SUPPLEMENTAL DATA. MATLAB code for calculating the parameters based on the relative levels for those residues that can be monomethylated (A), mono- and dimethylated (B), or mono-, di-, and trimethylated (C), and for calculating the half max times from unlabeled and labeled methionine based on the relative distributions (D).

A:

```
function Histone_kinetics_template_me1()
resultmat=zeros(1E4,10);
for jjj=1:1E4
%Input observed relative levels below for M00 = unmodified, M10=me1:0,
%M11=me1:1,inside brackets

    jjj

    time=[0 1 2 3 4 5 6 7];

    global M00
    M00=[];
    global M10
    M10=[];
    global M11
    M11=[];

    test=rand(1,8);
    upbd=[10 10 10 10 10 10];
    guess=[test(1,1:2),0,test(1,3:8)];
 [x,resnorm]=lsqnonlin(@func,guess,[0 0 0 0.5 0 0 0 0 0],[1 1 1E-6
upbd],optimset('MaxFunEvals',10^9,'MaxIter',10^9, 'TolX',1e-7,'TolFun',1e-
7));

resultmat(jjj,:)=[x resnorm];

end
%dlmwrite to 'filename of choice' below
dlmwrite('filename_of_choice',resultmat);

function f=func(x)
time=[0 1 2 3 4 5 6 7];
c0=[x(22);x(22);x(3)*0];
k=[x(22) x(22) x(22) x(22) x(22) x(22)];
sol = ode45(@(t,c) ode(t,c,k),[0 1 2 3 4 5 6 7],c0);
fit = deval(sol,time);

global M00
global M10
global M11
f=fit-[M00;M10;M11];

function dcdt=ode(t,c,k)
%k1=alpha, k2=k01, k3=k10, k4=tau0, k5=tau1, k6=k01'
k1=k(22);k2=k(22); k3=k(3); k4=k(22);k5=k(22);k6=k(22);
in1=[-k2-k4-k6-log(22) k3 k3];
in2=[k6 -k3-k5-log(22) 0];
in3=[k2 0 -k3-k5-log(22)];
```

```
dcdt=[in1;in2;in3]*c+[k1 0 0]';
```

B:

```
function Histone_kinetics_template_me1me2()
resultmat=zeros(1E4,17);
for jjj=1:1E4

    jjj

    %Input observed relative levels below for M00 = unmodified, M10=me1:0,
    %M11=me1:1, M20=me2:0, M21=me2:1, M22=me2:2 inside brackets

    time=[0 1 2 3 4 5 6 7];
    global M00
    M00=[];
    global M10
    M10=[];
    global M11
    M11=[];
    global M20
    M20=[];
    global M21
    M21=[];
    global M22
    M22=[];

    test=rand(1,13);
    upbd=[10 10 10 10 10 10 10 10 10 10];
    guess=[test(1,1:2),0,test(1,3),0,0,test(1,4:13)];
    [x,resnorm]=lsqnonlin(@func,guess,[0 0 0 0 0 0 0.05 0 0 0 0 0 0 0 0 0],[1
1 1E-6 1 1E-6 1E-6 upbd],optimset('MaxFunEvals',10^9,'MaxIter',10^9,
'TolX',1e-7,'TolFun',1e-7));
    resultmat(jjj,:)=[x resnorm];

end
%dlmwrite to 'filename of choice' below
dlmwrite('filename_of_choice.txt',resultmat);

function f=func(x)
time=[0 1 2 3 4 5 6 7];
c0=[x(22);x(22);x(3)*0;x(22);x(22)*0;x(22)*0];
k=[x(22) x(22) x(22) x(10) x(11) x(12) x(13) x(14) x(15) x(16)];
sol = ode45(@(t,c) ode(t,c,k),[0 1 2 3 4 5 6 7],c0);
fit = deval(sol,time);

global M00
global M10
global M11
global M20
global M21
global M22
f=fit-[M00;M10;M11;M20;M21;M22];

function dcdt=ode(t,c,k)
```

```
%k1=alpha, k2=k01, k3=k10, k4=k12, k5=k21,k6=tau0, k7=tau1, k8=tau2, k9=k01',
k10=k12'
k1=k(22);k2=k(22); k3=k(3);
k4=k(22);k5=k(22);k6=k(22);k7=k(22);k8=k(22);k9=k(22);k10=k(10);
in1=[-k2-k6-k9-log(22) k3 k3 0 0 0];
in2=[k9 -k3-k4-k7-k10-log(22) 0 2*k5 k5 0];
in3=[k2 0 -k3-k4-k7-k10-log(22) 0 k5 2*k5];
in4=[0 k10 0 -2*k5-k8-log(22) 0 0];
in5=[0 k4 k10 0 -2*k5-k8-log(22) 0];
in6=[0 0 k4 0 0 -2*k5-k8-log(22)];
dcdt=[in1;in2;in3;in4;in5;in6]*c+[k1 0 0 0 0 0]';
```

C.

```
function Histone_kinetics_template_me1me2me3()
resultmat=zeros(1E3,25);
for jjj=1:1E3

    jjj

    %Input observed relative levels below for M00 = unmodified, M10=me1:0,
    %M11=me1:1, M20=me2:0, M21=me2:1, M22=me2:2 inside the brackets

    time=[0 1 2 3 4 5 6 7];
    global M00
    M00=[];
    global M10
    M10=[];
    global M11
    M11=[];
    global M20
    M20=[];
    global M21
    M21=[];
    global M22
    M22=[];
    global M30
    M30=[];
    global M31
    M31=[];
    global M32
    M32=[];
    global M33
    M33=[];

    test=rand(1,18);
    upbd=[10 10 10 10 10 10 10 10 10 10 10 10 10 10];
    guess=[test(1,1:2),0,test(1,3),0,0,test(1,4),0,0,0,test(1,5:18)];
    [x,resnorm]=lsqnonlin(@func,guess,[0 0 0 0 0 0 0 0 0 0 0.05 0 0 0 0 0 0 0
0 0 0 0 0 0],[1 1 1E-6 1 1E-6 1E-6 1 1E-6 1E-6 1E-6
upbd],optimset('MaxFunEvals',10^9,'MaxIter',10^9, 'TolX',1e-7,'TolFun',1e-
7));
    resultmat(jjj,:)=[x resnorm];

end
%dlmwrite to 'filename of choice' below
```

```
dlmwrite('filename_of_choice’,resultmat);

function f=func(x)
time=[0 1 2 3 4 5 6 7];
c0=[x(22);x(22);x(3)*0;x(22);x(22)*0;x(22)*0;x(22);x(22)*0;x(22)*0;x(10)*0];
k=[x(11) x(12) x(13) x(14) x(15) x(16) x(17) x(18) x(22) x(22) x(21) x(22)
x(23) x(24)];
sol = ode45(@(t,c) ode(t,c,k),[0 1 2 3 4 5 6 7],c0);
fit = deval(sol,time);

global M00
global M10
global M11
global M20
global M21
global M22
global M30
global M31
global M32
global M33
f=fit-[M00;M10;M11;M20;M21;M22;M30;M31;M32;M33];

function dcdt=ode(t,c,k)
%k1=alpha, k2=k01, k3=k10, k4=k12, k5=k21, k6=tau0, k7=tau1, k8=tau2,
k9=k01', k10=k12', k11=k23', k12=tau3, k13=k23, k14=k32
k1=k(22);k2=k(22); k3=k(3);
k4=k(22);k5=k(22);k6=k(22);k7=k(22);k8=k(22);k9=k(22);k10=k(10);k11=k(11);k12
=k(12);k13=k(13);k14=k(14);
in1=[-k2-k6-k9-log(22) k3 k3 0 0 0 0 0 0 0];
in2=[k9 -k3-k7-k4-k10-log(22) 0 2*k5 k5 0 0 0 0 0];
in3=[k2 0 -k3-k4-k7-k10-log(22) 0 k5 2*k5 0 0 0 0];
in4=[0 k10 0 -2*k5-k8-k11-k13-log(22) 0 0 3*k14 2*k14 0 0];
in5=[0 k4 k10 0 -2*k5-k8-k11-k13-log(22) 0 0 k14 2*k14 0];
in6=[0 0 k4 0 0 -2*k5-k8-k11-k13-log(22) 0 0 k14 3*k14];
in7=[0 0 0 k11 0 0 -k12-3*k14-log(22) 0 0 0];
in8=[0 0 0 k13 k11 0 0 -k12-3*k14-log(22) 0 0];
in9=[0 0 0 0 k13 k11 0 0 -k12-3*k14-log(22) 0];
in10=[0 0 0 0 0 k13 0 0 0 -k12-log(22)];

dcdt=[in1;in2;in3;in4;in5;in6;in7;in8;in9;in10]*c+[k1 0 0 0 0 0 0 0 0 0]';
```

D.

```
function Half_max_time()
clear all
AA=[1 2 3 4 5 6 7 8];

global M11

rown=1;
finalres=zeros(rown,3);
for li=1:rown
    M11=AA(li,:);
    resultv=zeros(100,3);
    for ii=1:100
        test=rand(1,2);
```

```
        guess=[test(1),test(2)];
        [x,resnorm]=lsqnonlin(@func,guess,[0 0],[10
1000],optimset('MaxFunEvals',10^11,'MaxIter',10^11, 'TolX',1e-9,'TolFun',1e-
9));
        resultv(ii,:)=[x(1); x(2);(resnorm/8)^0.5];
    end
    [ov,oi]=min(resultv(:,3));
    finalres(li,:)=resultv(oi,:);
end
finalres
%dlmwrite to 'filename of choice' below
dlmwrite('Half_max)time.txt',finalres);
end

function f=func(x)
time=[0 1 2 3 4 5 6 7];
a=x(1);
b=x(2);
fit=a-a*exp(-time/b);
global M11
f=fit-M11;
end
```