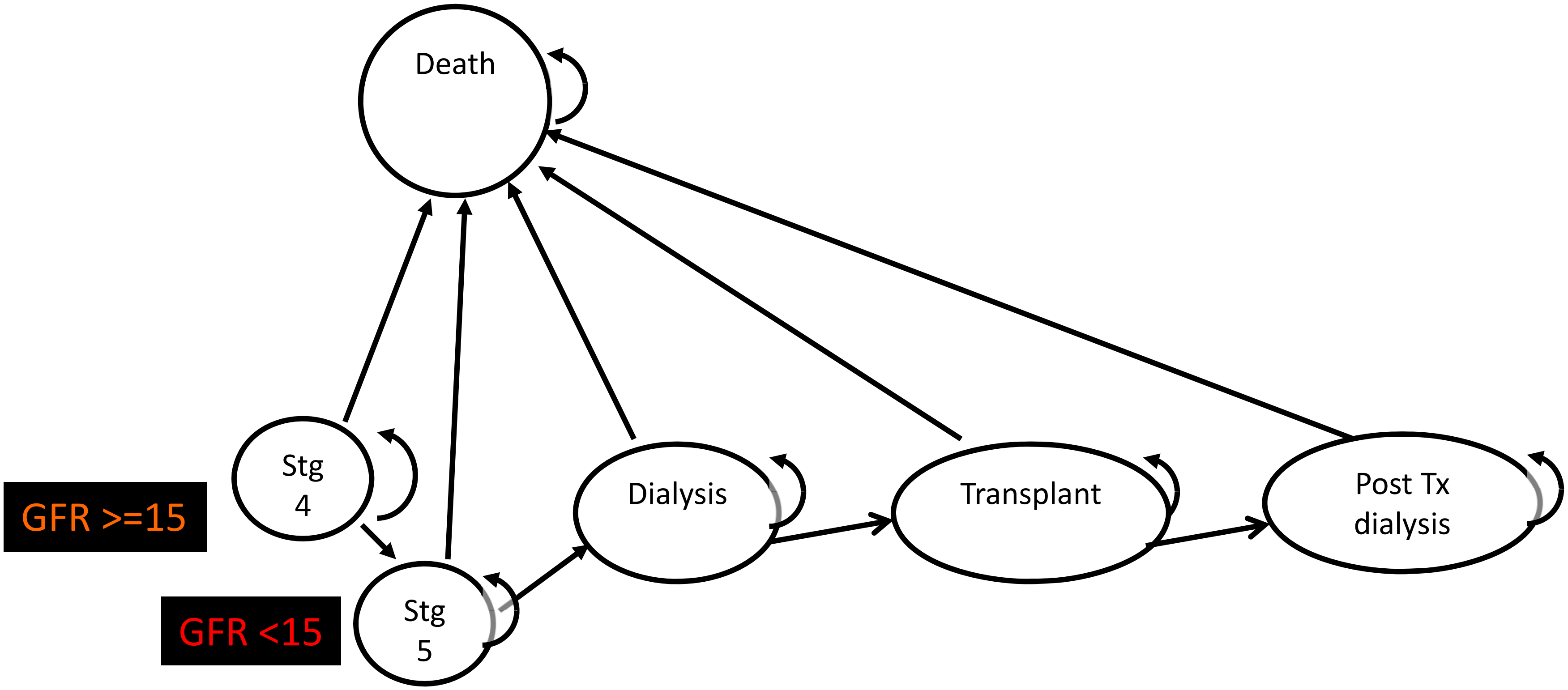
Trans cost: cInfection*DR
Trans Effectiveness: -duInfection*R
TreeAge Pro 2019
File Edit Node Subtree Tree Values Analysis Window Help
Views
Get Started
Build
Analyze
Simple
Advanced
Launch Pad
*ConcMarkovProbsRates_2019_InClassAnswer.trex
Markov Info
Name
Value
Rewards (Active Sets)
Trans Cost
cInfection*DR
Trans Effectiveness
-duInfection*DR
Rewards (Additional Sets)
Dialysis
--- Markov Information
Init Cost: cDialysis*DR
Incr Cost: cDialysis*DR
Init Effectiveness: uDialysis*DR
Incr Effectiveness: uDialysis*DR
1
GetTransplant
pTransplant
PrimaryGraftFailure
pPrimaryGraftFailure
die
live
Not
#
pDie*RRdie_Tx
Tx_lessThan120
pDie
Death
Dialysis
ExitTunnel
_tunnel>=120
Tx_greaterThan120
Tx_lessThan120
--- Markov Information
Tunnel max: 120
Init Cost: cTx*DR
Incr Cost: cTx*DR
ChronicGraftLoss
pChronicGraftLoss
pDie*RRdie_Dialysis...
Post_Tx_dialysis
160M of 517M



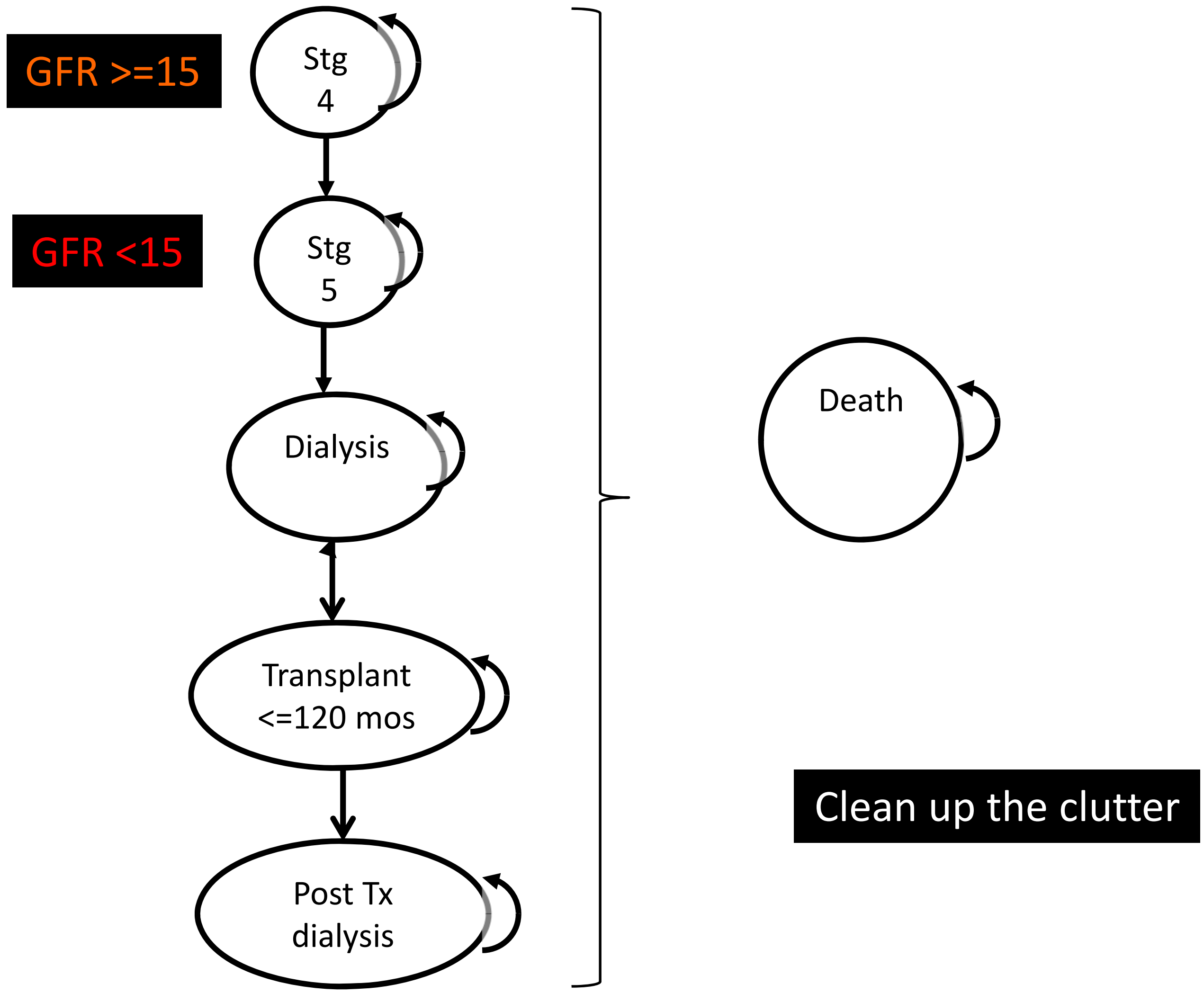
Transition rewards

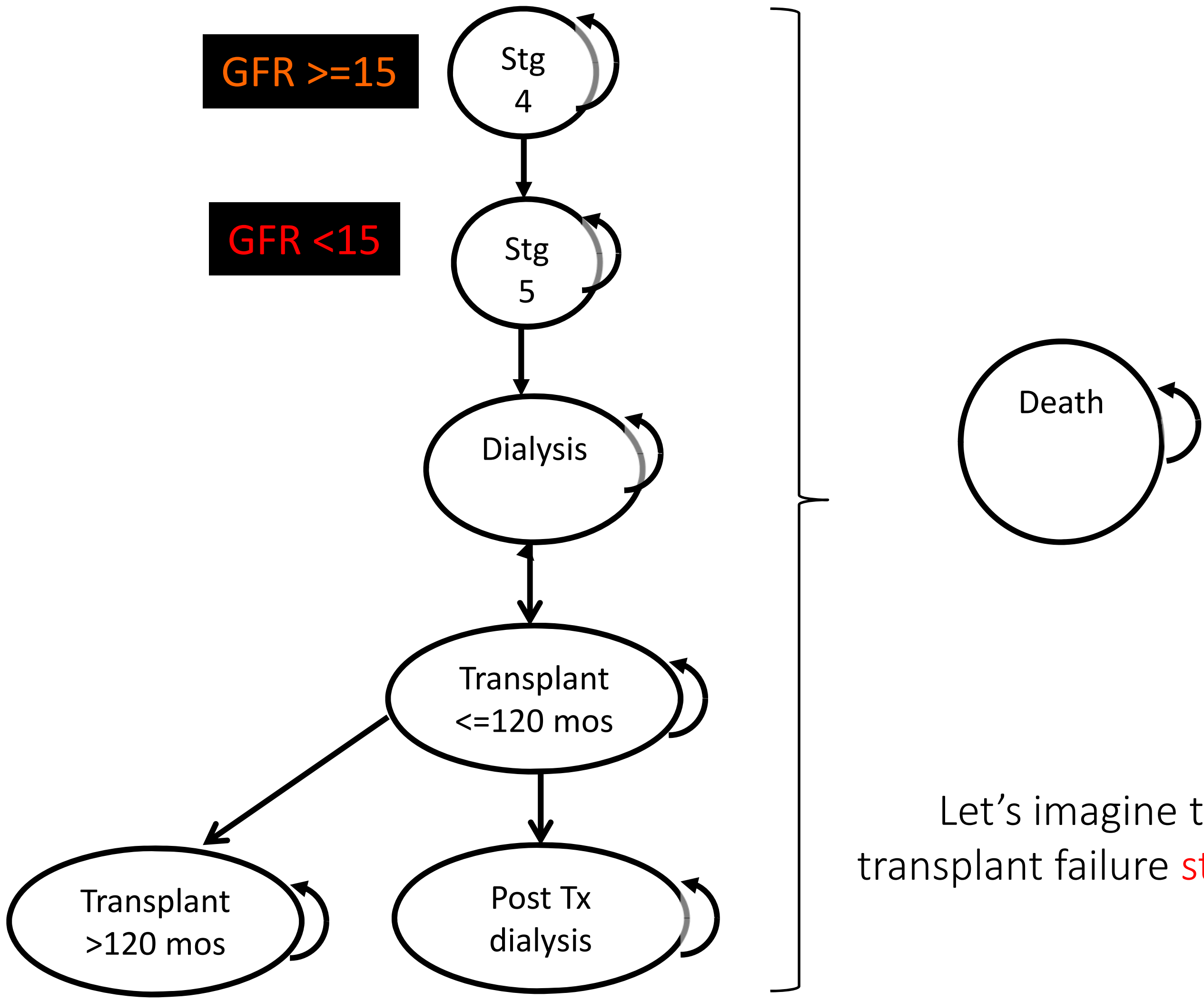


Our state structure and allowed transitions thus far.
For simplicity,
let's ignore cardiovascular events

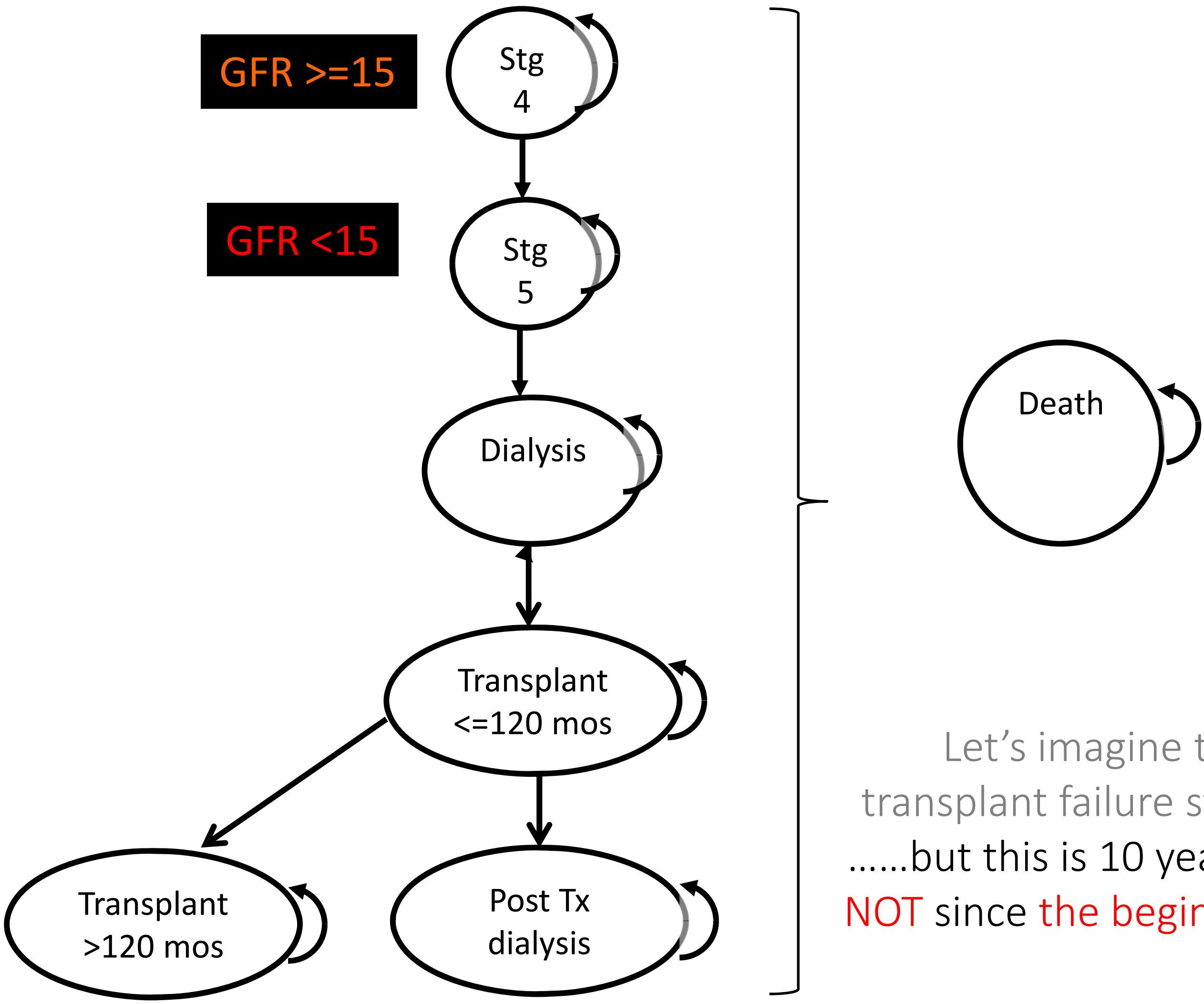


Suppose there is an early phase, post transplant, where there is a possibility of transplant failure, and a late phase where failure is impossible


GFR >=15
GFR <15
Stg
4
Stg
5
Dialysis
Transplant
<=120 mos
Post Tx
dialysis
Death
Clean up the clutter



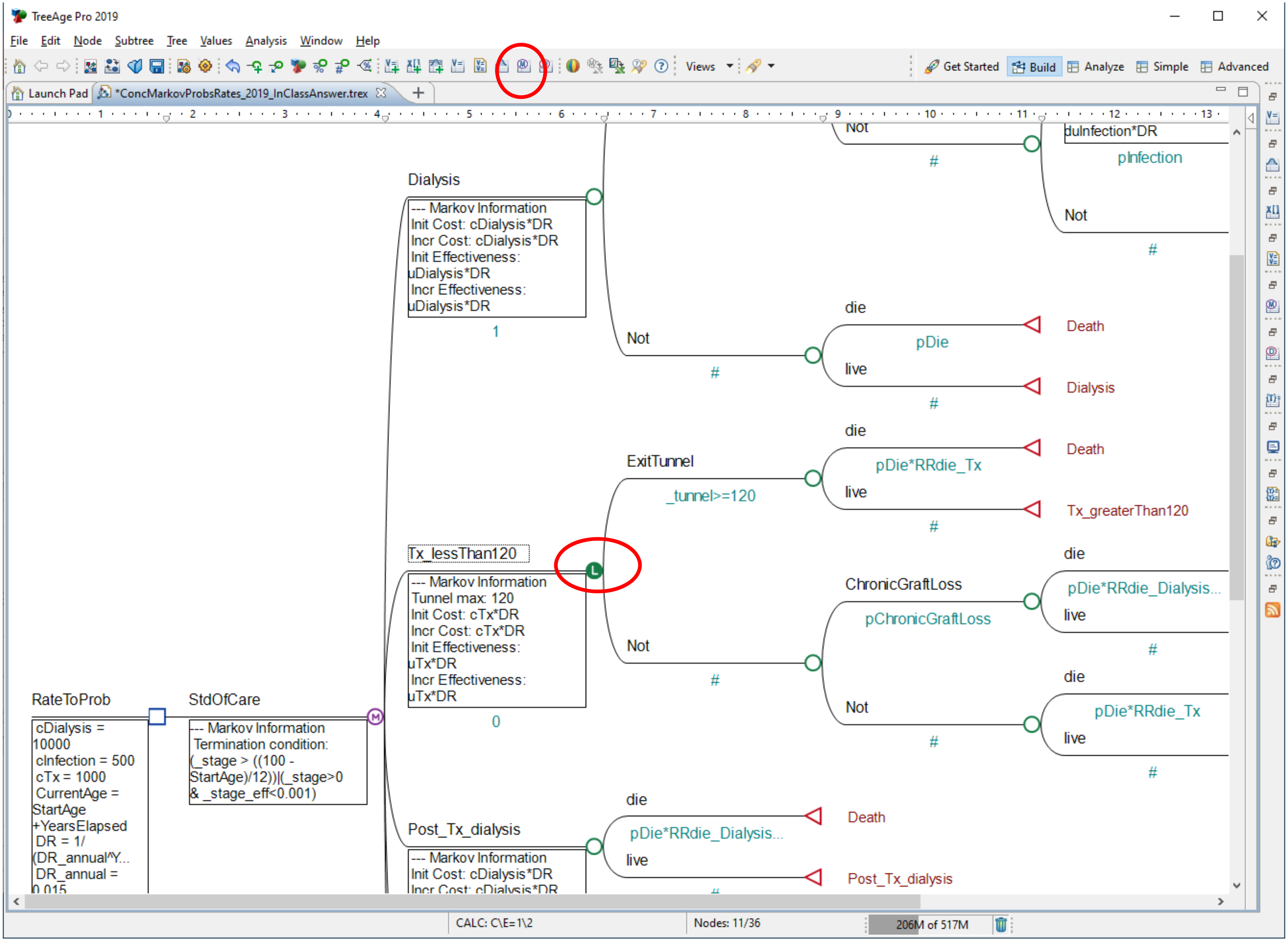
Let's imagine that the risk of transplant failure stops after 10 years



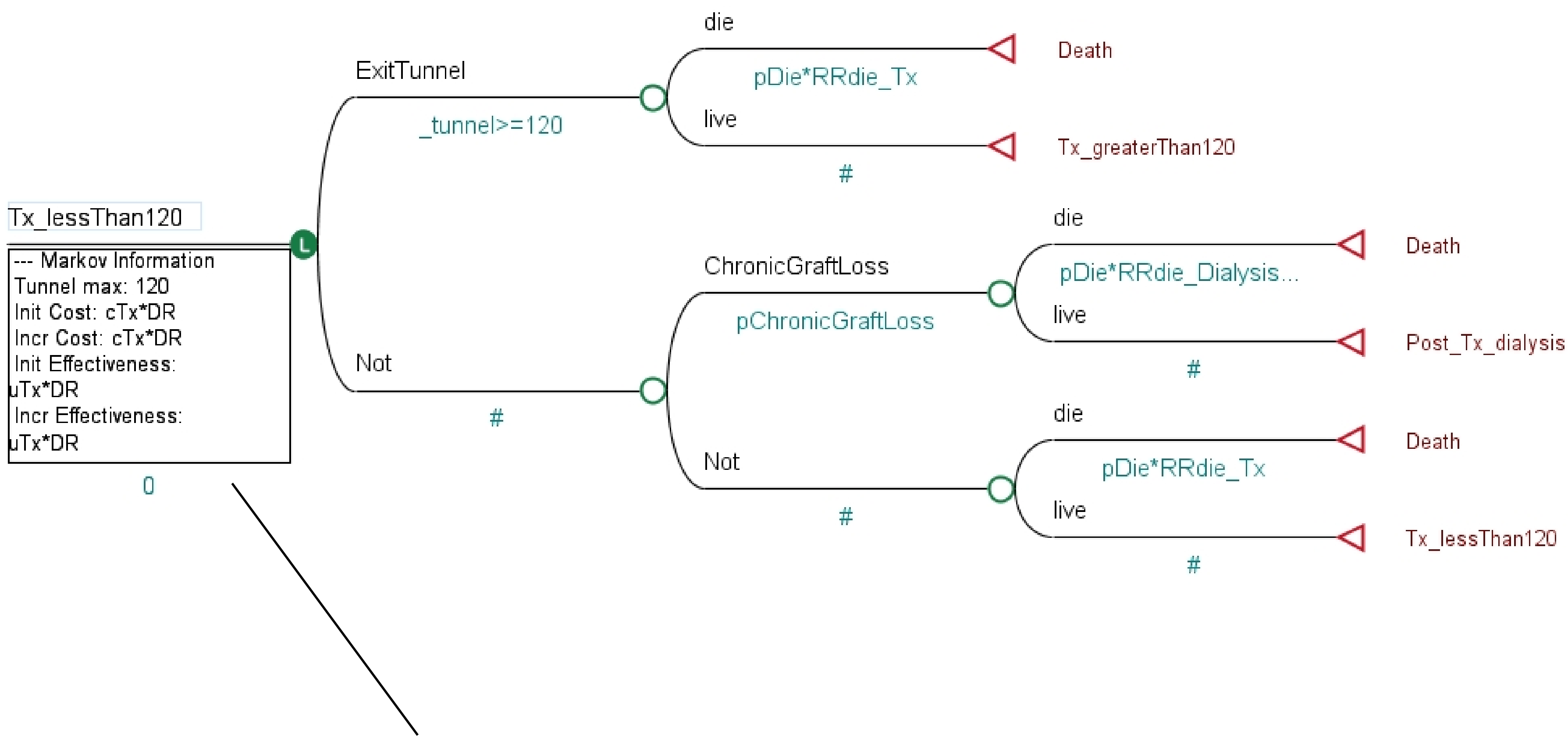
Let's imagine that the risk of transplant failure stops after 10 years ……but this is 10 years after transplant, NOT since the beginning of the process

The “two clock” problem

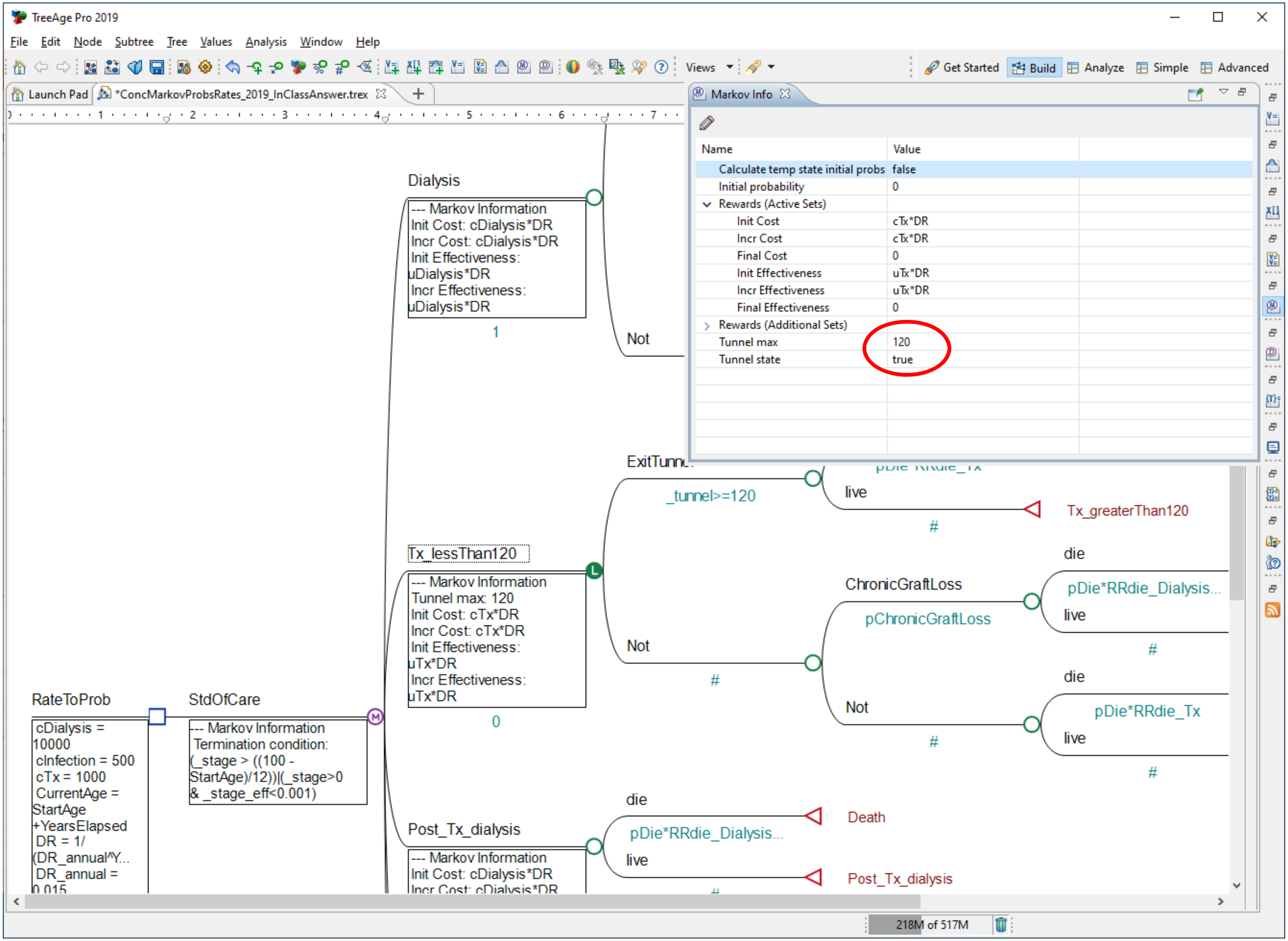
- Because of the Markovian assumption there is no explicit memory of when past events occured
- But, often want to base the probability of an event on the time since the occurrence of an interim event, not since the beginning of the process
- E.g. the probability of chronic rejection and transplant failure is a function of time since transplant, not since the start of the simulation
- Use tunnel states and the _tunnel counter



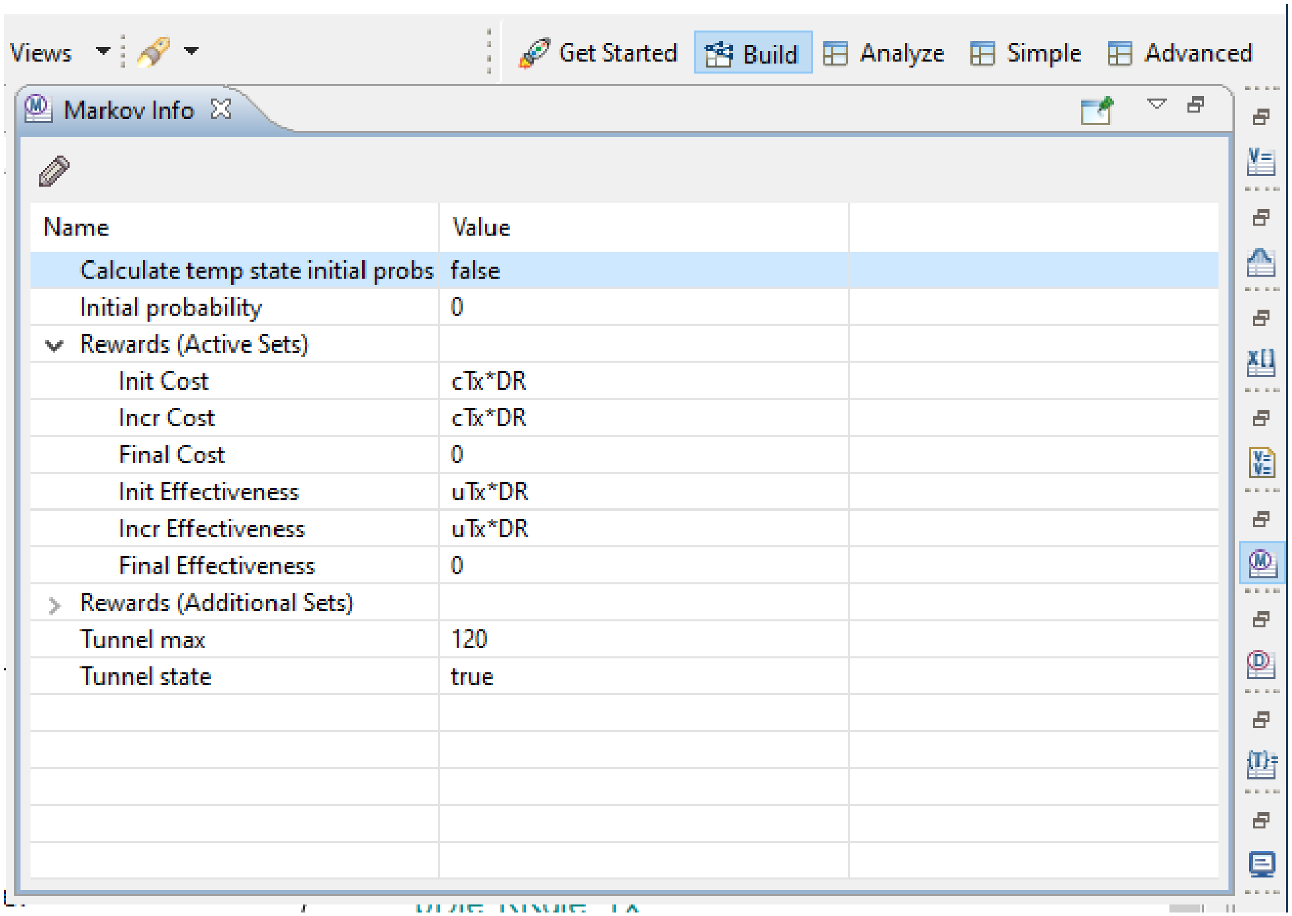
TreeAge Pro 2019
File Edit Node Subtree Tree Values Analysis Window Help
Views
Get Started Build Analyze Simple Advanced
Launch Pad
*ConcMarkovProbsRates_2019_InClassAnswer.trex
Dialysis
--- Markov Information
Init Cost: cDialysis*DR
Incr Cost: cDialysis*DR
Init Effectiveness: uDialysis*DR
Incr Effectiveness: uDialysis*DR
1
Not
dulnfection*DR
pInfection
Not
die
pDie
live
Death
Dialysis
ExitTunnel
_tunnel>=120
die
pDie*RRdie_Tx
live
Death
Tx_greaterThan120
Tx_lessThan120
--- Markov Information
Tunnel max: 120
Init Cost: cTx*DR
Incr Cost: cTx*DR
Init Effectiveness: uTx*DR
Incr Effectiveness: uTx*DR
0
Not
ChronicGraftLoss
pChronicGraftLoss
die
pDie*RRdie_Dialysis...
live
Not
die
pDie*RRdie_Tx
live
RateToProb
cDialysis = 10000
cInfection = 500
cTx = 1000
CurrentAge = StartAge +YearsElapsed
DR = 1/(DR_annual^Y...
DR_annual = 0.015
StdOfCare
--- Markov Information
Termination condition:
(_stage > ((100 - StartAge)/12))|(_stage>0 & _stage_eff<0.001)
Post_Tx_dialysis
--- Markov Information
Init Cost: cDialysis*DR
Incr Cost: cDialysis*DR
die
pDie*RRdie_Dialysis...
live
Death
Post_Tx_dialysis
CALC: C\E=1\2
Nodes: 11/36
206M of 517M

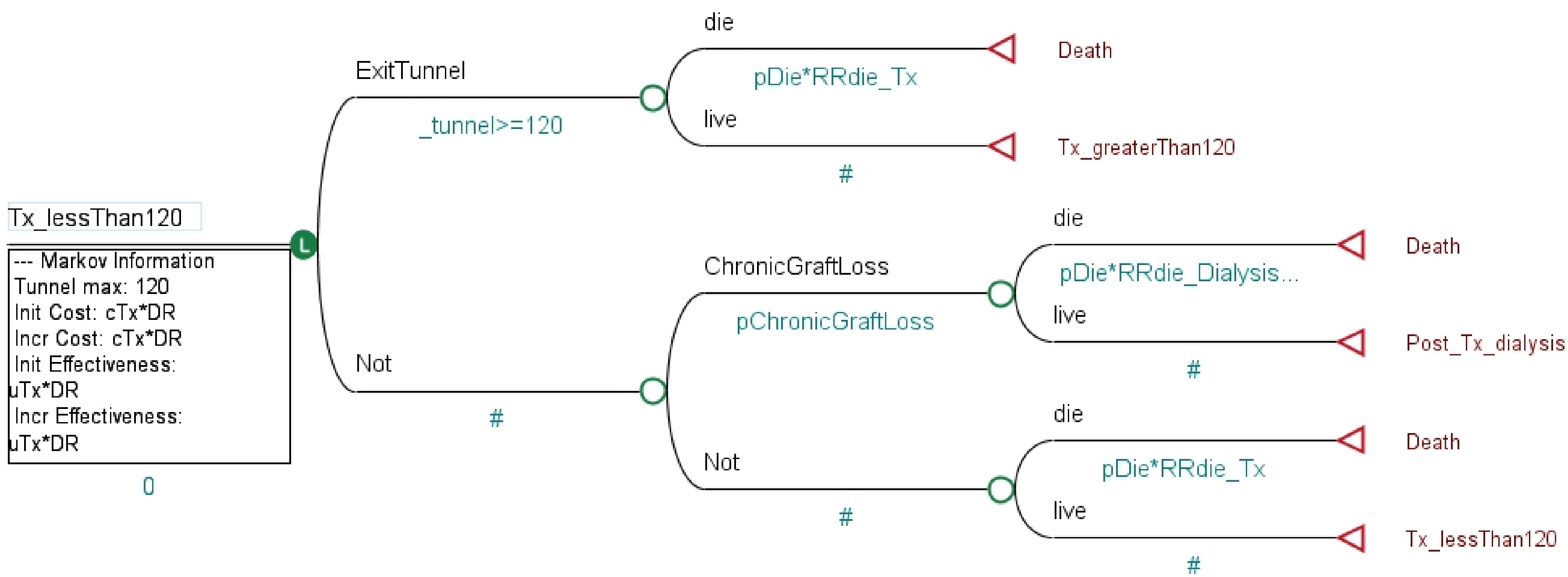


Set the max tunnel value to a number > 0 in the Markov dialogue box



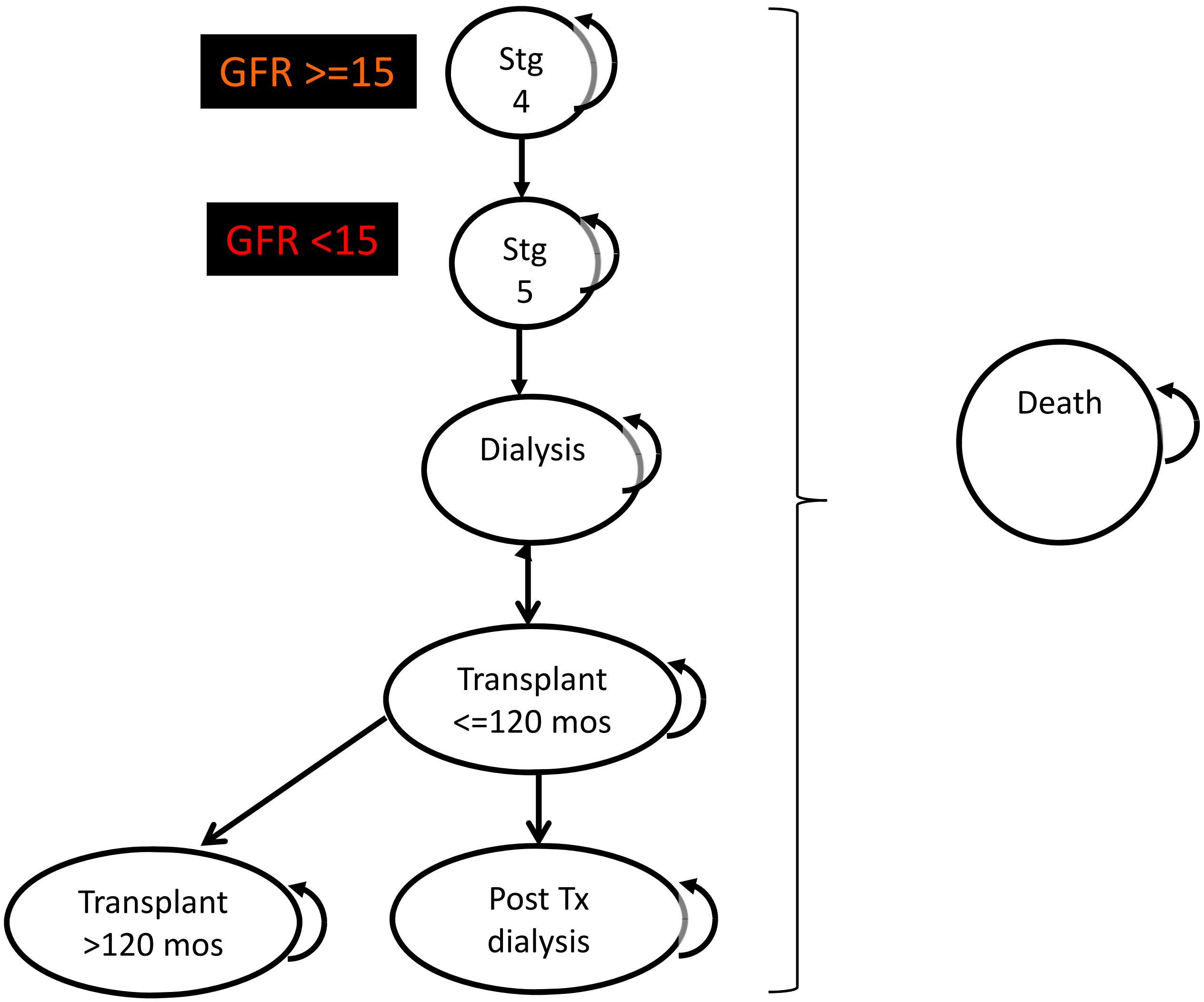
TreeAge Pro 2019
File Edit Node Subtree Tree Values Analysis Window Help
Views
Get Started Build Analyze Simple Advanced
Launch Pad
*ConcMarkovProbsRates_2019_InClassAnswer.trex
Markov Info
Name
Value
Calculate temp state initial probs
false
Initial probability
0
Rewards (Active Sets)
Init Cost
cTx*DR
Incr Cost
cTx*DR
Final Cost
0
Init Effectiveness
uTx*DR
Incr Effectiveness
uTx*DR
Final Effectiveness
0
Rewards (Additional Sets)
Tunnel max
120
Tunnel state
true
Dialysis
--- Markov Information
Init Cost: cDialysis*DR
Incr Cost: cDialysis*DR
Init Effectiveness:
uDialysis*DR
Incr Effectiveness:
uDialysis*DR
1
Not
ExitTunnel
_tunnel>=120
live
Tx_greaterThan120
#
Tx_lessThan120
--- Markov Information
Tunnel max: 120
Init Cost: cTx*DR
Incr Cost: cTx*DR
Init Effectiveness:
uTx*DR
Incr Effectiveness:
uTx*DR
0
die
ChronicGraftLoss
pDie*RRdie_Dialysis...
pChronicGraftLoss
live
Not
#
die
Not
pDie*RRdie_Tx
live
RateToProb
cDialysis =
10000
cInfection = 500
cTx = 1000
CurrentAge =
StartAge
+YearsElapsed
DR = 1/
(DR_annual^Y...
DR_annual =
0.015
StdOfCare
--- Markov Information
Termination condition:
(_stage > ((100 -
StartAge)/12))|(_stage>0
& _stage_eff<0.001)
die
pDie*RRdie_Dialysis...
Death
live
Post_Tx_dialysis
Post_Tx_dialysis
--- Markov Information
Init Cost: cDialysis*DR
218M of 517M

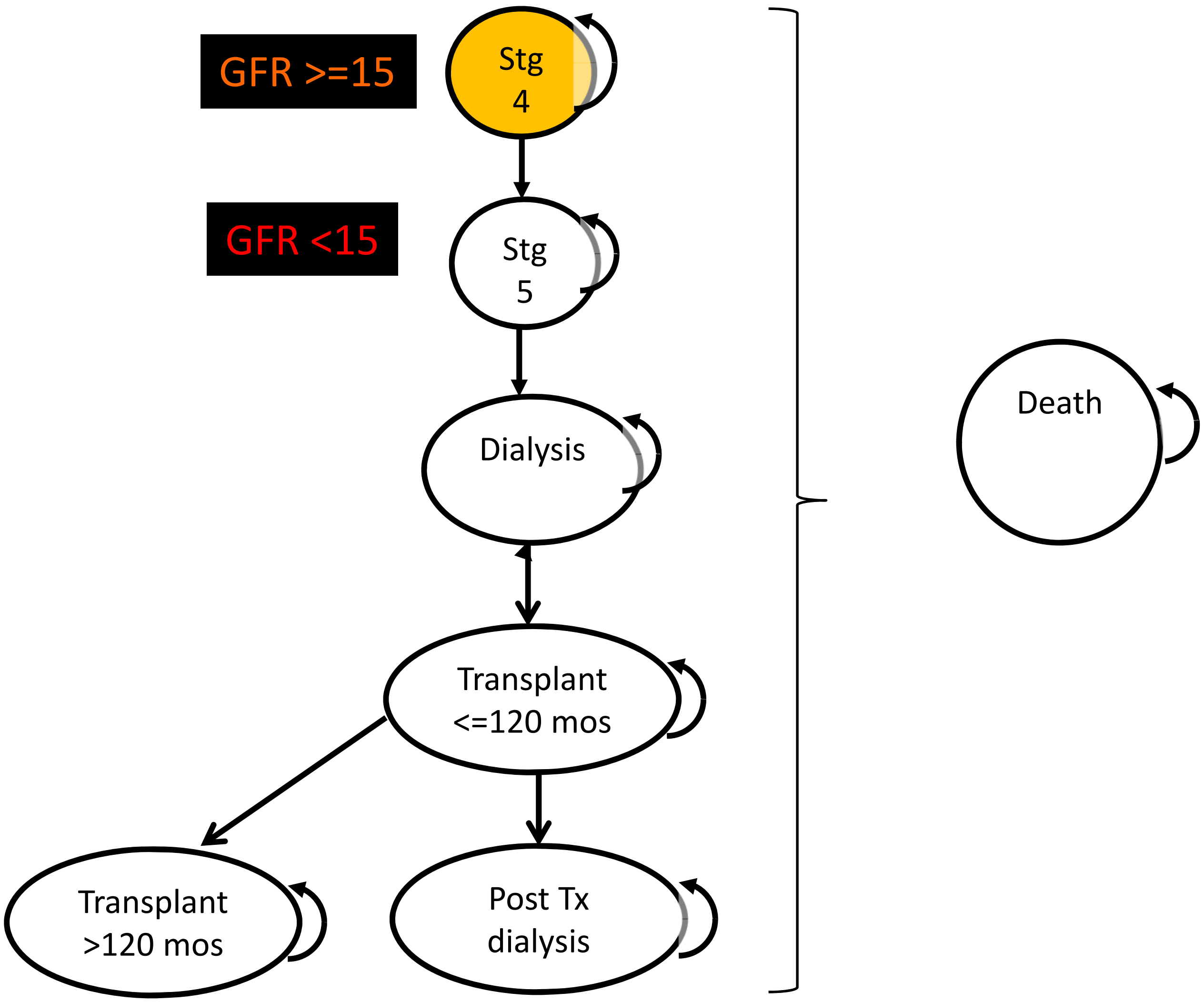

Views
Get Started
Build
Analyze
Simple
Advanced
Markov Info
Name
Value
Calculate temp state initial probs
false
Initial probability
0
Rewards (Active Sets)
Init Cost
cTx*DR
Incr Cost
cTx*DR
Final Cost
0
Init Effectiveness
uTx*DR
Incr Effectiveness
uTx*DR
Final Effectiveness
0
Rewards (Additional Sets)
Tunnel max
120
Tunnel state
true

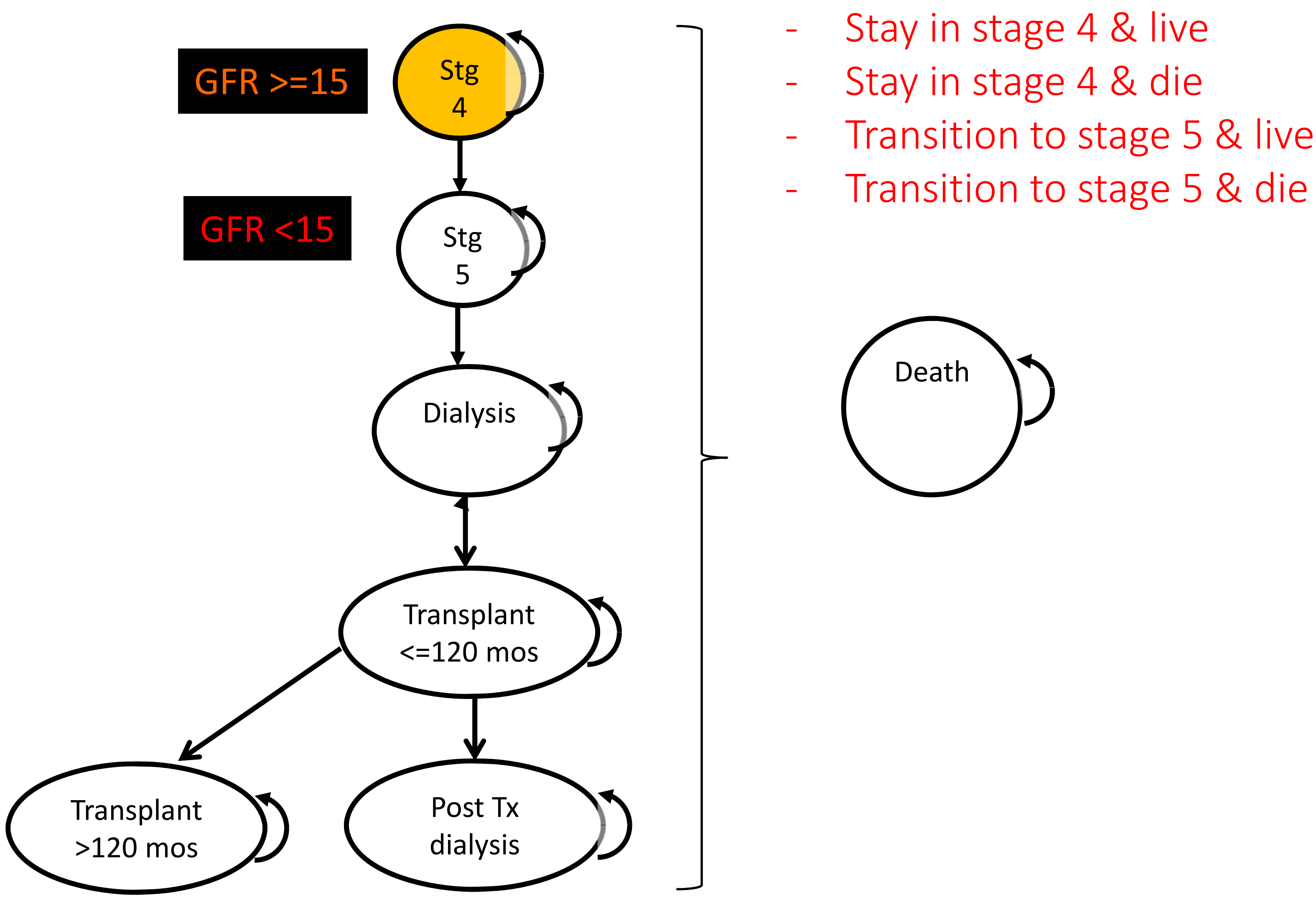

Tx_lessThan120
--- Markov Information
Tunnel max: 120
Init Cost: cTx*DR
Incr Cost: cTx*DR
Init Effectiveness:
uTx*DR
Incr Effectiveness:
uTx*DR
0
ExitTunnel
_tunnel>=120
Not
#
die
pDie*RRdie_Tx
live
#
Death
Tx_greaterThan120
ChronicGraftLoss
pChronicGraftLoss
Not
#
die
pDie*RRdie_Dialysis...
live
#
Death
Post_Tx_dialysis
die
pDie*RRdie_Tx
live
#
Death
Tx_lessThan120

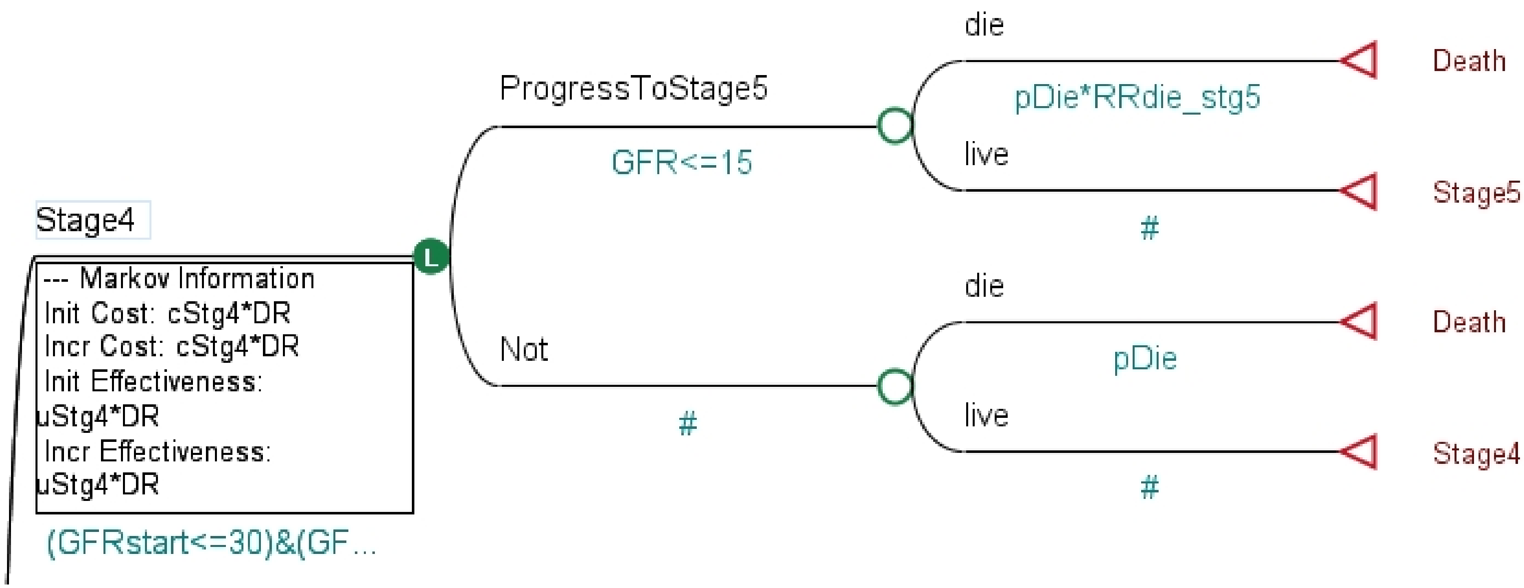
Steps to conceptualizing and building a Markov model

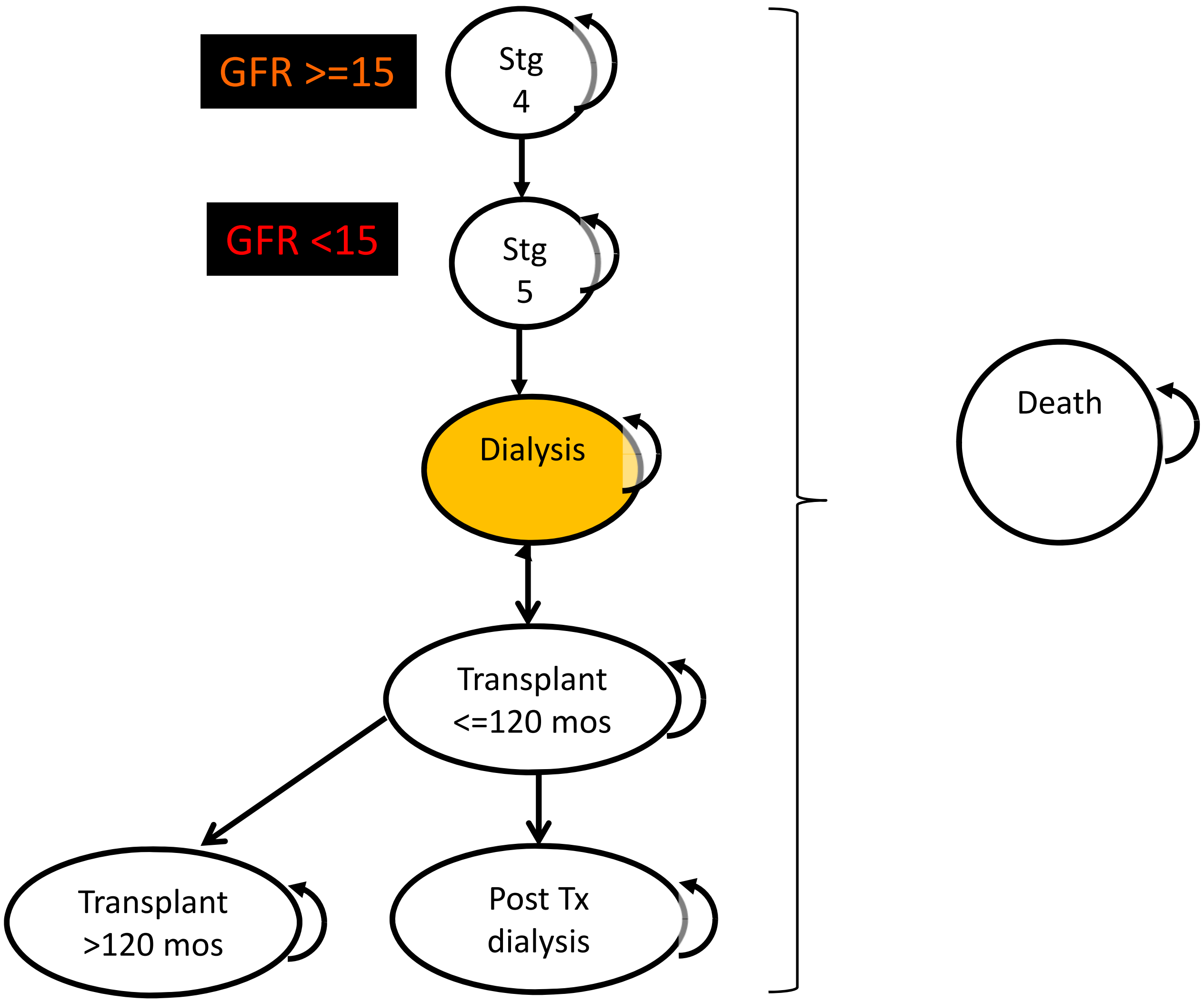
- Enumerate possible health states
- From each state consider possible stochastic events that could lead to transitions
  - Construct the cycle trees
- Populate model parameters
  - Incremental utilities
  - Costs
  - Transition probabilities


GFR >=15
GFR <15
Stg 4
Stg 5
Dialysis
Transplant <=120 mos
Transplant >120 mos
Post Tx dialysis
Death

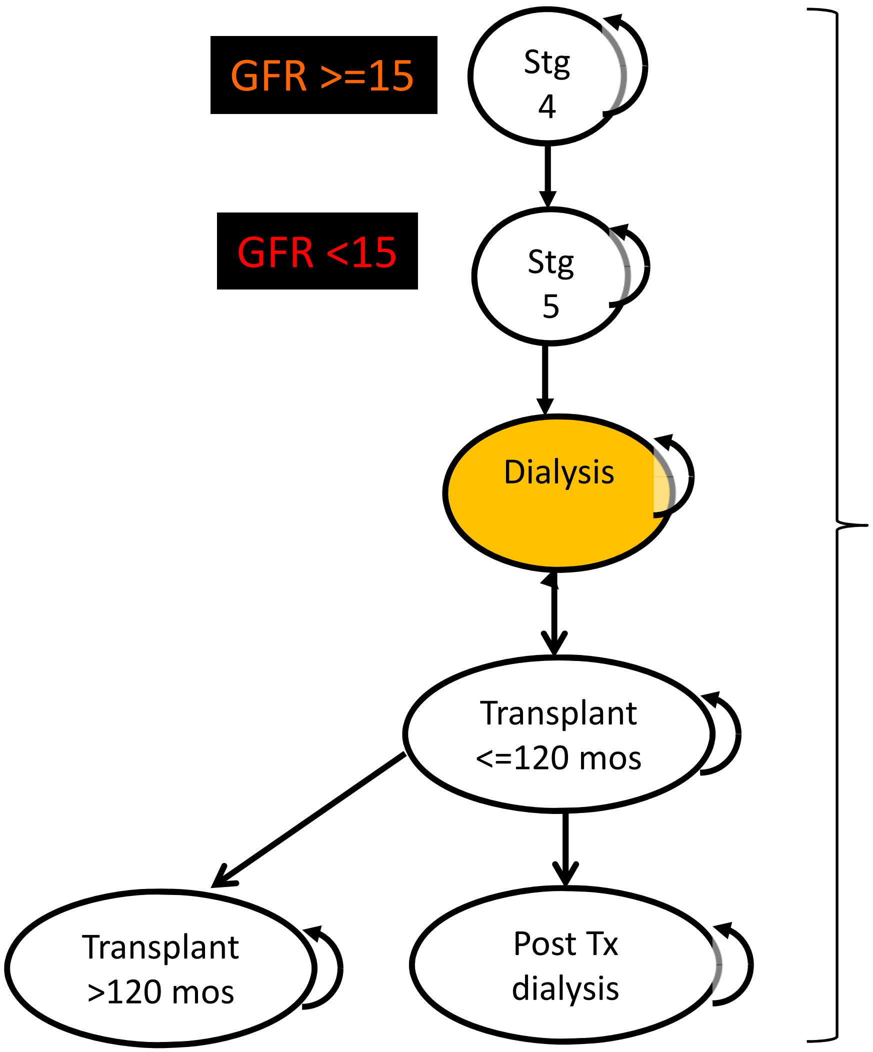

GFR >=15
GFR <15
Stg
4
Stg
5
Dialysis
Transplant
<=120 mos
Transplant
>120 mos
Post Tx
dialysis
Death


GFR >=15
Stg 4
GFR <15
Stg 5
Dialysis
Transplant <=120 mos
Transplant >120 mos
Post Tx dialysis
Death
- Stay in stage 4 & live
- Stay in stage 4 & die
- Transition to stage 5 & live
- Transition to stage 5 & die

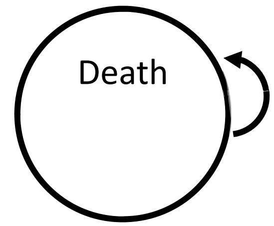

Stage4
--- Markov Information
Init Cost: cStg4*DR
Incr Cost: cStg4*DR
Init Effectiveness:
uStg4*DR
Incr Effectiveness:
uStg4*DR
(GFRstart<=30)&(GF...
ProgressToStage5
GFR<=15
Not
#
die
pDie*RRdie_stg5
live
#
die
pDie
live
#
Death
Stage5
Death
Stage4

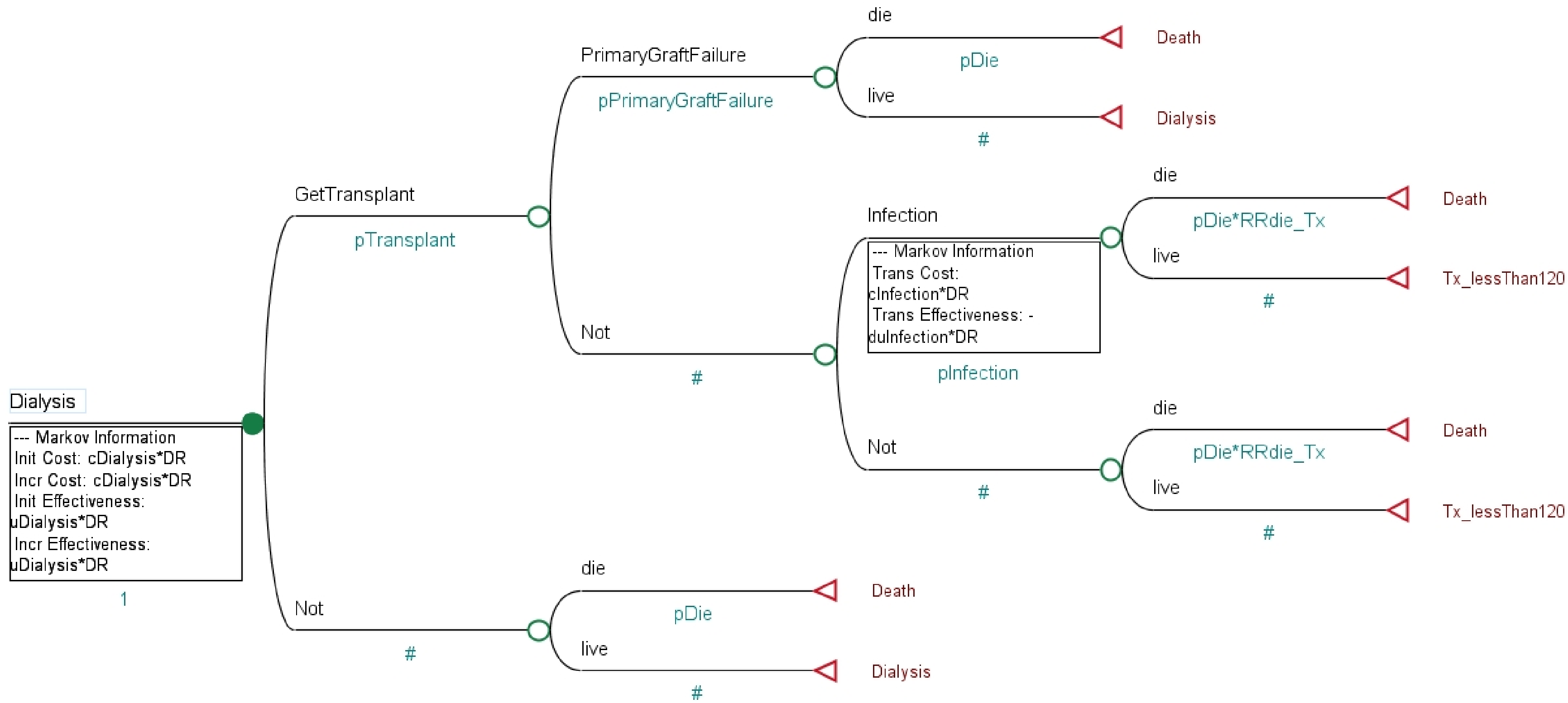


GFR >=15
Stg 4
GFR <15
Stg 5
Dialysis
Transplant <=120 mos
Transplant >120 mos
Post Tx dialysis
Death



- Receive a transplant:
  - Immediately fails & back to dialysis
  - Succeeds
    - Infection
    - No infection
- No transplant, stay on dialysis
- For all of these, live or die





Dialysis
--- Markov Information
Init Cost: cDialysis*DR
Incr Cost: cDialysis*DR
Init Effectiveness: uDialysis*DR
Incr Effectiveness: uDialysis*DR
1
GetTransplant
pTransplant
PrimaryGraftFailure
pPrimaryGraftFailure
die
pDie
Death
live
#
Dialysis
Not
#
Infection
--- Markov Information
Trans Cost: cInfection*DR
Trans Effectiveness: -duInfection*DR
pInfection
die
pDie*RRdie_Tx
Death
live
#
Tx_lessThan120
Not
#
die
pDie*RRdie_Tx
Death
live
#
Tx_lessThan120
Not
#
die
pDie
Death
live
#
Dialysis

Steps to conceptualizing and building a Markov model

- Enumerate possible health states
- From each state consider possible stochastic events that could lead to transitions
  - Construct the cycle trees
- Populate model parameters
  - Incremental utilities
  - Costs
  - Transition probabilities

Probabilities come in two flavors

- Bernoulli probabilities
  - Flip-of-a-coin events, do or do not happen
  - For example, immediate transplant failure (a.k.a. primary graft failure)

Probabilities come in two flavors

- Bernoulli probabilities
  - Why do I use this terminology?

Consider the binomial probability distribution for a moment

$$P(x = k|n, p) = \binom{n}{k} \cdot p^k \cdot (1 - p)^{n-k}$$

$$P(x = 4|10, 0.3) = \binom{10}{4} \cdot 0.3^4 \cdot (0.7)^6 = \frac{10!}{4!\,6!} \cdot 0.3^4 \cdot (0.7)^6$$

$$P(x = 4|10, 0.3) = \frac{10 \cdot 9 \cdot 8 \cdot 7}{4 \cdot 3 \cdot 2 \cdot 1} \cdot 0.3^4 \cdot (0.7)^6 = 210 \cdot 0.0081 \cdot 0.11765 = 0.2$$

What if $n = 1$?

- What values can k have?

What if n = 1?

- What values can k have?
  - Well, 0 or 1:

$$P(x = 0|1, p) = \binom{1}{0} \cdot p^0 \cdot (1 - p)^{1-0} = 1 - p$$

$$P(x = 1|1, p) = \binom{1}{1} \cdot p^1 \cdot (1 - p)^{1-1} = p$$

The Bernoulli distribution is a special case of the binomial where n = 1

- The percentage chance of drawing a ‘0’ is (1 – p)*100%
- The percentage chance of drawing a ‘1’ is p*100%
- This is exactly like flipping a weighted coin such that probability of a heads is p

Bernoulli probabilities

- May have a simple estimate of p from the literature

Bernoulli probabilities

- Or something a little more sophisticated:
- Say we have a baseline probability estimate
- And an odds ratio from a published logistic regression
- For example:
  - pPimaryGraftFail = 0.025 if age < 55
  - Odds Ratio = 2.1 if age >= 55
- How to model this?

Probabilities and Odds

1. Convert probability to odds

   p=0.025 → odds = (0.025/0.975)= 0.0256

2. Apply odds ratio

   0.0256 x 2.1 = 0.054

3. Convert back to probability

   0.054/1.054 = 0.051

Probabilities and Odds

1. Convert probability to odds

   $p=0.025 \rightarrow$ odds = $(0.025/0.975)= .0256$

2. Apply odds ratio

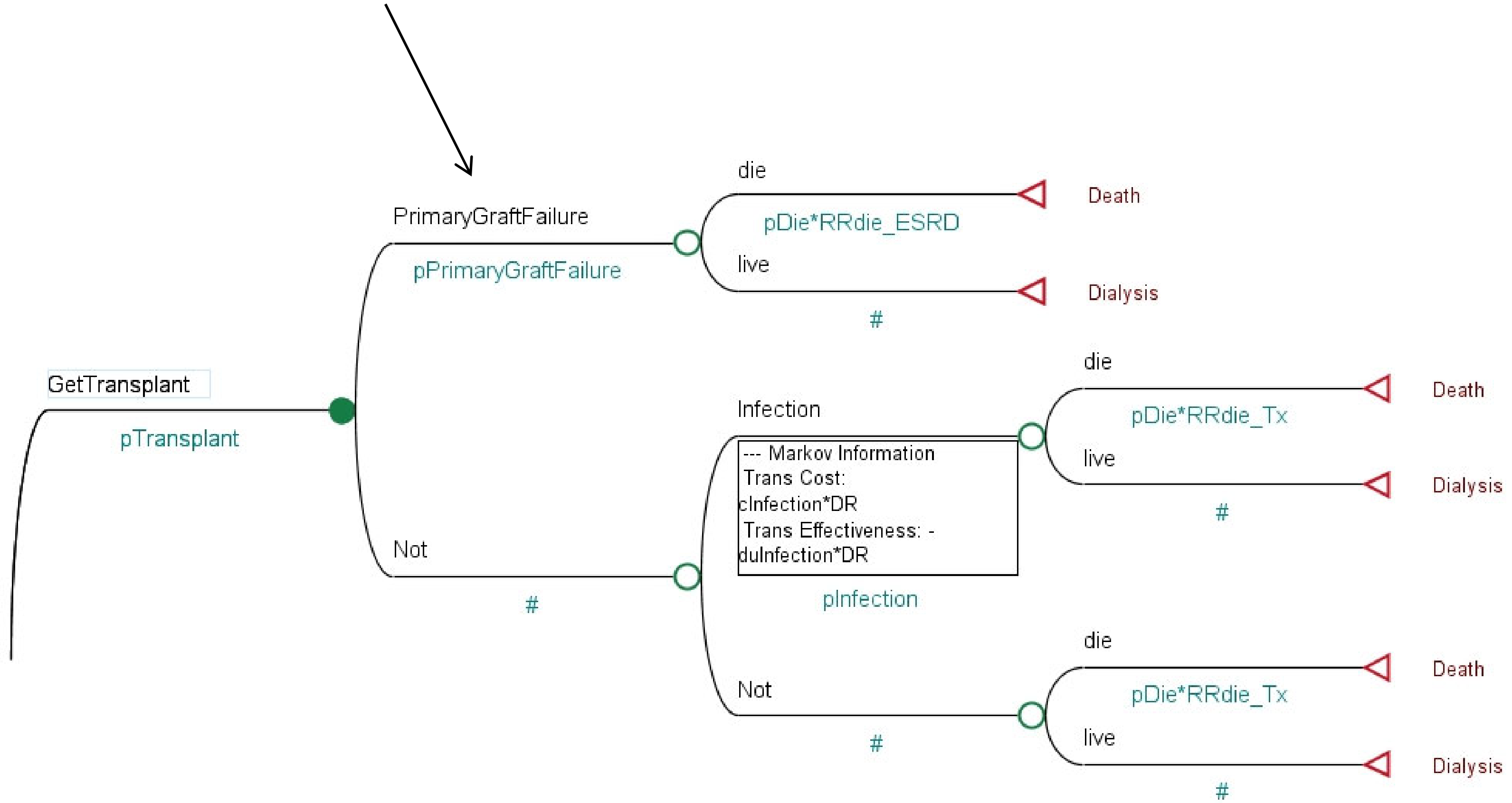
   $.0256 \times 2.1 = .054$

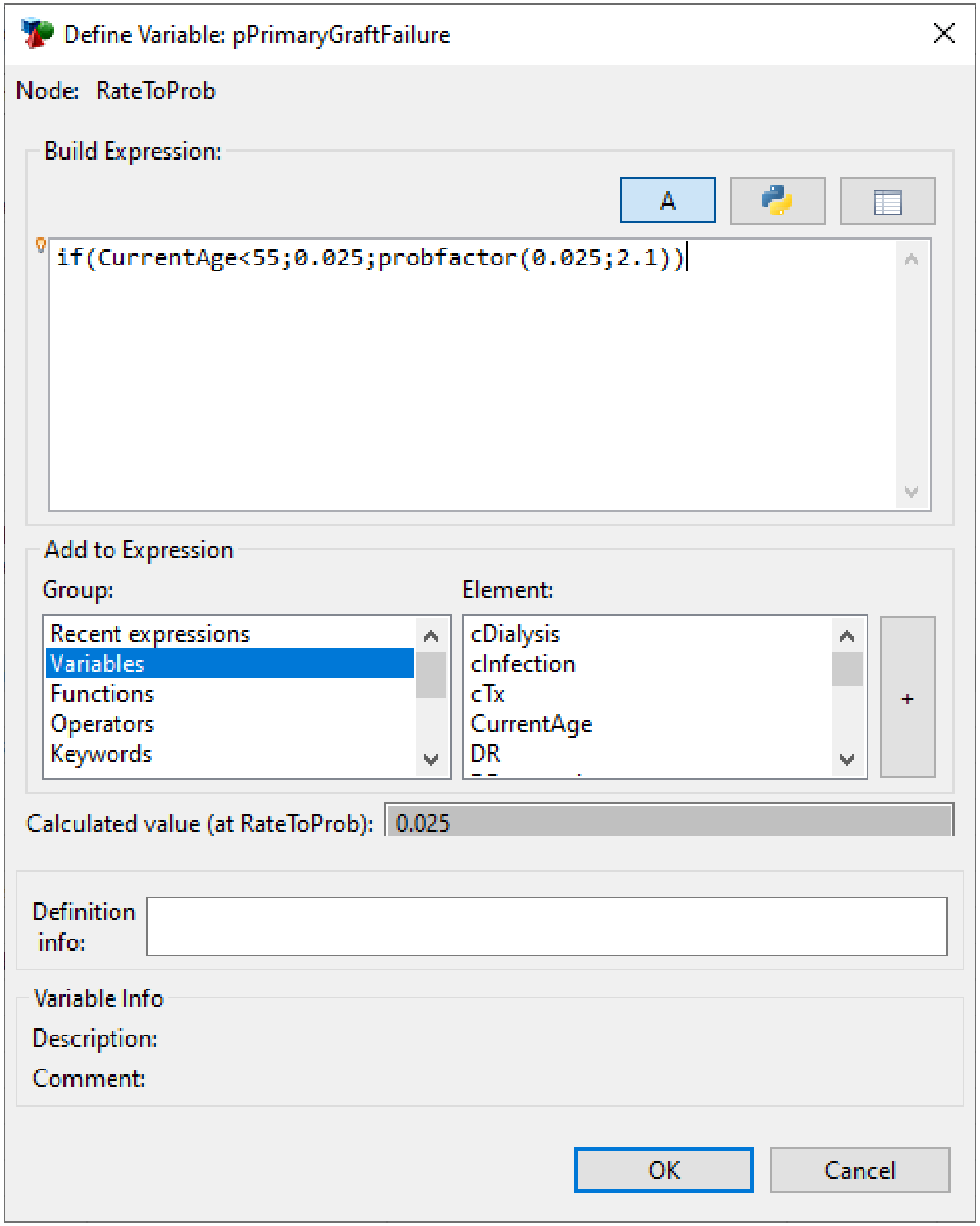
3. Convert back to probability

   $.054/1.054 = .051$

There is a built-in TreeAge function that does this for you:
probfactor(pBaseline; OR)

Example of a Bernoulli probability




Define Variable: pPrimaryGraftFailure
Node: RateToProb
Build Expression:
A
if(CurrentAge<55;0.025;probfactor(0.025;2.1))
Add to Expression
Group:
Recent expressions
Variables
Functions
Operators
Keywords
Element:
cDialysis
cInfection
cTx
CurrentAge
DR
+
Calculated value (at RateToProb):
0.025
Definition info:
Variable Info
Description:
Comment:
OK
Cancel

Probabilities come in two flavors

- Bernoulli probabilities
- Time-to-event probabilities
  - For example, risk of chronic transplant rejection and failure

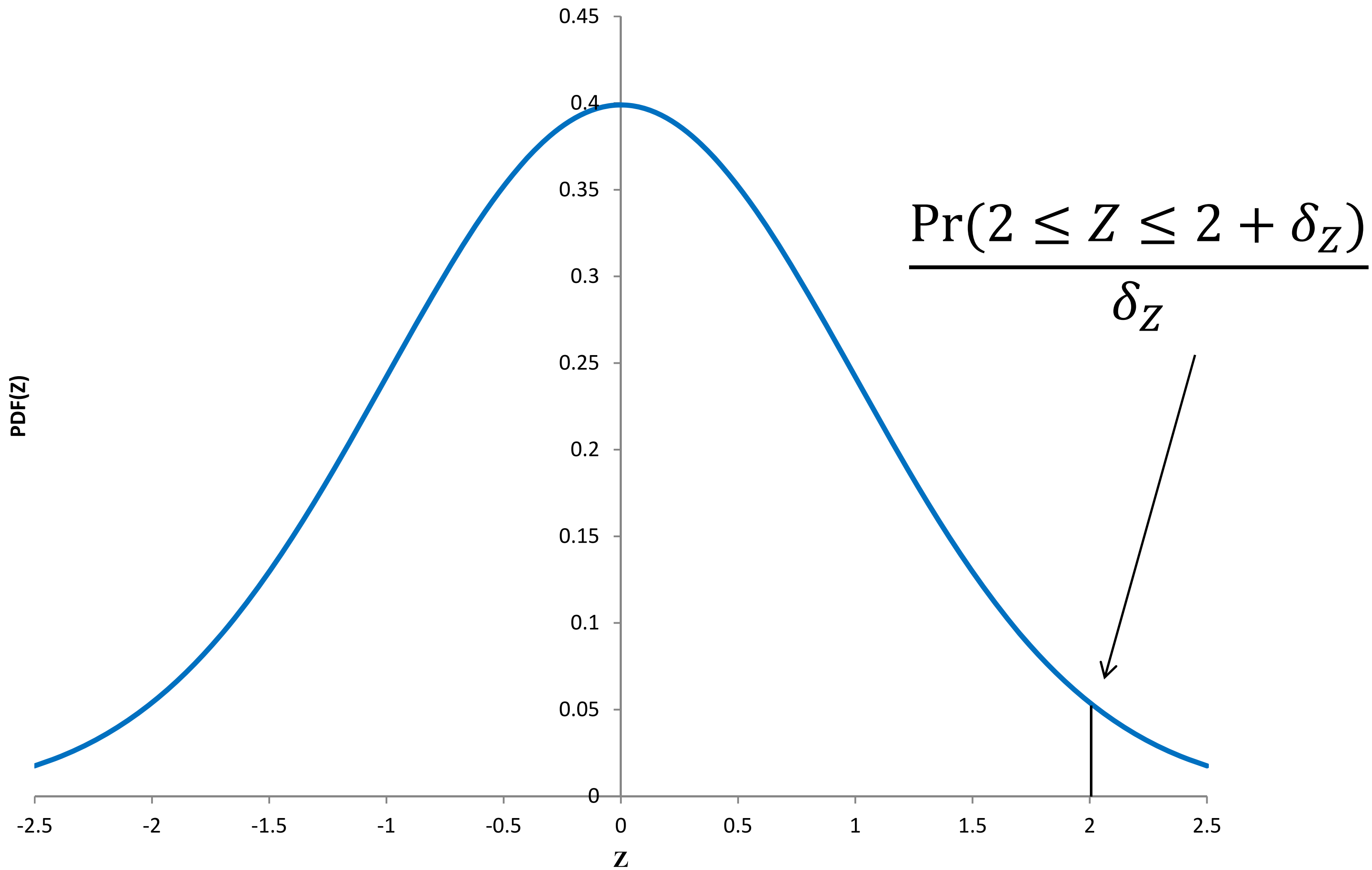
Time to event probabilities

- The situation when risk of an event accumulates over time
- Data is obtained from a time-to-event (TTE) a.k.a survival analysis
- Our job is to take such data and use it to estimate the per cycle probability of the event – i.e. what is the cumulative risk of the event over one cycle's worth of time?
- For this we need to consider rates and probabilities

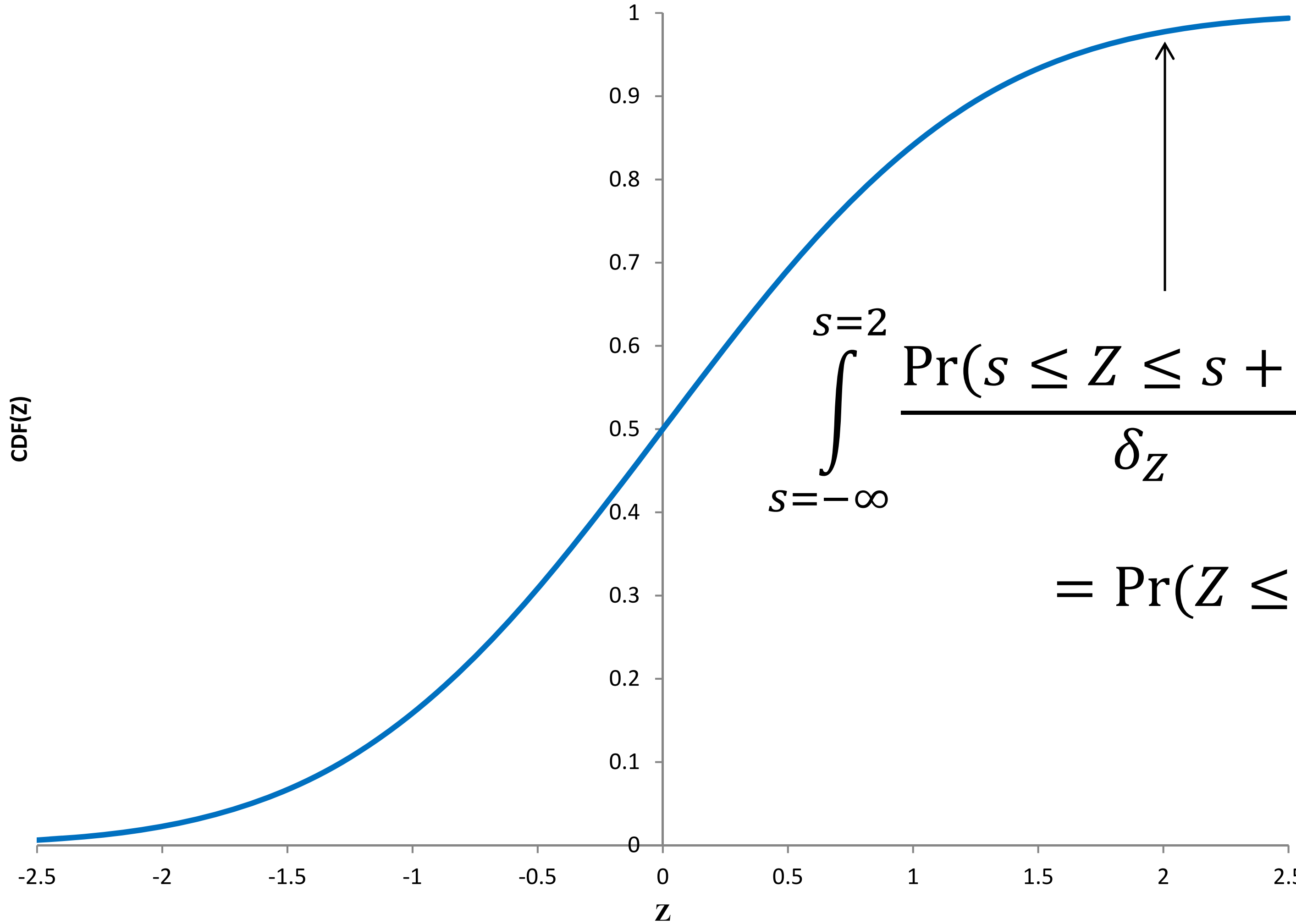
Probability = number of events / number of people at risk

Rate = number of events / time

The std. normal PDF



The std. normal CDF

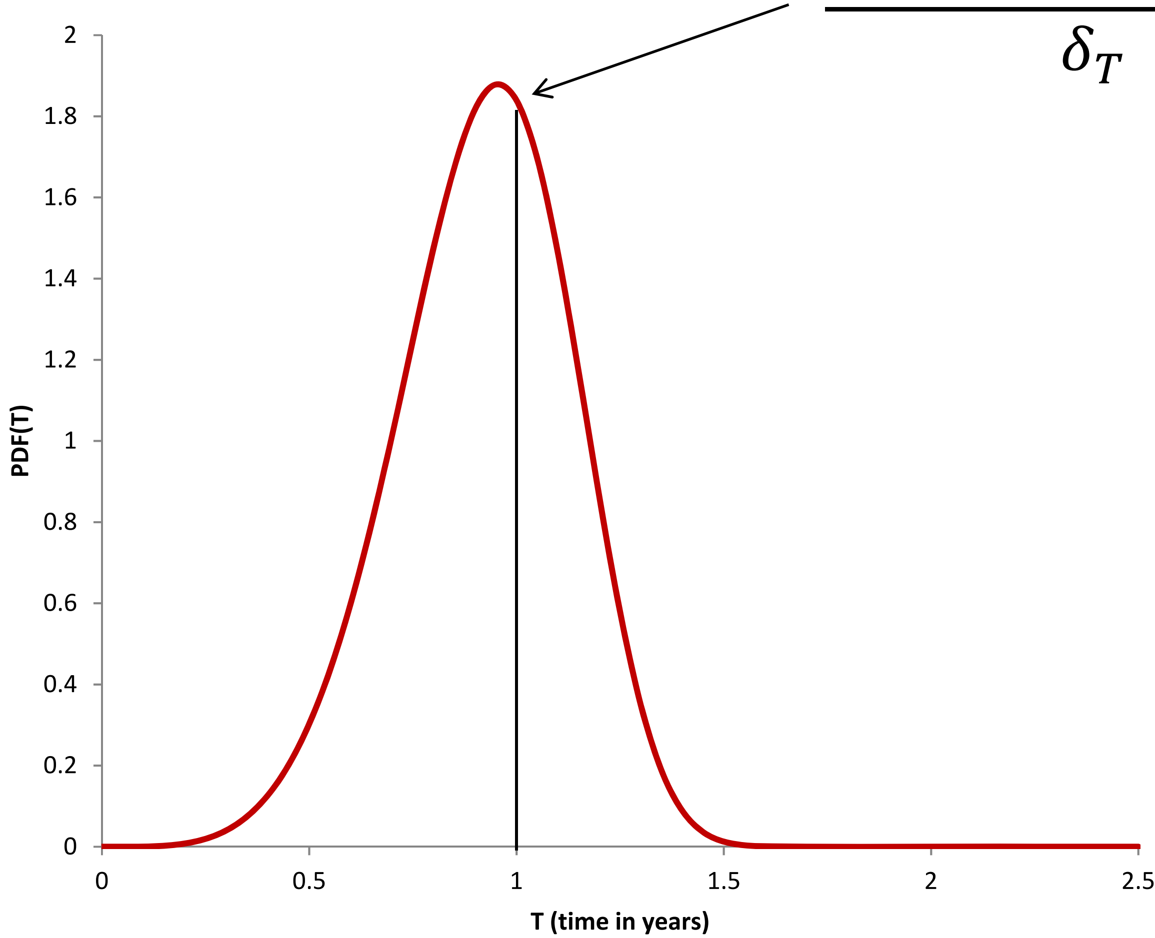


$$\int_{s=-\infty}^{s=2} \frac{\Pr(s \leq Z \leq s + \delta_Z)}{\delta_Z} \delta_Z$$

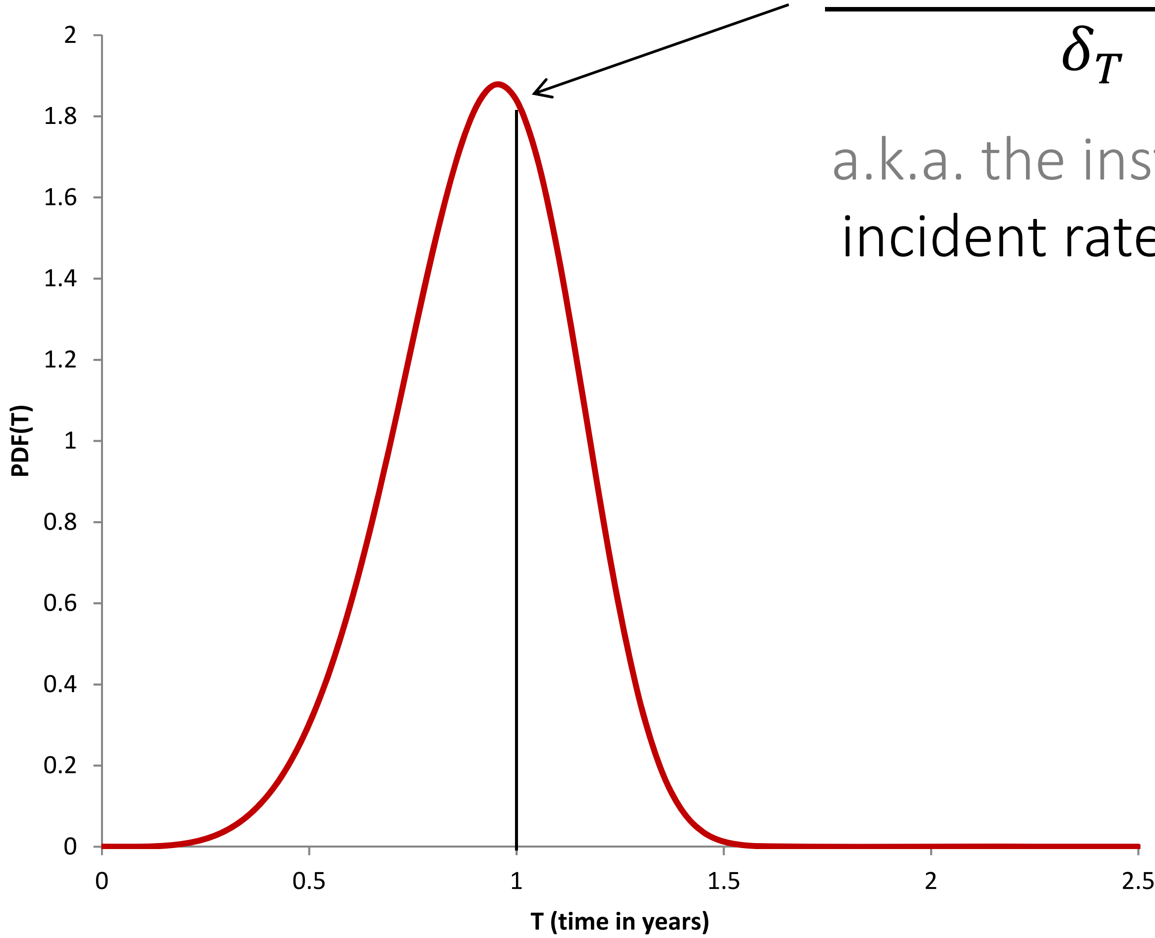
$$= \Pr(Z \leq 2)$$

A Weibull PDF (λ =1, k = 5)

$$\frac{\Pr(1 \leq T \leq 1 + \delta_T)}{\delta_T}$$



A Weibull PDF (λ =1, k = 5)

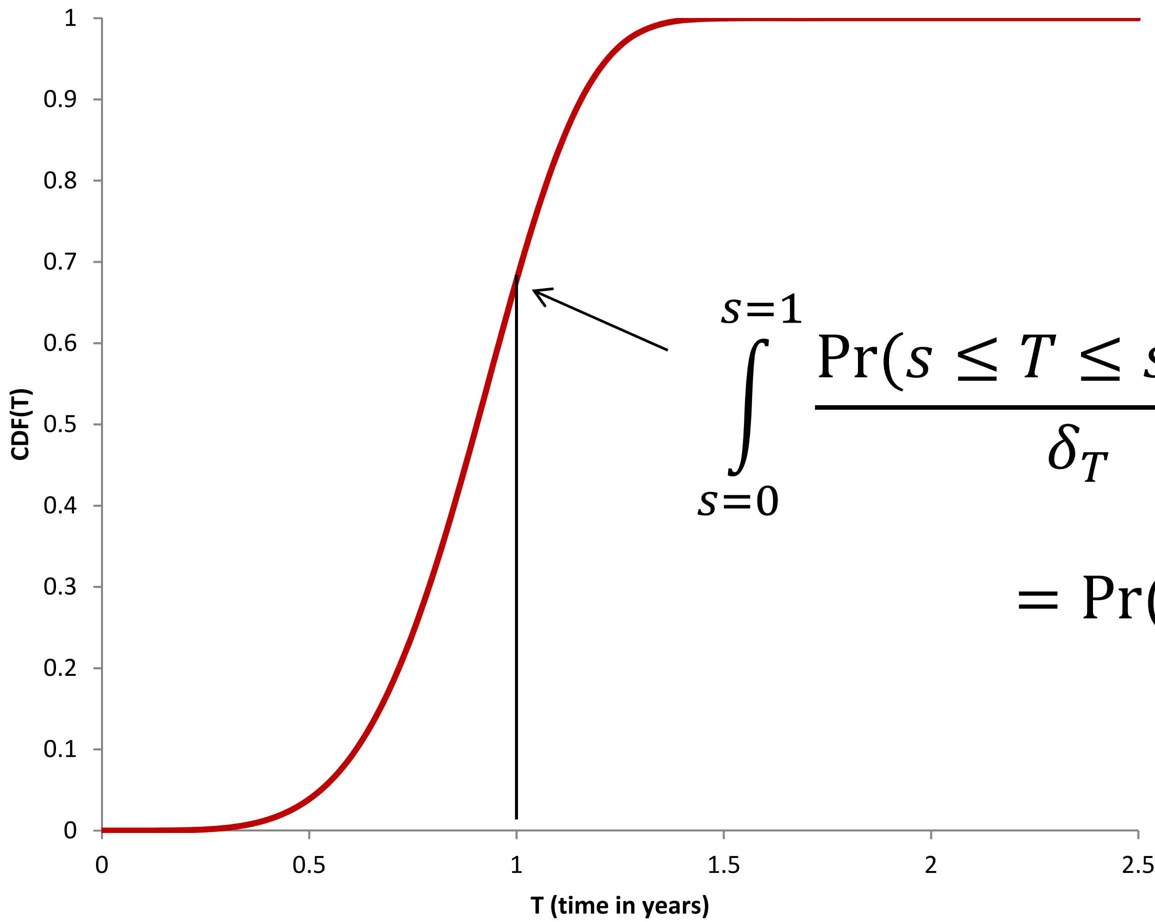


$$\frac{\Pr(1 \leq T \leq 1 + \delta_T)}{\delta_T}$$

a.k.a. the instantaneous incident rate at T = 1 yr

A Weibull CDF (λ =1, k = 5)

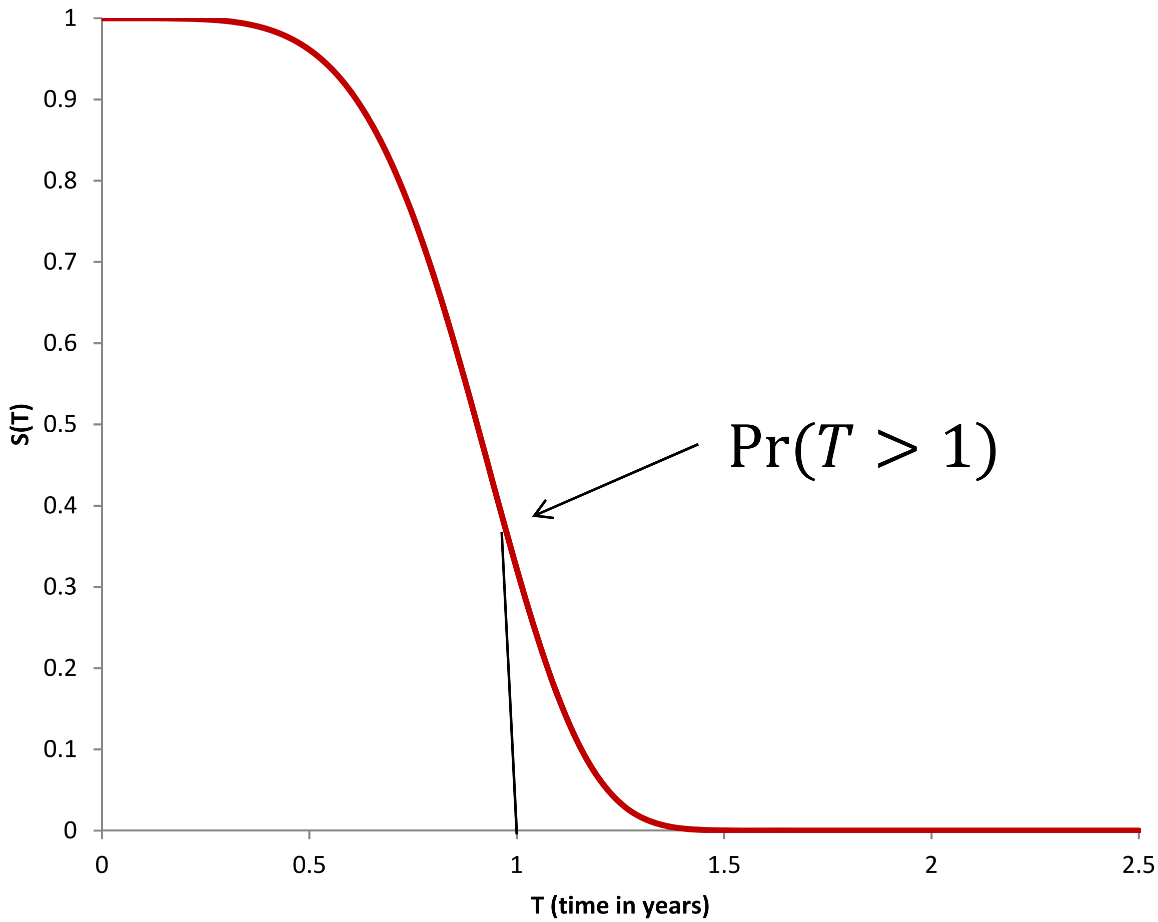
a.k.a the cumulative incidence curve



$$\int_{s=0}^{s=1} \frac{\Pr(s \leq T \leq s + \delta_T)}{\delta_T} \delta_T$$

$$= \Pr(T \leq 1)$$

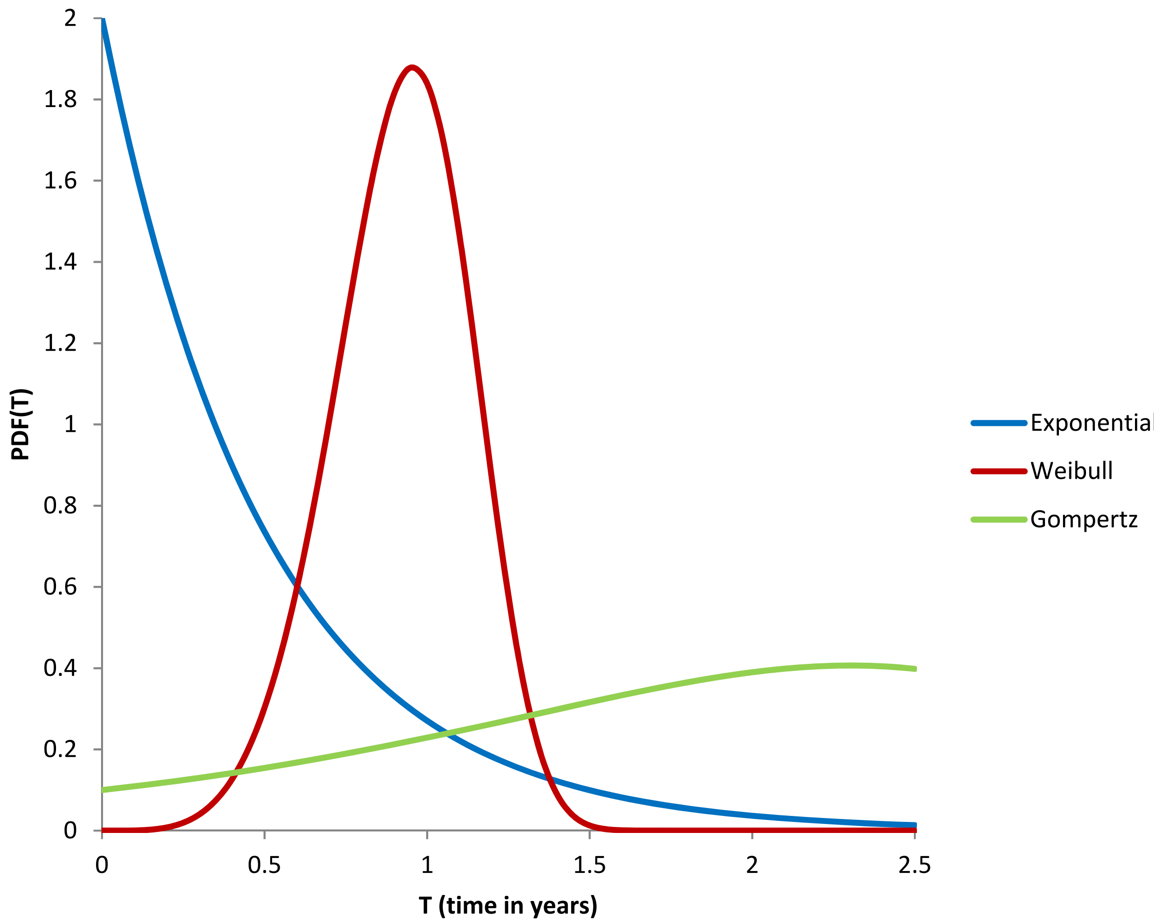
A Weibull Survival Function (λ =1, k = 5)
= 1 - CDF



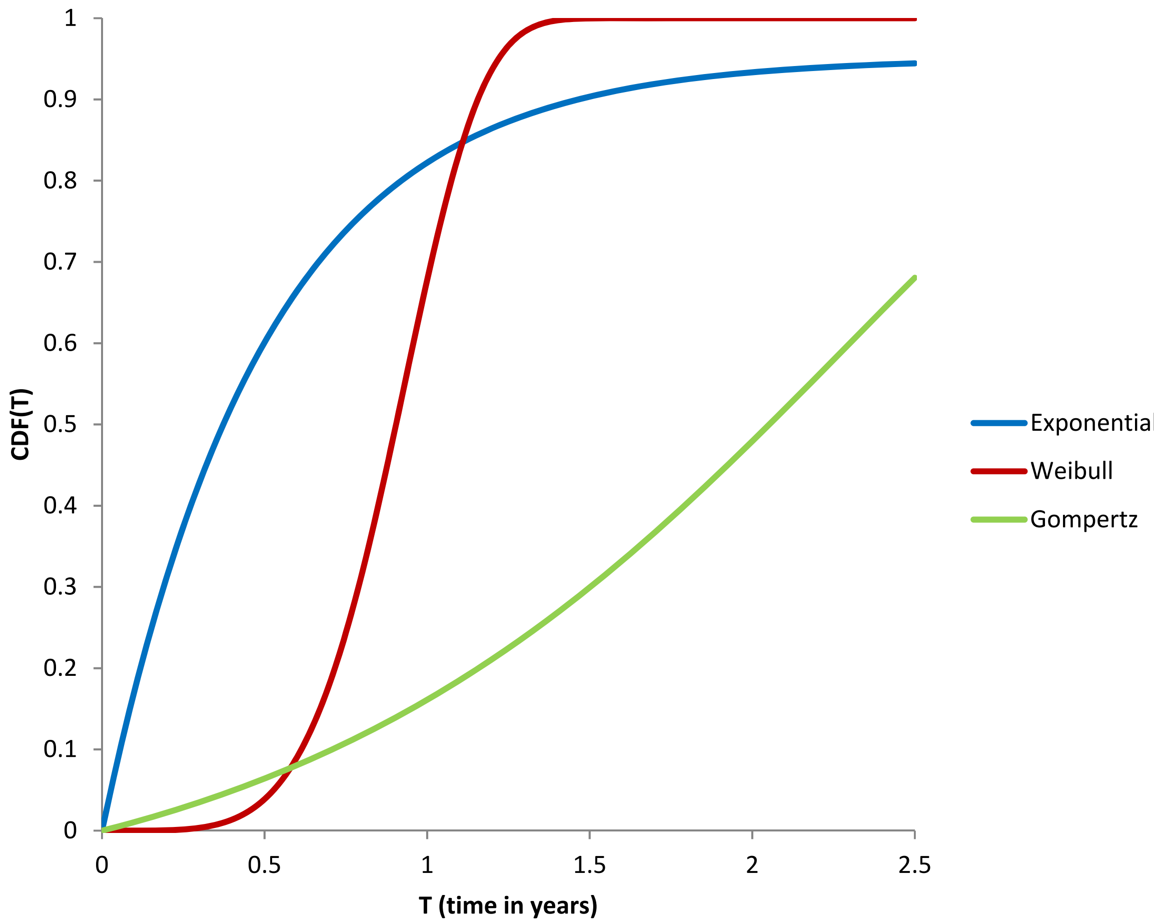
Types of TTE distributions/survival functions

- Fully parametric
  - Exponential
  - Weibull
  - Gompertz
- Semi-parametric
  - Cox
- Non-parametric
  - Kaplan-Meier

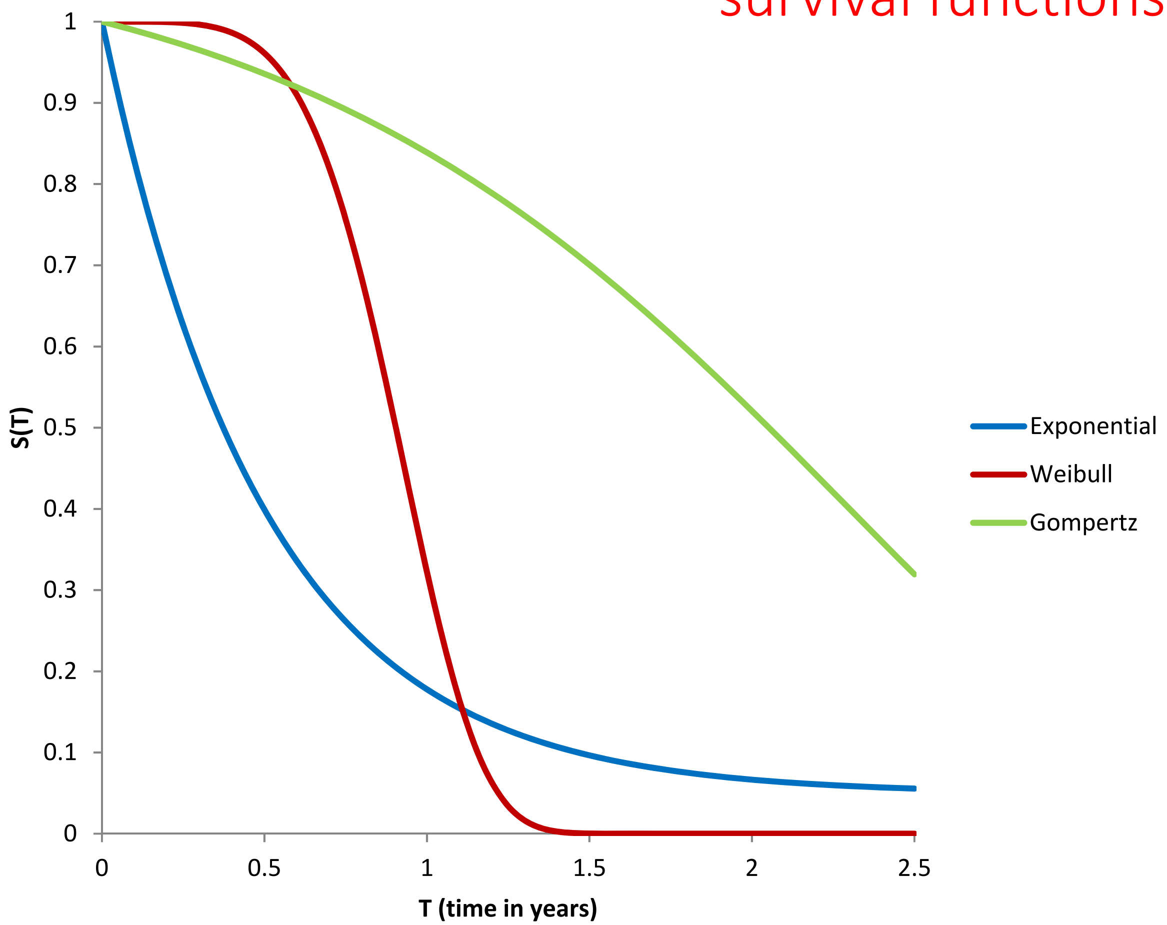
Various parametric TTE PDFs



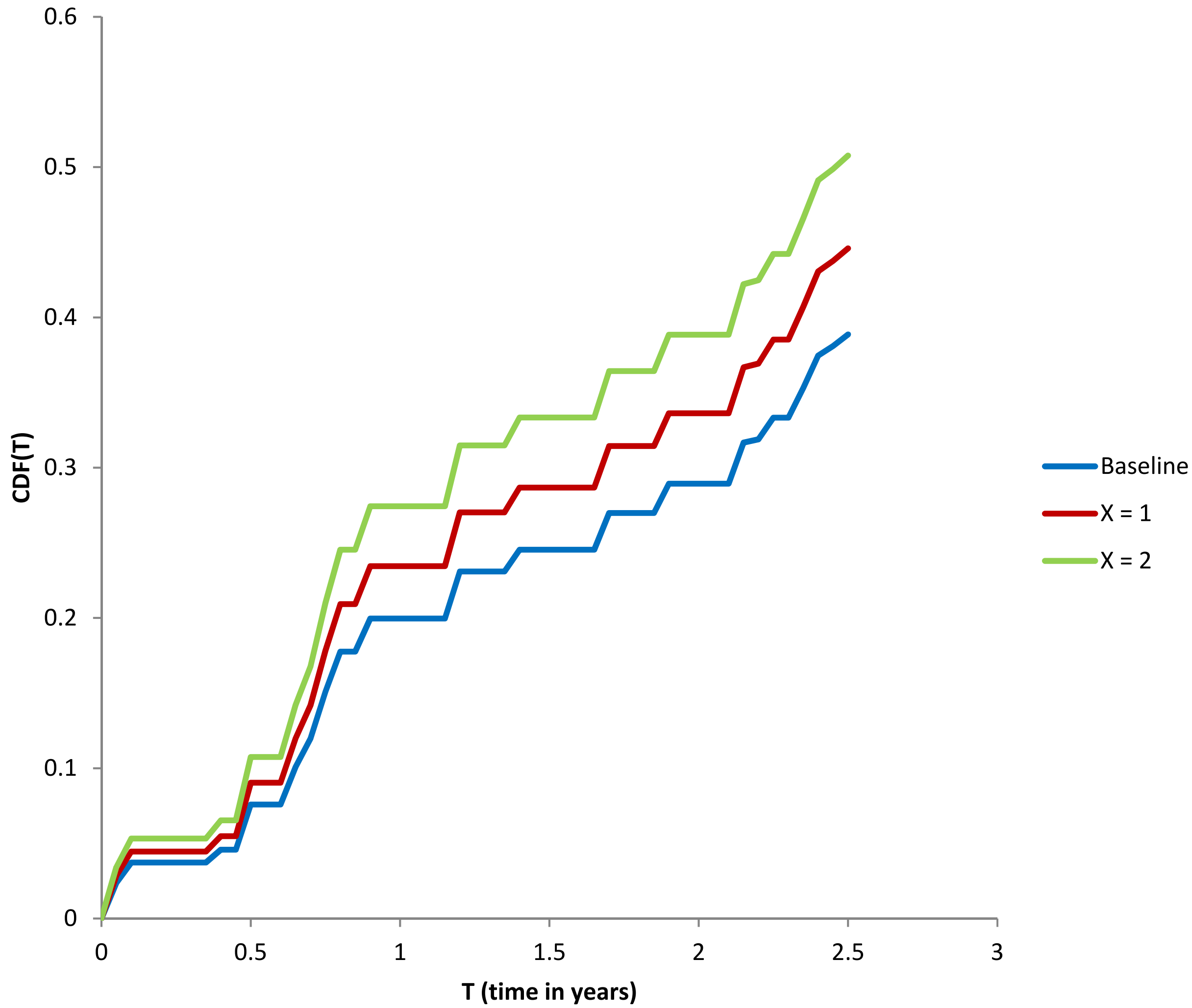
Various parametric TTE CDFs



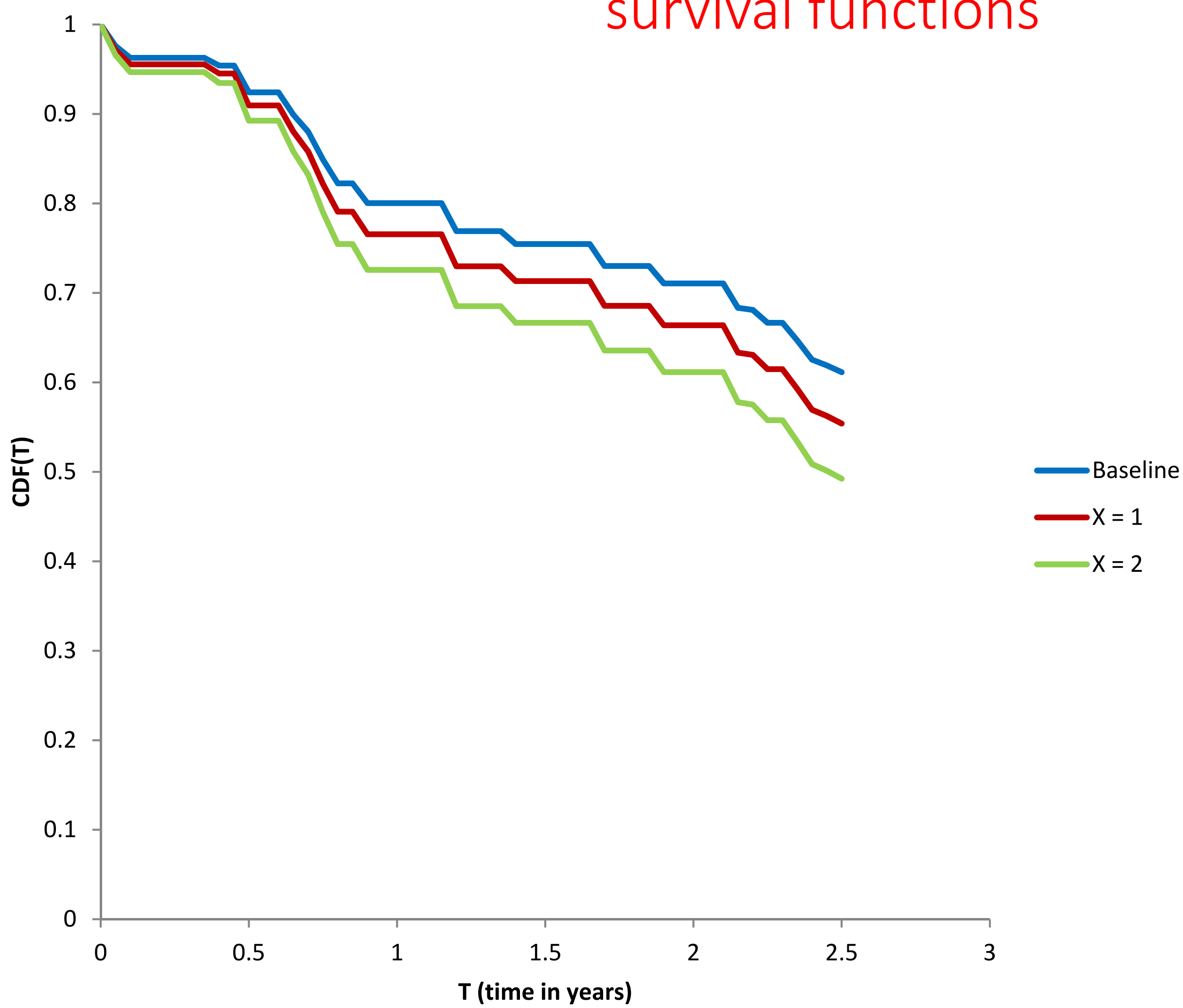
Various parametric survival functions



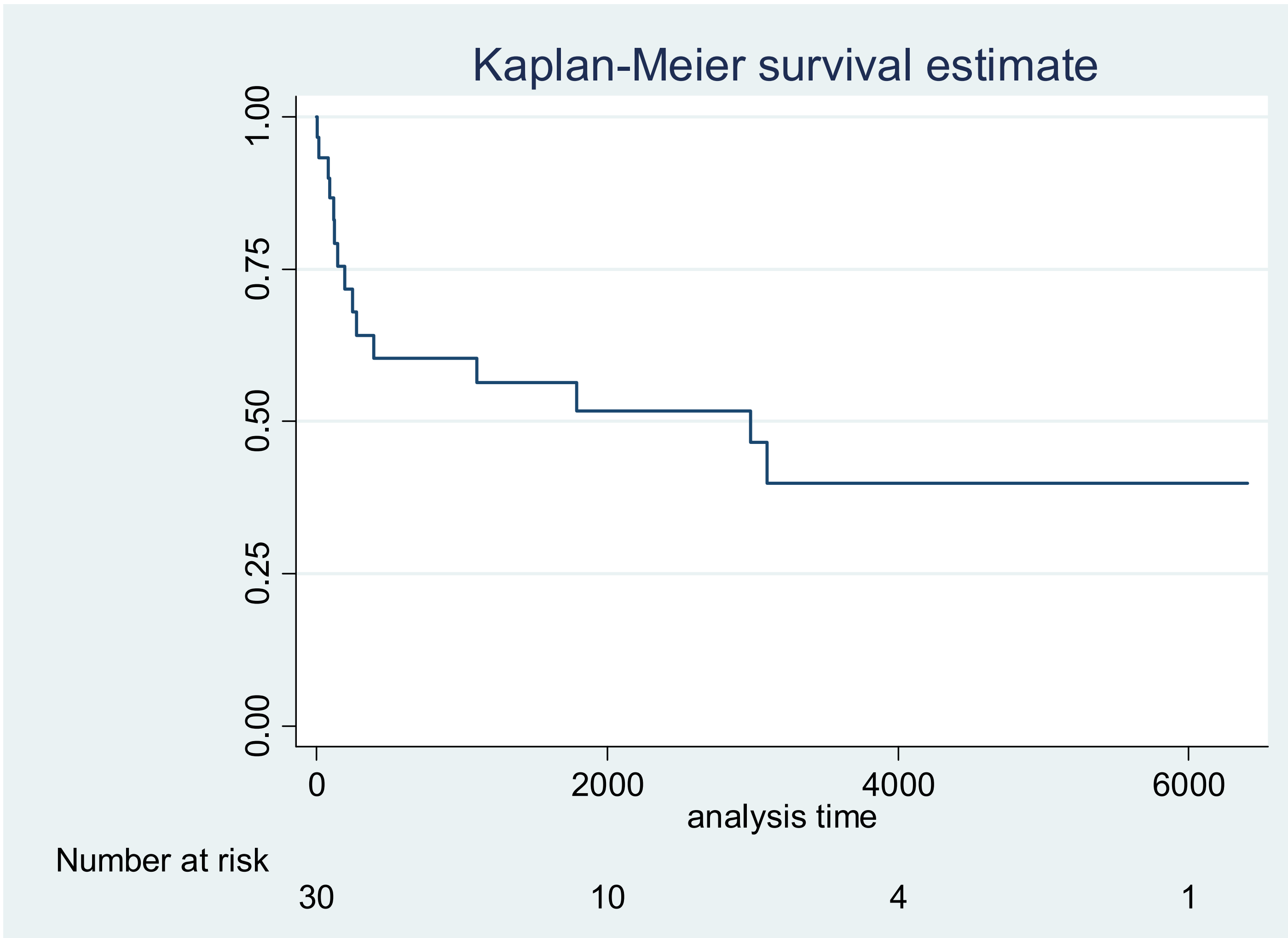
Cox semi-parametric TTE CDFs



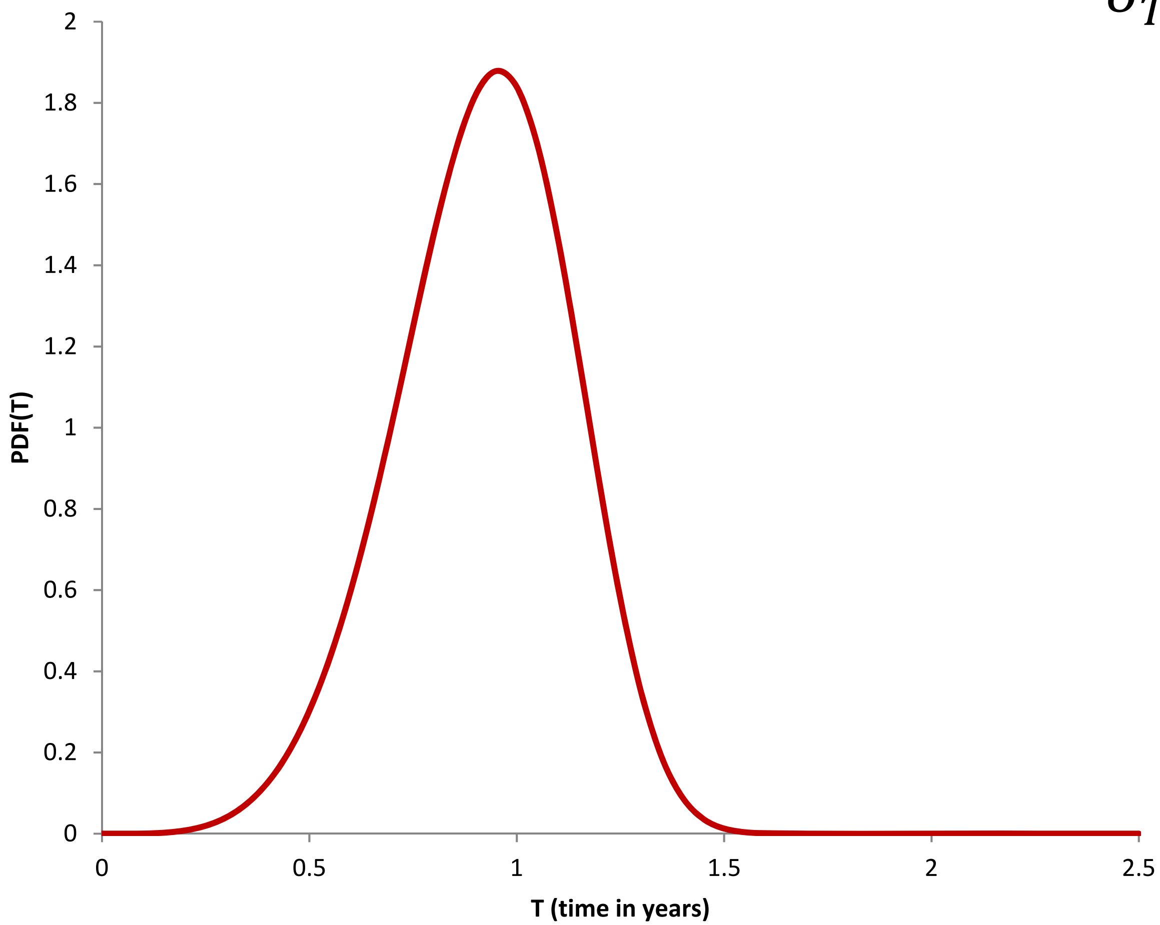
Cox semi-parametric survival functions

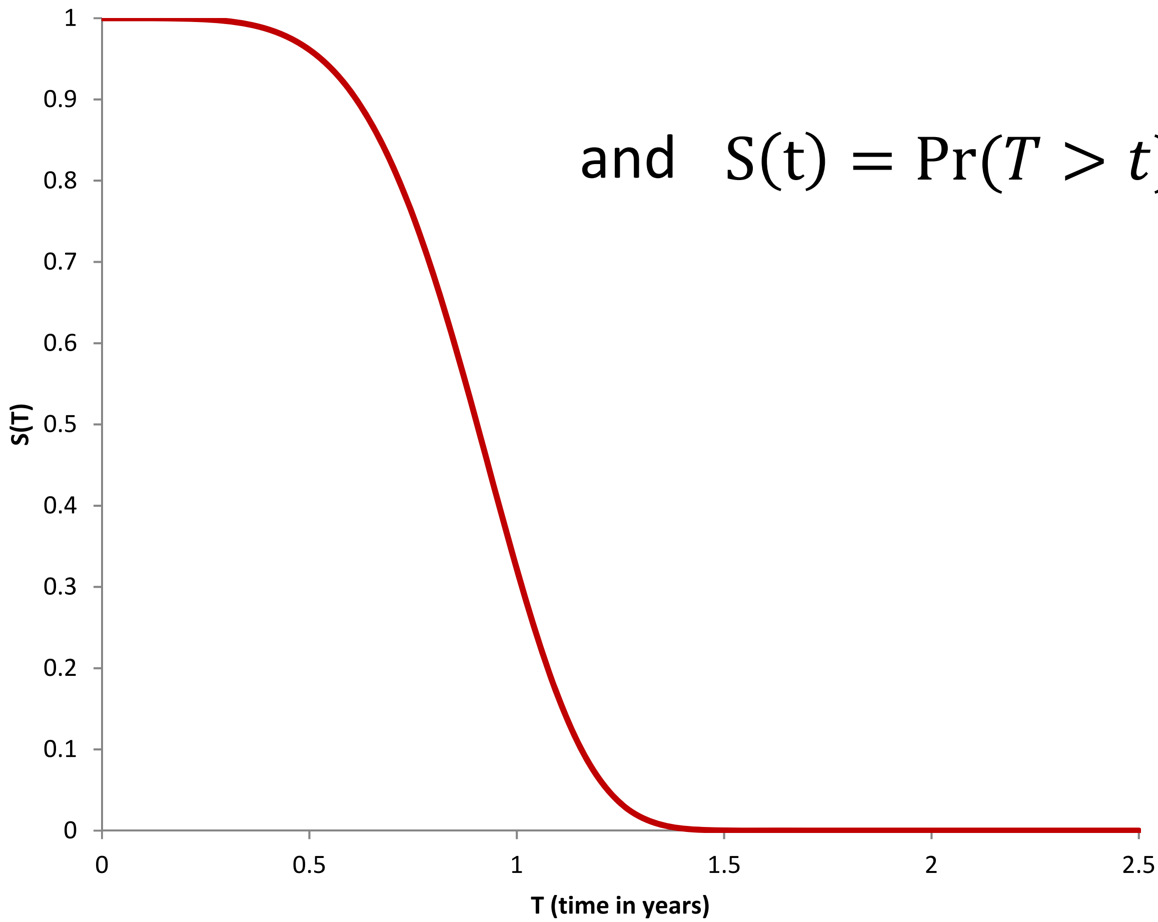


Kaplan-Meier completely non-parametric survival function



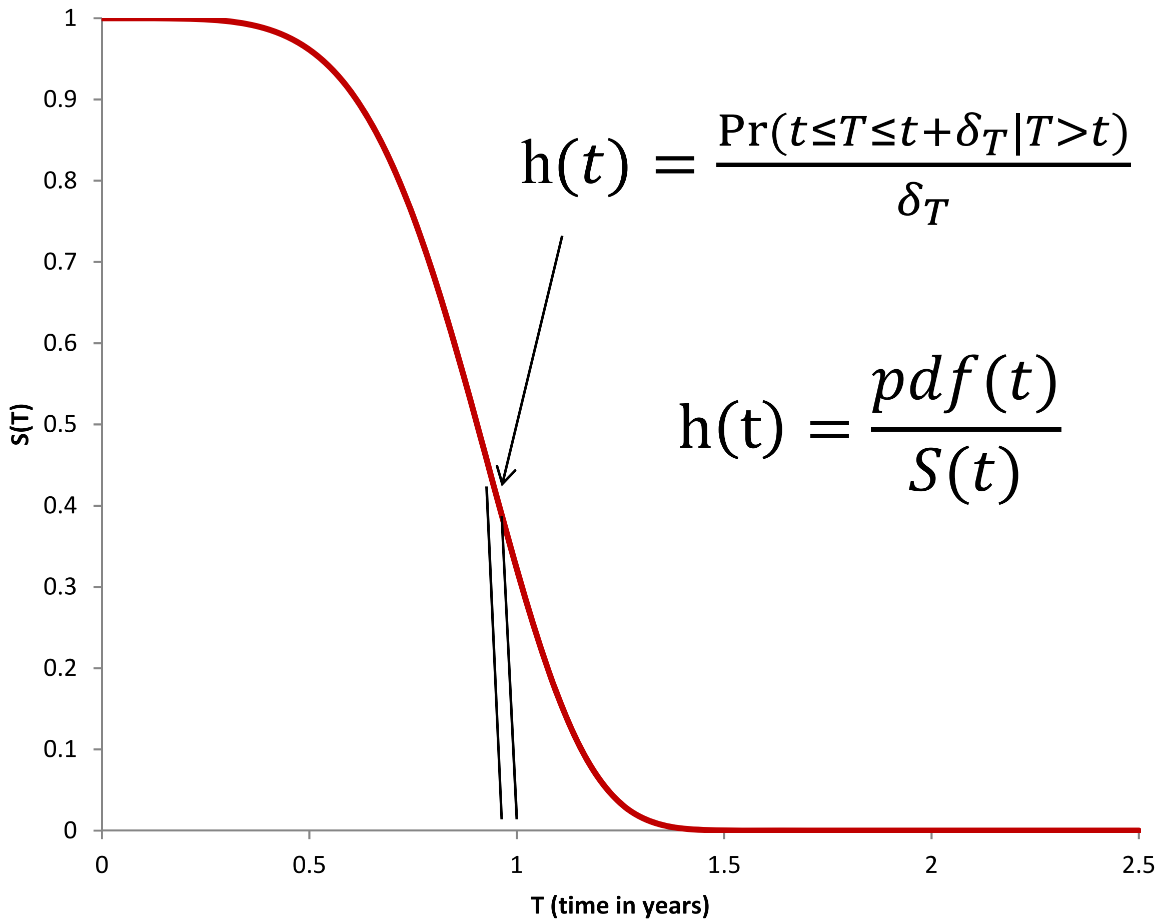
Recall $pdf(t) = \dfrac{\Pr(t \leq T \leq t + \delta_T)}{\delta_T}$





and $S(t) = \Pr(T > t)$

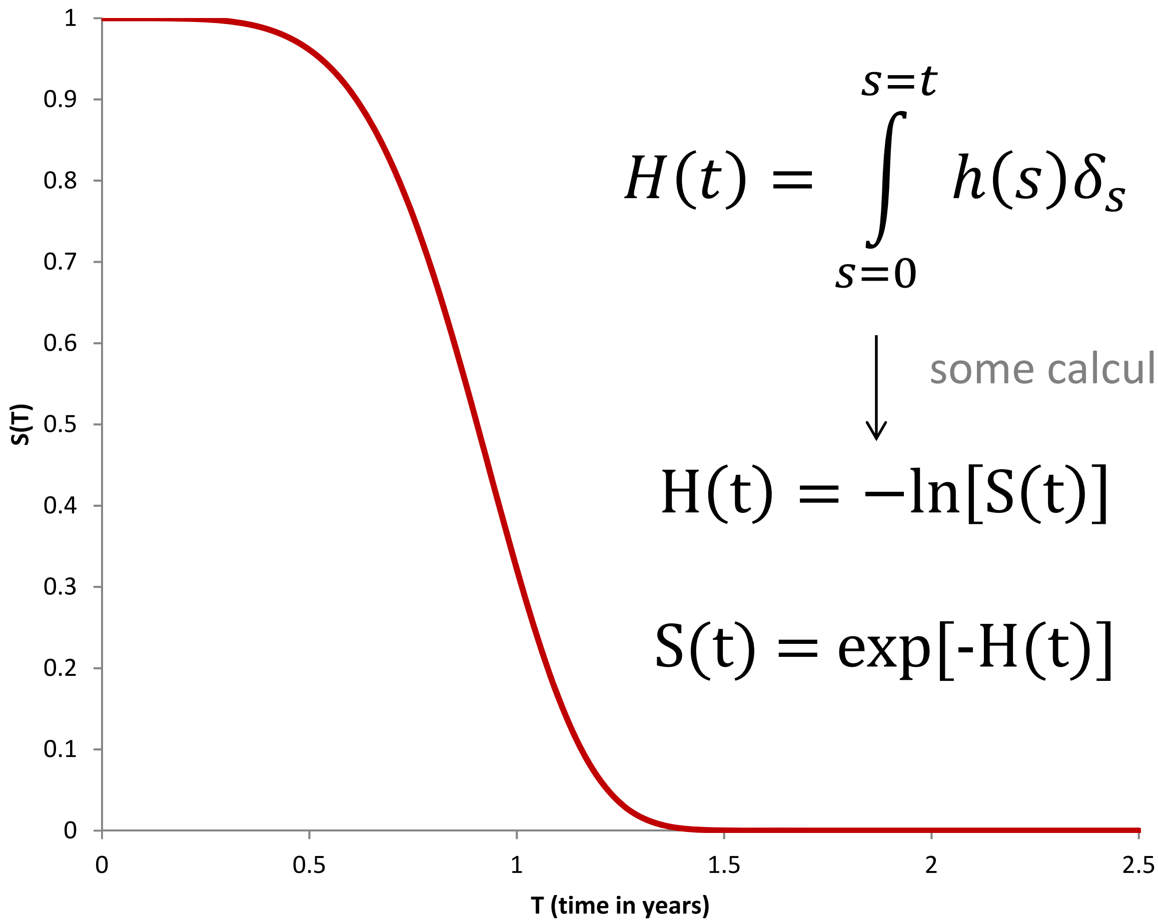
The instantaneous hazard rate or function, h(t)



$$\mathrm{h}(t) = \frac{\Pr(t \le T \le t + \delta_T \mid T > t)}{\delta_T}$$

$$\mathrm{h(t)} = \frac{pdf(t)}{S(t)}$$

The cumulative hazard function, H(t)



$$H(t) = \int_{s=0}^{s=t} h(s)\delta_s$$

some calculus

$$\mathrm{H(t)} = -\ln[\mathrm{S(t)}]$$

$$\mathrm{S(t)} = \exp[\text{-}\mathrm{H(t)}]$$

The relevance of this for decision models:

Probability of no event during a cycle = $S(t_{end})/S(t_{beginning})$

Probability of event during a cycle = $1 – [S(t_e)/S(t_b)]$

$= 1- [\exp(-H(t_e))/\exp(-H(t_b))]$

$= 1- \exp[H(t_b) - H(t_e)]$

The exponential survival model

$pdf(t) = \lambda * exp(-\lambda t)$

$cdf(t) = 1 - exp(-\lambda t)$

$S(t) = exp(-\lambda t)$

$h(t) = \lambda$

$H(t) = \lambda t$

The exponential survival model

$\text{pdf}(t) = \lambda * \exp(-\lambda t)$

$\text{cdf}(t) = 1 - \exp(-\lambda t)$

$S(t) = \exp(-\lambda t)$

$h(t) = \lambda$

$H(t) = \lambda t$

Probability of event during a cycle = $1 - [S(t_e)/S(t_b)]$

$$= 1 - [\exp(-H(t_e))/\exp(-H(t_b))]$$

$$= 1 - \exp[H(t_b) - H(t_e)]$$

$$= 1 - \exp[\lambda t_b - \lambda t_e] = 1 - \exp[\lambda(t_b - t_e)]$$

$$t_b - t_e = -1$$

So per cycle probability = $1 - \exp(-\lambda)$

Probability of event during a cycle = $1 - [S(t_e)/S(t_b)]$

$$= 1 - [\exp(-H(t_e))/\exp(-H(t_b))]$$

$$= 1 - \exp[H(t_b) - H(t_e)]$$

So per cycle probability = $1 - \exp(-\lambda)$

There is a built in TreeAge function for this:

ratetoprob(λ;1)

Probability of event during a cycle = $1 - [S(t_e)/S(t_b)]$

$$= 1 - [\exp(-H(t_e))/\exp(-H(t_b))]$$

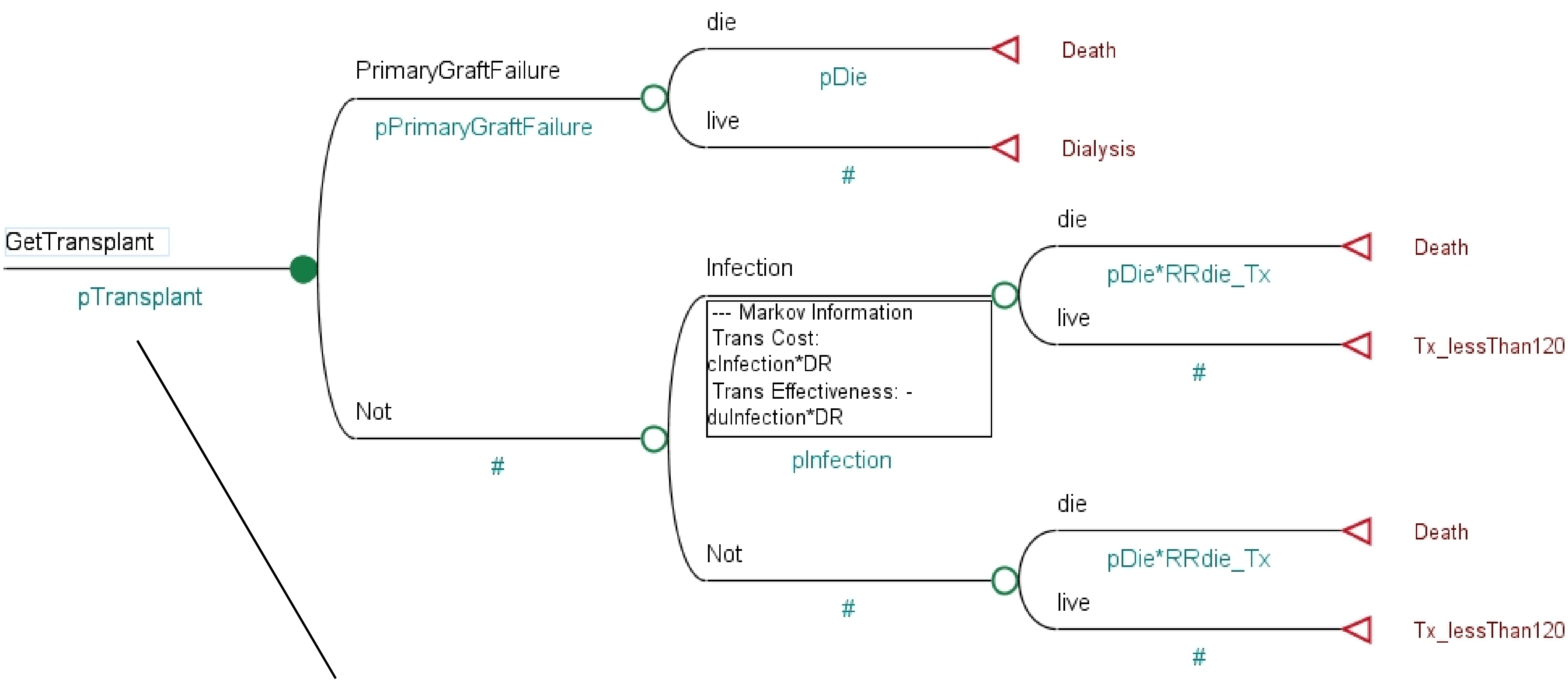
$$= 1 - \exp[H(t_b) - H(t_e)]$$

So per cycle probability = $1 - \exp(-\lambda)$

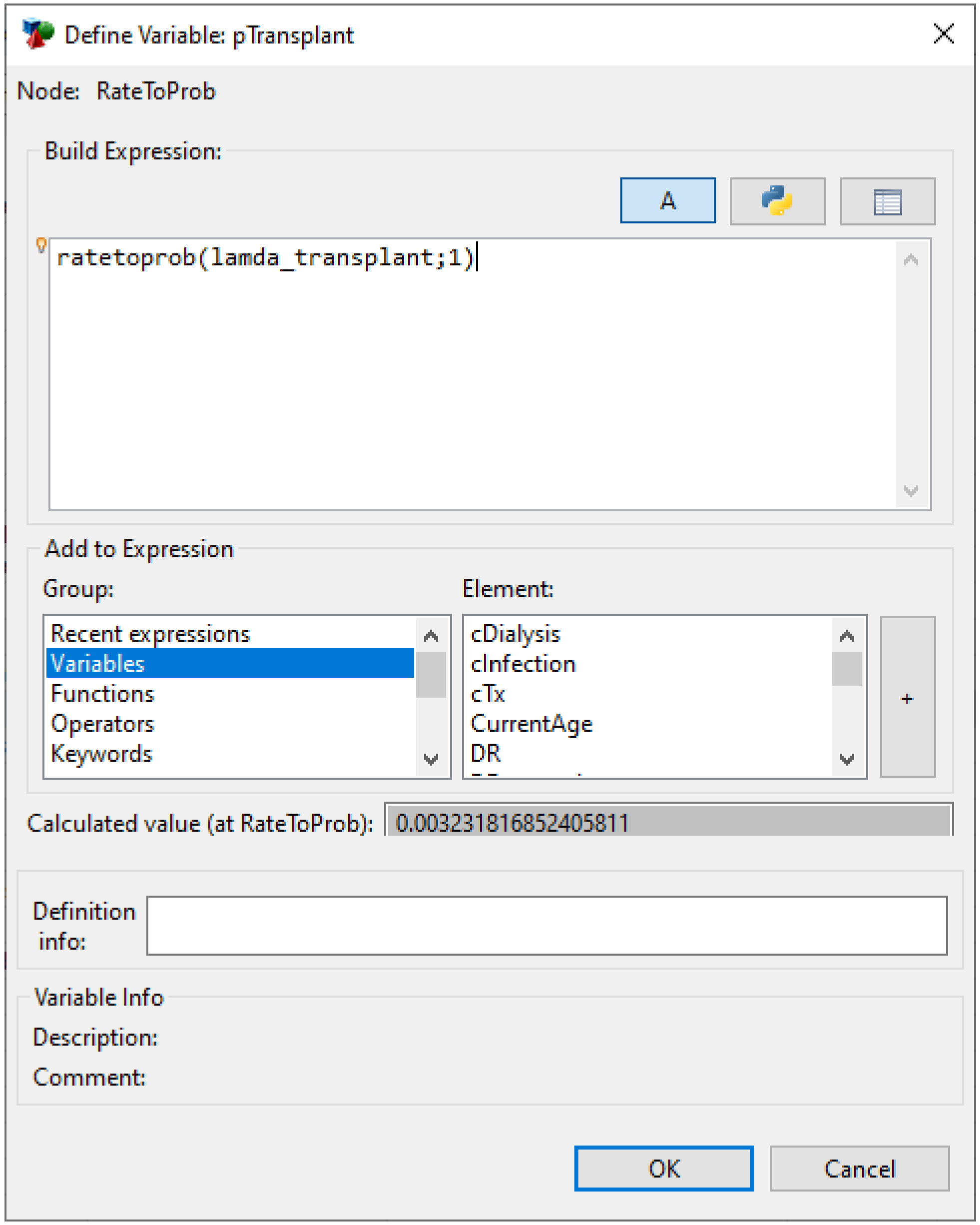
There is a built in TreeAge function for this:

ratetoprob(λ;1)

Note that λ here is the rate per cycle



Example of a time-to-event probability


Define Variable: pTransplant
Node: RateToProb
Build Expression:
A
ratetoprob(lamda_transplant;1)
Add to Expression
Group:
Recent expressions
Variables
Functions
Operators
Keywords
Element:
cDialysis
cInfection
cTx
CurrentAge
DR
+
Calculated value (at RateToProb):
0.003231816852405811
Definition info:
Variable Info
Description:
Comment:
OK
Cancel

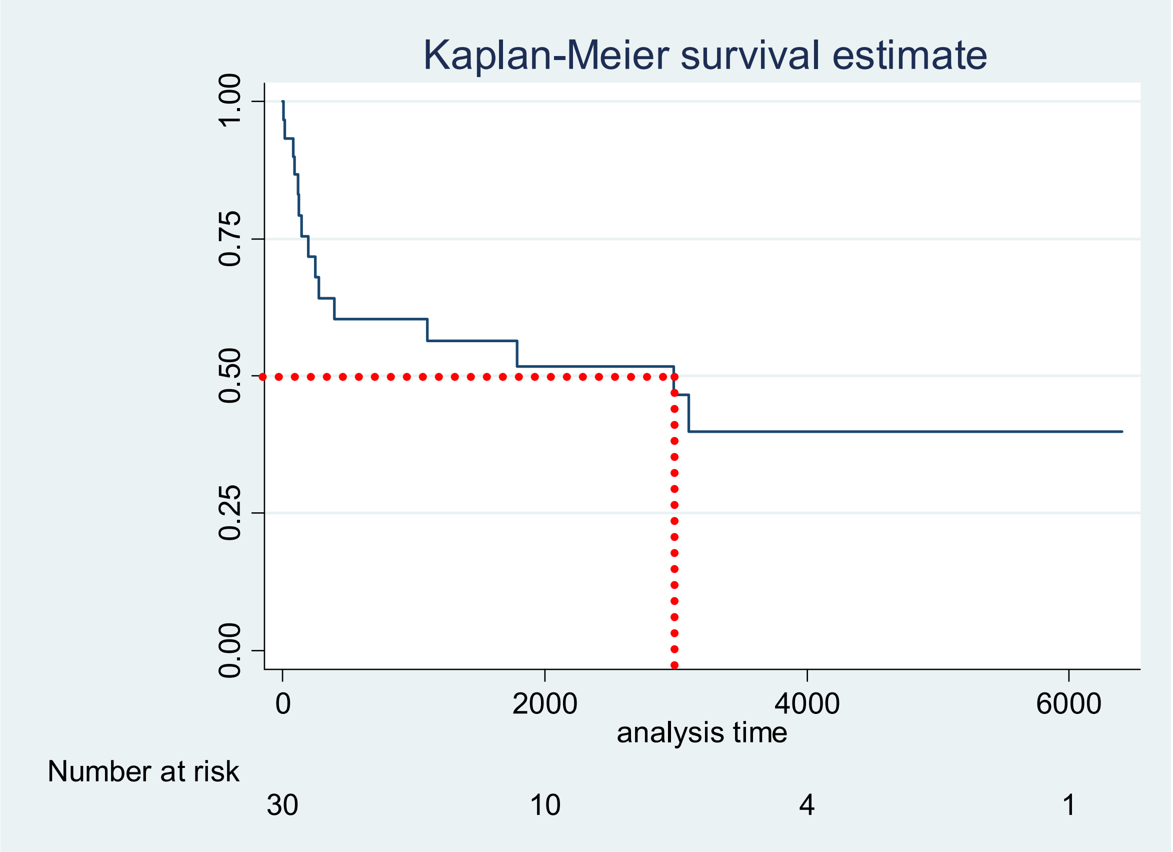
How to estimate the hazard rate, λ

- Method 1: use mean survival
  - If a cohort is observed to 100% having the event
  - Assume an exponential distribution
  - Rate (λ) is the reciprocal of the mean survival time
  - e.g. if the mean is 8000 days = 22 years
  - Then the hazard rate, $\lambda = 1/22 = 0.046$ deaths/year

Method 2: use median survival

Median survival is about 3000 days or 8.213 years

$$\lambda = -\ln[1-S(t)]/t^*$$
$$= -\ln(.5)/8.213$$
$$= 0.085 \text{ deaths / year}$$



*Rearrange $S(t) = \exp(-\lambda t)$

Method 3: Run an exponential model

```
. streg, d(exp) nohr

         failure _d:  dead
   analysis time _t:  followup/365.25

Iteration 0:   log likelihood = -57.014599
Iteration 1:   log likelihood = -57.014599  (backed up)

Exponential regression -- log relative-hazard form

No. of subjects =           30                     Number of obs   =        30
No. of failures =           15
Time at risk    =  142.4229979
                                                   LR chi2(0)      =      0.00
Log likelihood  =   -57.014599                     Prob > chi2     =         .

------------------------------------------------------------------------------
          _t |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       _cons |  -2.250751   .2581989    -8.72   0.000    -2.756812   -1.744691
------------------------------------------------------------------------------

. di exp(b[1,1])
.10532007
```

$\ln(\lambda)$, $\lambda = \exp(-2.25) = 0.105$

Method 4: use cumulative probability (cdf):

E.g. if the 3 year cumulative probability of the event is 11%, i.e. the cdf(3) = 0.11

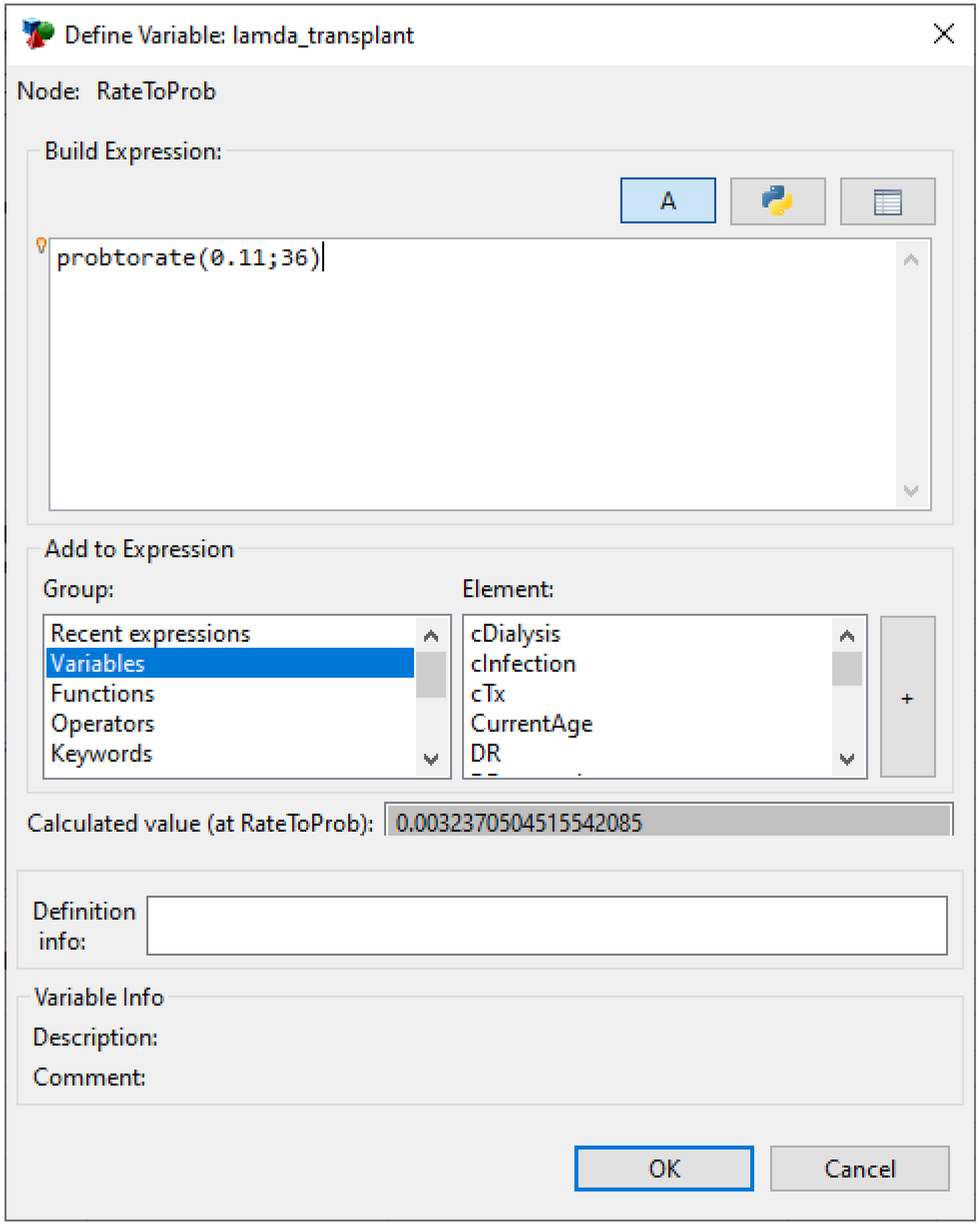
cdf(3) = 0.11 = 1 – exp(-λ3); λ = -ln[1-0.11]/3

TreeAge has a built in function for this:

If the cycle length is a month then

λ = probtorate(0.11; 36)

Per cycle probability is then ratetoprob(λ;1)


Define Variable: lamda_transplant
Node: RateToProb
Build Expression:
A
probtorate(0.11;36)
Add to Expression
Group:
Element:
Recent expressions
Variables
Functions
Operators
Keywords
cDialysis
cInfection
cTx
CurrentAge
DR
+
Calculated value (at RateToProb):
0.0032370504515542085
Definition info:
Variable Info
Description:
Comment:
OK
Cancel

Let's build a CKD Markov employing rates and probabilities

- Build a 1-strategy Markov model where all patients start in dialysis with transplant (Tx) and death states considered.
- Cycle length = 1 month, DR = 1.5%/yr, no HCC/WCC, terminate if age > 100 years or the number of QALMs generated by the cohort per cycle is < 0.001
- Only 1 Tx is allowed. Write an expression for pTransplant using probtorate() and ratetoprob() functions from a study where the 3-year cumulative probability for dialysis patients receiving a transplant was 11% assuming an exponential distribution of transplant times

Let’s build a CKD Markov employing rates and probabilities

- In the transition from dialysis to Tx, consider the possibility of immediate transplant failure (primary graft loss). Use probfactor() for the probability of primary graft loss, baseline 0.025, OR = 2.1 if current age >= 55
- Also, model a transient infection event (given that primary graft loss doesn’t happen) with a one-time cost and disutility using transition rewards
- Transplants may fail due to chronic graft loss requiring a return to dialysis. These follow an exponential distribution at a rate of 0.005 per month
- After 120 months, risk of Tx failure due to chronic graft loss = 0 (use a tunnel state and _tunnel counter)

The end.