In[●]:= `DSolve[n '[t] == r - (b + a n[t]) / pop, n[t], t]`
  求解微分方程

Out[●]= $\left\{\left\{n[t] \to \dfrac{-b + pop\ r}{a} + e^{-\frac{a\,t}{pop}}\,c_1\right\}\right\}$

In[●]:= `Solve[` $\dfrac{-b + pop\ r}{a} + e^{-\frac{a\,t}{pop}}\,const == 2,\ const$ `]`
  求解

Out[●]= $\left\{\left\{const \to \dfrac{e^{\frac{a\,t}{pop}}\,(2\,a + b - pop\ r)}{a}\right\}\right\}$

In[●]:= `Simplify[` `DSolve[` $-\left(\dfrac{-b + pop\ r}{a} + e^{-\frac{a\,t}{pop}}\,\dfrac{(2\,a + b - pop\ r)}{a} - 1\right) \Big/ pop * f[t] == f\,'[t],\ f[t],\ t$ `]]`
  化简   求解微分方程

Out[●]= $\left\{\left\{f[t] \to e^{\frac{e^{-\frac{a\,t}{pop}}(2\,a+b-pop\ r)+\frac{a\,(a+b-pop\ r)\,t}{pop}}{a^2}}\,c_1\right\}\right\}$

In[●]:= $e^{\frac{e^{-a\,tau}\,(2\,a+b-rho)+a\,(a+b-rho)\,tau}{a^2}}\,c_1$ `/. {tau → 0}`

Out[●]= $e^{\frac{2\,a+b-rho}{a^2}}\,c_1$

In[●]:= `Simplify[` `-D[` $e^{\frac{e^{-a\,tau}\,(2\,a+b-rho)+a\,(a+b-rho)\,tau}{a^2}}\,e^{-\frac{2\,a+b-rho}{a^2}},\ tau$ `]]`
  化简   偏导

Out[●]= $-\dfrac{e^{\frac{-2\,a-b+e^{-a\,tau}\,(2\,a+b-rho)+rho+a\,(a+b-rho)\,tau}{a^2}}\,\left(a + b - rho + e^{-a\,tau}\,(-2\,a - b + rho)\right)}{a}$

In[ ]:= `Manipulate`[`Plot`[{$-\dfrac{e^{\frac{-2\,a-b+e^{-a\,tau}\,(2\,a+b-rho)+rho+a\,(a+b-rho)\,tau}{a^2}}\left(a+b-rho+e^{-a\,tau}\,(-2\,a-b+rho)\right)}{a}$,

　　交互式操作　　　繪圖

$-$`Exp`$[-tau-rho/2*tau^2](-1-rho\,tau)$}, {tau, 0, 2},

　　　　指數形式

`PlotRange` $\to$ {0, 5}], {rho, 0, 50}, {a, 0.01, 5}, {b, 0, 5}]

　繪製範圍

Out[ ]=
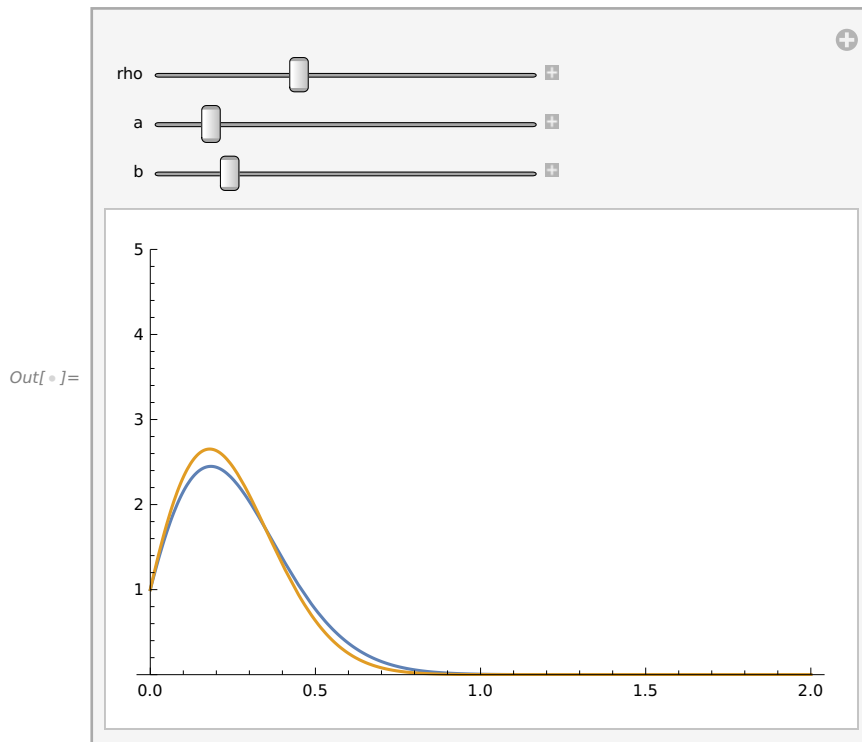
Define the function that convert numbers to the format in file names

In[ ]:= `number2Printed[number_]` := `Module`[{returnedString = "e", foo, bar, idx, oom},

　　　　　　　　　　　　　　模組

`If`[number == 1, `Return`["1.0e+00"], `If`[number == 0, `Return`["0.0e+00"],

　如果　　　　　　　返回　　　　　　　　如果　　　　　　　返回

`If`[number < 1,

　如果

`For`[idx = 1, `StringLength`[returnedString] == 1, idx = idx + 1,

　For迴圈　　　　　字串長度

foo = `Floor`[number / 10 ^ (- idx)];

　　　　　弱取整

`If`[foo == 0, ,

　如果

```
    bar = Round[(number - foo * 10 ^ (-idx)) / 10 ^ (-idx - 1)];
           捨入

    If[StringLength[ToString[idx]] == 1,
    …  字串長度          轉換成字串

      returnedString = StringJoin[ToString[foo],
                       字串結合      轉換成字串

        ".", ToString[bar], returnedString, "-0", ToString[idx]],
             轉換成字串                              轉換成字串

      returnedString = StringJoin[ToString[foo], ".", ToString[bar],
                       字串結合      轉換成字串            轉換成字串

        returnedString, "-", ToString[idx]]
                             轉換成字串

     ]
    ]
  ];
  Return[returnedString]
  返回

  ,
  oom = (StringLength[ToString[DecimalForm[Floor[number] * 1.]]] - 2);
         字串長度      轉換成字串 十進位形式    弱取整

  foo = Floor[number / 10 ^ oom];
        弱取整

  bar = Round[(number - foo * 10 ^ oom) / 10 ^ (oom - 1)];
        捨入

  If[StringLength[ToString[oom]] == 1,
  …  字串長度          轉換成字串

    returnedString = StringJoin[ToString[foo],
                     字串結合      轉換成字串

      ".", ToString[bar], returnedString, "+0", ToString[oom]],
           轉換成字串                              轉換成字串

    returnedString = StringJoin[ToString[foo], ".", ToString[bar],
                     字串結合      轉換成字串            轉換成字串

      returnedString, "+", ToString[oom]]
                           轉換成字串

  ];
  Return[returnedString]
  返回

 ];
]
]
]
```

Import parameters and data

```
In[ ]:= (* combinedParameters= Interpreter[DelimitedSequence["Number",{"[",", ","]"}]][
                         解譯器        分隔序列               數

     Import[StringJoin[NotebookDirectory[],"combined_parameters.txt"]]];
     導入    字串結合    筆記本目錄

   sequenceLengths= Interpreter[DelimitedSequence["Number",{"[",", ","]"}]][
                        解譯器        分隔序列               數

     Import[StringJoin[NotebookDirectory[],"sequence_lengths.txt"]]];
     導入    字串結合    筆記本目錄

   populationSizes= Interpreter[DelimitedSequence["Number",{"[",", ","]"}]][
                        解譯器        分隔序列               數

     Import[StringJoin[NotebookDirectory[],"population_sizes.txt"]]]; *)
     導入    字串結合    筆記本目錄

   combinedParameters = ToExpression[StringReplace[Import[
                         轉換成表示式      字串替代      導入

       StringJoin[NotebookDirectory[], "combined_parameters.txt"]], {"[" → "{", "]" → "}"}]];
       字串結合    筆記本目錄

   sequenceLengths = ToExpression[StringReplace[Import[
                      轉換成表示式      字串替代      導入

       StringJoin[NotebookDirectory[], "sequence_lengths.txt"]], {"[" → "{", "]" → "}"}]];
       字串結合    筆記本目錄

   populationSizes = ToExpression[StringReplace[Import[
                      轉換成表示式      字串替代      導入

       StringJoin[NotebookDirectory[], "population_sizes.txt"]], {"[" → "{", "]" → "}"}]];
       筆記本目錄

In[ ]:= histograms = Table[Table[Table[Transpose[
                      表格    表格    表格    轉置

         ToExpression[StringReplace[Import[StringJoin[NotebookDirectory[], number2Printed[
         轉換成表示式      字串替代      導入    字串結合    筆記本目錄

            combinedParameters[[idx1]]], "_", number2Printed[sequenceLengths[[idx2]]], "_",
            number2Printed[populationSizes[[idx3]]], ".txt"]], {"[" → "{", "]" → "}"}]]
         ], {idx3, Length[populationSizes]}], {idx2, Length[sequenceLengths]}],
                        長度                                 長度
      {idx1, Length[combinedParameters]}];
             長度
```

Fit scaling factors

```
In[ ]:= predictionFree[tau_, gamma_, beta_, alpha_, combinedParameter_] :=
     (3 gamma * tau ^ 2 + beta * combinedParameter * tau + alpha) *
     Exp[- gamma * tau ^ 3 - beta * combinedParameter * tau^2 / 2 - alpha * tau]
     指數形式
```

```
In[ ]:= gammas = Table[0, {idx, Length[combinedParameters]}];

betas = Table[0, {idx, Length[combinedParameters]}];

alphas = Table[0, {idx, Length[combinedParameters]}];

Module[{fit = Table[0, {idx0, Length[combinedParameters]}]}, Do[

   fit[[idx0]] = NonlinearModelFit[

     Flatten[Table[Table[histograms[[idx0, idx1, idx2]], {idx2, Length[populationSizes]}],

       {idx1, Length[sequenceLengths]}], 2],

     predictionFree[tau, gamma, beta, alpha, combinedParameters[[idx0]]],

     {gamma, beta, alpha}, tau];
   gammas[[idx0]] = fit[[idx0]]["ParameterTable"][[1, 1, 2, 2]];
   betas[[idx0]] = fit[[idx0]]["ParameterTable"][[1, 1, 3, 2]];
   alphas[[idx0]] = fit[[idx0]]["ParameterTable"][[1, 1, 4, 2]],
   {idx0, Length[combinedParameters]}]]
```

⸱⸱⸱ General: Exp[−6525.05] is too small to represent as a normalized machine number; precision may be lost.

⸱⸱⸱ General: Exp[−6525.05] is too small to represent as a normalized machine number; precision may be lost.

⸱⸱⸱ General: Exp[−6525.05] is too small to represent as a normalized machine number; precision may be lost.

⸱⸱⸱ General: Further output of General::munfl will be suppressed during this calculation.

```
In[ ]:= newPredictionFree[tau_, a_, b_, rho_] :=
```

$$-\frac{e^{\frac{-2a-b+e^{-a\,tau}\,(2a+b-rho)+rho+a\,(a+b-rho)\,tau}{a^2}}\,\left(a+b-rho+e^{-a\,tau}\,(-2a-b+rho)\right)}{a}$$

```
In[ ]:= zetas = Table[0, {idx, Length[combinedParameters]}];

xis = Table[0, {idx, Length[combinedParameters]}];

Module[{fit = Table[0, {idx0, Length[combinedParameters]}]}, Do[

    fit[[idx0]] = NonlinearModelFit[

      Flatten[Table[Table[histograms[[idx0, idx1, idx2]], {idx2, Length[populationSizes]}],

        {idx1, Length[sequenceLengths]}], 2],

      newPredictionFree[tau, zeta, xi, combinedParameters[[idx0]]], {zeta, xi}, tau];
    zetas[[idx0]] = fit[[idx0]]["ParameterTable"][[1, 1, 2, 2]];
    xis[[idx0]] = fit[[idx0]]["ParameterTable"][[1, 1, 3, 2]],
    {idx0, Length[combinedParameters]}]]
```

⚫⚫⚫ General: Exp[−1107.4] is too small to represent as a normalized machine number; precision may be lost.

⚫⚫⚫ General: Exp[−976.056] is too small to represent as a normalized machine number; precision may be lost.

⚫⚫⚫ General: Exp[−1107.4] is too small to represent as a normalized machine number; precision may be lost.

⚫⚫⚫ General: Further output of General::munfl will be suppressed during this calculation.

```
In[ ]:= With[{idx0 = 7}, Show[ListPlot[
          Flatten[Table[Table[histograms[[idx0, idx1, idx2]], {idx2, Length[populationSizes]}],
            {idx1, Length[sequenceLengths]}], 1], ImageSize → Full, PlotRange → All],
        Plot[{predictionFree[tau, gammas[[idx0]], betas[[idx0]], alphas[[idx0]],
            combinedParameters[[idx0]]], predictionFree[tau, 0, 1, 1, combinedParameters[[idx0]]],
           newPredictionFree[tau, zetas[[idx0]], xis[[idx0]], combinedParameters[[idx0]]]},
          {tau, 0, Transpose[Max[histograms[[idx0, 1, 1]]]][[1]]}, PlotRange → All,
          PlotStyle → Thick, AxesLabel → {"τ(N×gen)", "P(τ)(1/N/gen)"},
          PlotLegends → Placed[{"old", "bad", "new"}, Below]]]]
```
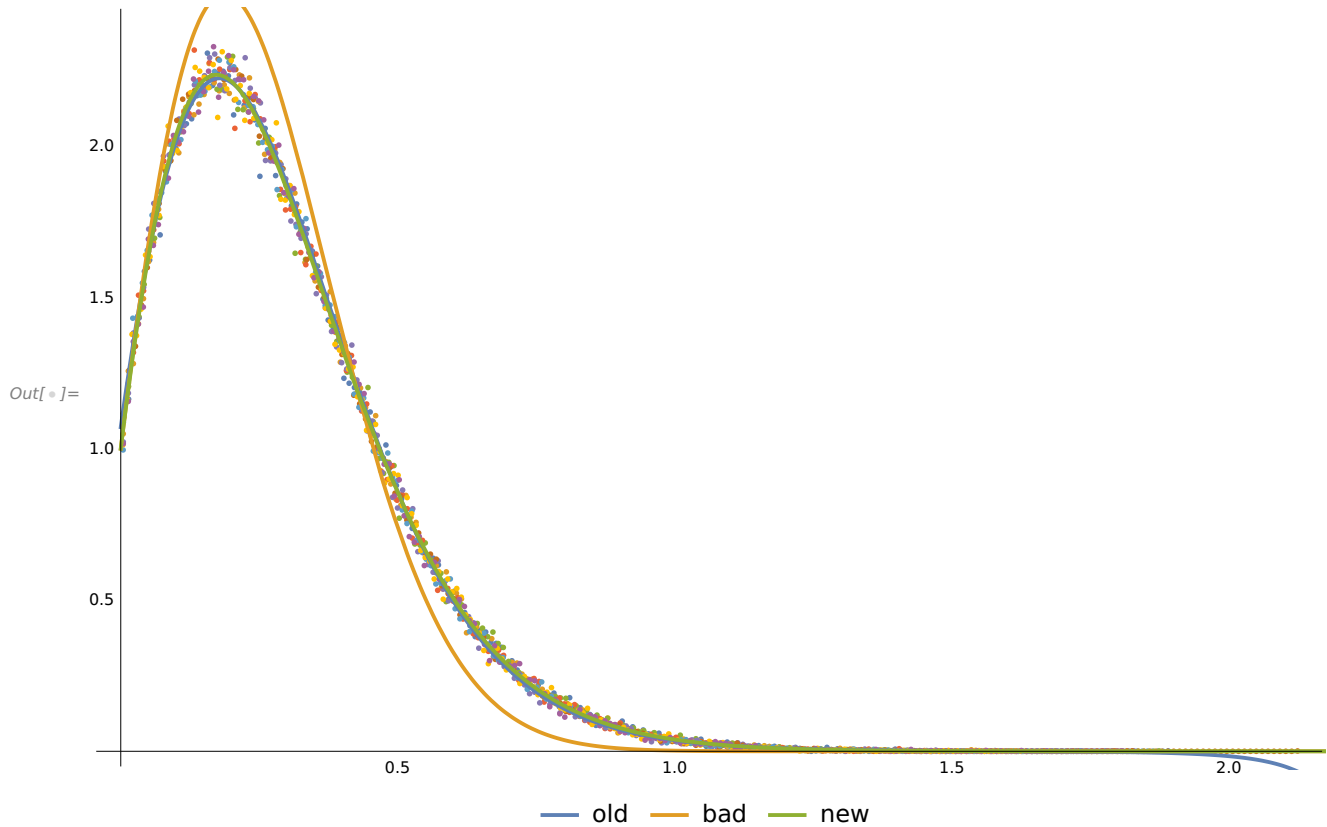
In[●]:= `ListPlot[Transpose[{combinedParameters, zetas}]]`
　　　`⌊點集圖　　⌊轉置`

Out[●]=
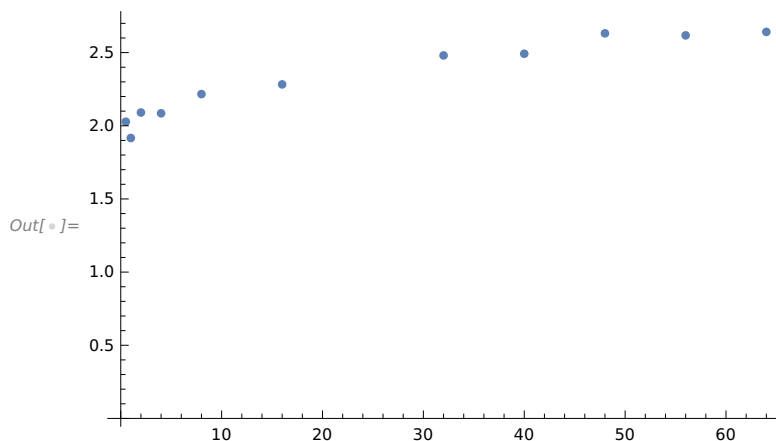
In[●]:= `ListPlot[Transpose[{combinedParameters, xis}]]`
　　　`⌊點集圖　　⌊轉置`

Out[●]=
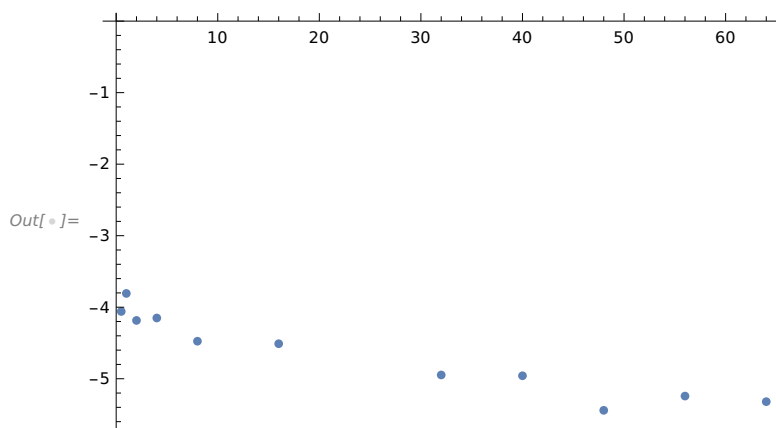
In[●]:= `Print[zetas, xis]`
　　　`⌊列表`

`{-0.721406, 2.02774, 1.91637, 2.09056, 2.08521,`
`  2.21654, 2.2827, 2.48073, 2.49219, 2.63157, 2.61782, 2.64158}`
`{1.44222, -4.0596, -3.80729, -4.18477, -4.15076, -4.47539, -4.51038,`
`  -4.94735, -4.95828, -5.44162, -5.24234, -5.32046}`