Earth Sciences Research Journal
UNIVERSIDAD NACIONAL DE COLOMBIA

Supplementary Information B

Algorithm for modeling in RStudio

```
#------------------------------------------------------------------------
title: Modeling of the large Miocene epithermal and
porphyry gold deposits of Colombia using Monte Carlo
simulations
subtitle: Modified from Chiaradia & Caricchi, (2017)
and Chiaradia (2020)
output:
  pdf_document: default
  toc: yes
  toc_depth: 5
  html_document: default
#------------------------------------------------------------------------
# LOW FLUX MODELING
## Data Input
# Time: 0 -5 Ma (Annen et al., 2006)
# Pressure: 1,5 - 9 Kbar (Annen et al., 2006)
# 5 mm/a of basaltic melt injection rate (Annen,
2009; Annen et al., 2006)
# 2 - 4% H2O in parent magma (Plank et al., 2013)
# 0,2 - 1% H2O in crustal rocks (Chiaradia &
Caricchi, 2017)
# 10-100 of fluid-melt partition coefficient of gold
(Simon et al., 2005)
# 6 – 9 ppb of gold content in calc-alkaline magmas
(Moss et al., 2001)
# Au Efficiency: Variable sensible
```{r inputs, echo = T}
#------------------------------------------------------------------------
# Random variables as inputs.
n = 3000000         # Number of iterations
set.seed(123)     # Seeds
a = runif(n, 0, 5)               # a = time in Ma
bx = runif(n, 0.5, 1)              # bx = Au Efficiency (%)
b = runif(n, 2, 4)            # c= H2O in parent magma
(wt.%)
c = runif(n, 0.2, 1)              # c = H2O in crustal
rocks(wt.%)
d = runif(n, 1.5, 9)             # d = Pressure (kbars)
e = rep(5000, n)               # e = magma rate (m/Ma)
g = runif(n, 6500, 8500)         # g = Disk radius (m)
j = runif(n, 10, 100)           # j = Kd Au
k = runif(n, 0.006, 0.009)         # K = Au total melt
(ppm)
#------------------------------------------------------------------------
# Data generating process:
#------------------------------------------------------------------------
#Crustal Melt Calculations
#------------------------------------------------------------------------
f = a*e                      # f= Magma rate time
integrated (m/Ma)
hist(f, breaks = 75)
#------------------------------------------------------------------------
#l= x crustal
l = -0.00000476*(a^{5}) + 0.00010012*(a^{4}) -
0.00087381*(a^{3}) + 0.00390394*(a^{2}) -
0.0087404*(a) + 0.00542933
hist(l, breaks = 75)
#------------------------------------------------------------------------
#m= y crustal
m = 0.00025805*(a^{5}) - 0.00451301*(a^{4}) +
0.03096005*(a^{3}) -
  0.1047359*(a^{2}) + 0.17540876*(a) - 0.08277333
hist(m, breaks = 75)
#------------------------------------------------------------------------
#nn= x crustal
nn = -0.00183672*(a^{5}) + 0.02924904*(a^{4}) -
0.1769496*(a^{3}) +
  0.50100047*(a^{2}) - 0.64552174*(a) + 0.2529533
hist(n, breaks = 75)
#------------------------------------------------------------------------
#o= crustal melt fraction Annen
o = l*(d^{2}) + m*(d) + nn
hist(o, breaks = 75)
#------------------------------------------------------------------------
#p= crustal Melt fraction Annen logical test
p = 0
for(i in 1:n){
  if(o[i] < 0){
    p[i] = 0
  }
  else{
    p[i] = o[i]
  }
}
hist(p, breaks = 75)
#------------------------------------------------------------------------
```
```

```
#q=crustal Melt fraction Annen logical test2 low
q = 0
for(i in 1:n){
  if((a[i] > 1) & (d[i] > 3)){
    q[i] = p[i]
  }
  else{
    q[i] = NA
  }
}
hist(q, breaks = 75)
#--------------------------------------------------------------------
#r=crustal Melt fraction Annen logical test3
r = 0
for(i in 1:n){
  if((a[i] > 0.25) | (d[i] > 6)){
    r[i] = q[i]
  }
  else{
    r[i] = NA
  }
}
hist(r, breaks = 75)
#--------------------------------------------------------------------
#s=volume of crustal melt fraction rate time
integrated (m^3)
s = pi*(g^{2})*(f)*r
hist(s, breaks = 75)
#--------------------------------------------------------------------
#t= tons of crustal melt fraction time integrated
t = s*(2.6)
hist(t, breaks = 75)
#--------------------------------------------------------------------
# u= crustal Melt percent Annen
u = r*100
#--------------------------------------------------------------------
# Residual Melt Calculations
#--------------------------------------------------------------------
#v= x residual
v = -0.00011309*(a^{6}) + 0.00186017*(a^{5}) -
0.01179089*(a^{4}) +
  0.03594339*(a^{3}) - 0.05316974*(a^{2}) +
0.03207473*(a) + 0.00018778
hist(v, breaks = 75)
#--------------------------------------------------------------------
#w= y residual
w = 0.00120531*(a^{6}) - 0.01970898*(a^{5}) +
0.12379992*(a^{4}) -
  0.37149275*(a^{3}) + 0.53160183*(a^{2}) -
0.28832793*(a) - 0.00343741
hist(w, breaks = 75)
#--------------------------------------------------------------------
# x= z residual
x = -0.00264884*(a^{6}) + 0.04269856*(a^{5}) -
0.26252652*(a^{4}) +
  0.7613096*(a^{3}) - 1.02505321*(a^{2}) +
0.52268727*(a) + 0.01527694
hist(x, breaks = 75)
#--------------------------------------------------------------------
#y= residual Melt fraction Annen
y = v*(d^{2}) + w*(d) + x
hist(y, breaks = 75)
#--------------------------------------------------------------------
#z= residual Melt fraction Annen logical test
z = 0
for(i in 1:n){
  if(y[i] < 0){
    z[i] = 0
  }
  else{
    z[i] = y[i]
  }
}
hist(z, breaks = 75)
#--------------------------------------------------------------------
#aa= residual Melt fraction Annen logical test2
aa = 0
for(i in 1:n){
  if((a[i] > 0.5) & (d[i] > 3)){
    aa[i] = z[i]
  }
}
#--------------------------------------------------------------------
#ab= volume of residual melt fracton time integrated
ab = pi*(g^{2})*(f)*aa
hist(ab, breaks = 75)
#--------------------------------------------------------------------
# ac: tons of residual melt fraction time integrated
ac = 2.6*ab
hist(ac, breaks = 75)
#--------------------------------------------------------------------
#ad= residual Melt percent Annen
ad = aa*100
hist(aa, breaks = 75)
#--------------------------------------------------------------------
#Total Melt Calculations
#--------------------------------------------------------------------
#ae= all_melt_percent_low
ae = u + ad
hist(ae, breaks = 75)
#--------------------------------------------------------------------
#af= Total melt in tons
af = ac + t
#--------------------------------------------------------------------
#ag= Volume km^3 total melt
ag = (af/2.5)/(10^{9})
hist(af, breaks = 75)
#--------------------------------------------------------------------
#H2O Content Calculations in residual melt
#--------------------------------------------------------------------
#aj= H2O wt.% in residual melt
aj = 100*b*(ad^{-1})
#--------------------------------------------------------------------
#al= r H2O
al =0.00000163*(d^{2})+0.00002024*(d)-
0.00000718
hist(al, breaks = 75)
#--------------------------------------------------------------------
#am= s H2O
am = -0.00088884*(d^{2})-
0.0110706*(d)+0.00391827
hist(am, breaks = 75)
#--------------------------------------------------------------------
```

````
# an: t H2O
an = 0.05908101*(d^{2}) + 1.91102201*(d) + 1.79358317
hist(an, breaks = 75)
#---------------------------------------------------------------------
#ao: residual melt Annen logical test for H2O solubility (F=100-10%)
ao = 0
for(i in 1:n){
  if(is.na(ad[i]) == FALSE){
    if(ad[i] <= 10){
    ao[i] = 10
    }
  else if(ad[i] > 10){
    ao[i] = ad[i]
    }
  }
  else{
    ao[i] = NA
  }
}
hist(ao, breaks = 75)
#---------------------------------------------------------------------
# ak: log test H2O wt% in residual melt
ak = 100*(b)*(ao^{-1})
hist(ak, breaks = 75)
#---------------------------------------------------------------------
# ap: H2O solubility in melt (F =100-10 %)
ap = al*(ao^{2}) + am*ao +  an
hist(ap, breaks = 75)
#---------------------------------------------------------------------
# aq: H2O wt% in residual melt logical test 100-10% F range
aq = 0
for(i in 1:n){
  if((is.na(aj[i]) == FALSE) & (is.na(ap[i]) == FALSE)){
    if(aj[i] > ap[i]){
      aq[i] = ap[i]
    }
    else if(aj[i] <= ap[i]){
      aq[i] = aj[i]
    }
  }
  else{
    aq[i] = NA
  }
}
#---------------------------------------------------------------------
# ar: Excess residual H2O
ar = ak - ap
hist(ar, breaks = 75)
#---------------------------------------------------------------------
# at: Excess H2O residual (tons)
at = (ac*ar)/100
hist(at, breaks = 75)
#---------------------------------------------------------------------
# au: Log test excess residual H2O tons
au = 0
# =SI(AT3>0,AT3,0)
for(i in 1:n){
  if(is.na(at[i]) == FALSE){
    if(at[i] > 0){
      au[i] = at[i]
    }
    else{
      au[i] = 0
    }
  }
  else{
    au[i] = 0
  }
}
hist(au, breaks = 75)
#---------------------------------------------------------------------
# as: H2O residual (tons)
as  = (ac-au)*aq/100
#---------------------------------------------------------------------
## H2O Content Calcutation in Crustal Partial Melt
```{r pink, echo = T, eval = T}
#---------------------------------------------------------------------
# av: H2O wt% in partial melt
av = 100*(c)*(u^{-1})
#---------------------------------------------------------------------
# aw : H2O solubility (wt.%) with Annen crustal melt %
aw = (al*(u^{-2})) + (am*u) + an
#---------------------------------------------------------------------
# ax: H2O solubility (wt.%) with Annen crustal melt % logical test
ax = 0
for(i in 1:n){
  if((is.na(av[i]) == FALSE) & (is.na(aw[i]) == FALSE)){
    if(av[i] > aw[i]){
      ax[i] = av[i]
    }
    else if(av[i] <= aw[i]){
      ax[i] = aw[i]
    }
  }
  else{
    aq[i] = NA
  }
}
#---------------------------------------------------------------------
# AY: Excess H2O partial melt
ay = av - aw
hist(ay, breaks = 75)
#---------------------------------------------------------------------
# ba: Excess H2O crustal (tons)
ba = t*ay/100
#---------------------------------------------------------------------
# BB: log test excess crustal H2O tons
# =SI(BA2>0,BA2,0)
bb = 0
for(i in 1:n){
  if((ba[i] > 0) & (is.na(ba[i])==FALSE)){
    bb[i] = ba[i]
  }
  else{
    bb[i] = 0
  }
}
````
````

````
hist(bb, breaks = 75)
#----------------------------------------------------------------------
#az = H2O crustal (tons)
az = (t-bb)*ax/100
#----------------------------------------------------------------------
#ah= M res hydrous melt Ton
ah = ac-au+t-bb
#----------------------------------------------------------------------
#ai=Volume Km^3 hydrous melt
ai = (ah/2.5)/1000000000
#----------------------------------------------------------------------
```
## H2O Content calculations in total melt
```{r violeta, echo = T, eval = T}
#----------------------------------------------------------------------
# bc: H2O total in melt (tons)
bc = as + az
#----------------------------------------------------------------------
# bd: H2O% in hydrous melt
bd = bc/ah*100
#----------------------------------------------------------------------
# be: exsolvable H2O tons
be = as + az
#----------------------------------------------------------------------
# bf: Excess H2O total in melt (tons)
bf = at + ba
hist(bf, breaks = 75)
#----------------------------------------------------------------------
# bg: Excess total H2O NEW log test
bg = au + bb
hist(bg, breaks = 75)
#----------------------------------------------------------------------
# bh: M melt - excess H20 tons
bh = ac - au + t + bb
hist(bh, breaks = 75)
#----------------------------------------------------------------------
# bi: % H2O excess
bi = af - bh
hist(bi, breaks = 75)
#----------------------------------------------------------------------
# bj: Moles of excess H2O
bj = (bf*1000000)/(2*1.0079 + 15.99)
hist(bj, breaks = 75)
#----------------------------------------------------------------------
## Gold endowment calculations
```{r amarillo, echo = T, eval = T}
#----------------------------------------------------------------------
# bk: Au tot ton
bk = (k*af)/(10^{6})
hist(bk, breaks = 75)
#----------------------------------------------------------------------
# bl: Au ppm res melt
bl = bk/(j*bg + (af - bg))*1000000
hist(bl, breaks = 75)
#----------------------------------------------------------------------
# bm: Au ppm fluid
bm = j*bl
hist(bm, breaks = 75)
#----------------------------------------------------------------------
# bn: Au tot res melt Mt
bn = bl*(af-bg)/(1000000*1000000)
hist(bn, breaks = 75)
#----------------------------------------------------------------------
# bo: Au tot in excess fluid Mt
bo = bm*bg/(1000000*1000000)
hist(bo, breaks = 75)
#----------------------------------------------------------------------
# bp: 50% Au in excess fluid Mt
bp = bo/2
hist(bp, breaks = 75)
#----------------------------------------------------------------------
# bq: Au total Mt
bq = bn + bo
hist(bq, breaks = 75)
#----------------------------------------------------------------------
# br: Au tot ton
br = bl*ah/1000000
hist(br, breaks = 75)
#----------------------------------------------------------------------
# bs: Au tot ton
bs =br/(j*be+(ah-be))*1000000
#----------------------------------------------------------------------
# bt: Au fluid
bt = j*bs
#----------------------------------------------------------------------
# bu: Au tot res melt Mt
bu = bs*(ah-be)/(1000000*1000000)
#----------------------------------------------------------------------
# bv: Au tot in exsolvable fluid t
bv = bt*be/(1000000)
#----------------------------------------------------------------------
# bw: Au tot ton
bw =(bu+bv)*1000000
#----------------------------------------------------------------------
# by: Gold Endowment in  tons
by =bv*bx
hist(by, breaks = 75)
#----------------------------------------------------------------------
### Graphics:
```{r}
gold <- data.frame(cbind(by, ai, a, bx))
gold %>%
  ggplot(aes(x = by, y = ai, colour = a)) +
  geom_point(aes(colour = a)) +
  xlab("Gold Endowment (tons)") +
  ylab("Volume hydrous melt (Km^3)") +
  scale_colour_gradientn(colours = terrain.colors(n)) +
  geom_vline(xintercept = 0, color = "red") +
  geom_vline(xintercept = 1000, color = "red")

gold %>%
  ggplot(aes(x=by)) +
  geom_histogram() +
  xlab("Gold endowment (tons)") +
  geom_vline(xintercept = 0, color = "red") +
  ylab("Frequency") + geom_vline(xintercept = 1000, color = "red")
gold %>%
  ggplot(aes(x = by, y = ai, colour = bx)) +
  geom_point(aes(colour = bx)) + xlab("Gold Endowment (tons)") +
  ylab("Volume hydrous melt (Km^3)") +
````

```
  scale_colour_gradientn(colours = terrain.colors(n)) +
  geom_vline(xintercept = 0, color = "red") +
  geom_vline(xintercept = 1000, color = "red")
#----------------------------------------------------------------------
# HIGH FLUX
## Data Input
# Time: 40 Ka- 200 Ka (Annen, 2009)
# Pressure: 0,1-2 Kbar (Sanchez-Alfaro et al., 2016)
# 50 mm/a of basaltic melt injection rate (Annen, 2009; Annen et al., 2006)
# 2 - 4% H2O in parent magma (Plank et al., 2013)
# 0,2 - 1% H2O in crustal rocks (Chiaradia & Caricchi, 2017)
# 10-100 of fluid-melt partition coefficient of gold (Simon et al., 2005)
# 6 – 9 ppb of gold content in calc-alkaline magmas (Moss et al., 2001)
# Au Efficiency
#----------------------------------------------------------------------
```{r high, echo = T}
# High flux:
# # Random variables as inputs.
N = 3000000         # Number of iterations
set.seed(123)     # Seeds
A = runif(N, 40000, 200000)              # A Time in years
BD = runif(N, 1, 2)                 # BD = Au Efficiency (%)
B = runif(N, 2, 4)                  # B = H2O in parent magma (%)
C = runif(N, 0.2, 1)                # C = H2O in crustal rocks (%)
D = runif(N, 0.3, 3)                     # D = Pressure (kbars)
E = rep(0.007, N)                    # E = Magma rate (Km3/Y)
FF = runif(N, 10, 100)               # F = Kd Au
G = runif(N, 0.006, 0.009)                # G = Au total (ppm)
V = rep(27, N)                  #V= random dacite
#----------------------------------------------------------------------
# Data generating process:
#----------------------------------------------------------------------
H = 0.00657*A-338.52065    #H = Volume of mobile magma Km^3
hist(H, breaks = 75)
#----------------------------------------------------------------------
#I= % mobile melt
I = runif(N, 0.61, 0.8)
#----------------------------------------------------------------------
#J=Volume melt km3
J = H*I
#----------------------------------------------------------------------
#K= Vol melt m3
K = J*1000000000
#----------------------------------------------------------------------
#L= Mass melt (tons)
L = K*2.5
#----------------------------------------------------------------------
#M= Volume magma injected
M = 0.01566*A+31.01613
hist(M, breaks = 75)
#----------------------------------------------------------------------
#NN= Volume injection m^3
NN = M*1000000000
#----------------------------------------------------------------------
#O= mass injection
O = NN*2.5
#----------------------------------------------------------------------
#P= % melt
P = L/O*100
hist(P, breaks = 75)
#----------------------------------------------------------------------
#Q= a Pressure high flux
Q = 0.0000000000212*P^{5} - 0.00000000358966*P^{4} + 0.0000002281984*P^{3} - 0.00000547616979*P^{2} + 0.00000241918607*P - 0.00368596666626
#----------------------------------------------------------------------
#R= b Pressure high flux
R = -0.00002265*P^{2} + 0.0063569*P + 0.46599948
#----------------------------------------------------------------------
#S= c Pressure high flux
S = 0.000026409*P^{2} - 0.014169134*P - 0.750571377
#----------------------------------------------------------------------
#W= residual melt Annen logical test for H2O solubility (F=100-10%)
W = I*V
hist(W, breaks = 75)
#----------------------------------------------------------------------
#TT= P saturation
TT = Q*W^{2} + R*W + S
#----------------------------------------------------------------------
#U = Delta P
U = D - TT
hist(U, breaks = 75)
#----------------------------------------------------------------------
#H2O Content calculations in total melt
```{r high, echo = T}
#----------------------------------------------------------------------
#X= H2O wt% in residual melt
X = 100*B*30^{-1}
hist(X, breaks = 75)
#----------------------------------------------------------------------
#Y= log test H2O wt% in residual melt
Y= 100*B*W^{-1}
#----------------------------------------------------------------------
#Z= a H2O high Flux
Z = 0.00000163*D^{2} +0.00002024*D - 0.00000718
#----------------------------------------------------------------------
#AA= b H2O high flux
AA = -0.00088884*D^{2} - 0.0110706*D + 0.00391827
#----------------------------------------------------------------------
#AB= b H2O high flux
AB = 0.05908101*D^{2} + 1.91102201*D + 1.79358317
#----------------------------------------------------------------------
```

````
#AC = H2O solubility in melt (F =100-10 %)
AC = Z*W^{2} + AA*W + AB
hist(AC, breaks = 75)
#-----------------------------------------------------------------------
#AD= H2O wt% in melt logical test 100-10% F range
AD = 0
for(i in 1:N){
    if(X[i] > AC[i]){
      AD[i] = AC[i]
    }
    else if(X[i] <= AC[i]){
      AD[i] = X[i]
    }
  }
#-----------------------------------------------------------------------
#AE= Excess H2O %
AE = Y - AC
#-----------------------------------------------------------------------
#AG= Excess H2O (tons)
AG = (K*2.5)*AE/100
#-----------------------------------------------------------------------
#AH = Excess H2O (tons)
AH = 0
for(i in 1:N){
  if((AG[i] > 0) & (is.na(AG[i])==FALSE)){
    AH[i] = AG[i]
  }
  else{
    AH[i] = 0
  }
}

#-----------------------------------------------------------------------
#AI = new total hydrous melt
AI = (K*2.5)-AH
#-----------------------------------------------------------------------
#AF= H2O residual (tons)
AF = AI*AD/100
#-----------------------------------------------------------------------
#AJ= excess H2O ton
AJ = AE*L/100
#-----------------------------------------------------------------------
#AK = a Pressure
AK = 0.0000000000212*P^{5} - 0.00000000358966*P^{4} + 0.0000002281984*P^{3} - 0.00000547616979*P^{2} + 0.00000241918607*P - 0.00368596666626
#-----------------------------------------------------------------------
#AL = b Pressure
AL = -0.00002265*P^{2} + 0.0063569*P + 0.46599948
#-----------------------------------------------------------------------
#AM = c Pressure
AM = 0.000026409*P^{2} - 0.014169134*P - 0.750571377
#-----------------------------------------------------------------------
#AP= H2O % in melt
AP = AF/AI*100
#-----------------------------------------------------------------------
#AN = P saturation
AN = K*AP^{2} + AL*AP + AM
hist(AN, breaks = 75)
#-----------------------------------------------------------------------
#AO= Delta P
AO = D-AN
hist(AO, breaks = 75)
#-----------------------------------------------------------------------
#Gold endowment calculations
```{r high, echo = T}
#-----------------------------------------------------------------------
#AQ= Au tot ton
AQ = G*L/1000000
#-----------------------------------------------------------------------
#AR= Au ppm res melt
AR = AQ/(FF*AG+(L-AG))*1000000
#-----------------------------------------------------------------------
#AS= Au ppm fluid
AS =  FF*AR
hist(AS, breaks = 75)
#-----------------------------------------------------------------------
#AT= Au tot res melt Mt
AT = AR*(L-AG)/(1000000*1000000)
#-----------------------------------------------------------------------
#AU=Au tot in excess fluid Mt
AU = AS*AG/(1000000*1000000)
#-----------------------------------------------------------------------
#AV= 50% Au in excess fluid Mt
AV = AU/2
#-----------------------------------------------------------------------
#AW= Au tot Mt
AW = (AT+AU)
#-----------------------------------------------------------------------
#BF=M res hydrous  melt (ton)
BF = AI
hist(BF, breaks = 75)
#-----------------------------------------------------------------------
#BG=% H2O in hydrous melt
BG = Y
hist(BG, breaks = 75)
#-----------------------------------------------------------------------
#BH=Exsolvable H2O (ton)
BH = BG*BF/100
#-----------------------------------------------------------------------
#BI=Volume hidrous melt Km3
BI = AI/(2.3*1000000000)
#-----------------------------------------------------------------------
#AX=Au tot ton
AX = AR*BF/1000000
hist(AX, breaks = 75)
#-----------------------------------------------------------------------
#AY= Au res melt
AY = AX/(FF*BH+(BF-BH))*1000000
#-----------------------------------------------------------------------
#AZ=Au fluid
AZ = FF*AY
#-----------------------------------------------------------------------
#BA=Au tot res melt Mt
BA = AY*(BF-BH)/(1000000*1000000)
hist(BA, breaks = 75)
#-----------------------------------------------------------------------
#BB=Au tot in exsolvable fluid t
````
````

```
BB = AZ*AG/(1000000)
hist(BB, breaks = 75)
#-----------------------------------------------------------------------
#BC=Au tot t
BC = (BA+BB)*1000000
#-----------------------------------------------------------------------
#BE=Gold Endowments
BE = BB*BD
hist(BE, breaks = 75)
#-----------------------------------------------------------------------
### Graphics:
```{r high, echo = T}
gold <- data.frame(cbind(BE, BI, A, D))
gold %>%
  ggplot(aes(x = BE, y = BI, colour = A)) +
  geom_point(aes(colour = A)) + xlab("Gold
Endowment (tons)") +
  ylab("Volume hydrous melt (Km^3)") +
   scale_colour_gradientn(colours =
terrain.colors(n)) +
  geom_vline(xintercept = 0, color = "red") +
  geom_vline(xintercept = 1000, color = "red")

gold %>%
  ggplot(aes(x=BE)) +
  geom_histogram() +
  xlab("Gold endowment (tons)") +
  geom_vline(xintercept = 0, color = "red") +
  ylab("Frequency") + geom_vline(xintercept =
1000, color = "red")

gold %>%
  ggplot(aes(x = BE, y = BI, colour = D)) +
  geom_point(aes(colour = D)) + xlab("Gold
Endowment (tons)") +
  ylab("Volume hydrous melt (Km^3)") +
   scale_colour_gradientn(colours =
terrain.colors(n)) +
  geom_vline(xintercept = 0, color = "red") +
  geom_vline(xintercept = 1000, color = "red")
#-----------------------------------------------------------------------
```
```

Bibliography

1-Annen, C. (2009). From plutons to magma chambers: Thermal constraints on the accumulation of eruptible silicic magma in the upper crust. Earth and Planetary Science Letters, 284(3–4), 409–416. https://doi.org/10.1016/j.epsl.2009.05.006
2-Annen, C., Blundy, J. D., & Sparks, R. S. J. (2006). The genesis of intermediate and silicic magmas in deep crustal hot zones. Journal of Petrology, 47(3), 505–539. https://doi.org/10.1093/petrology/egi084
3-Chiaradia, M., & Caricchi, L. (2017). Stochastic modelling of deep magmatic controls on porphyry copper deposit endowment. Scientific Reports, 7(January 2016), 1–11. https://doi.org/10.1038/srep44523
4-Chiaradia, M. (2020). Gold endowments of porphyry deposits controlled by precipitation efficiency. Nature Communications, (2020) 11:248, 1-10. https://doi.org/10.1038/s41467-019-14113-1
5-Moss, R., Scott, S. D., & Binns, A. R. A. (2001). Gold content of eastern Manus basin volcanic rocks: Implications for enrichment in associated hydrothermal precipitates. Economic Geology, 96(1), 91–107. https://doi.org/10.2113/gsecongeo.96.1.91
6-Plank, T., Kelley, K. A., Zimmer, M. M., Hauri, E. H., & Wallace, P. J. (2013). Why do mafic arc magmas contain 4wt% water on average? Earth and Planetary Science Letters, 364, 168–179. https://doi.org/10.1016/j.epsl.2012.11.044
7-Sanchez-Alfaro, P., Reich, M., Driesner, T., Cembrano, J., Arancibia, G., Pérez-Flores, P., Heinrich, C. A., Rowland, J., Tardani, D., Lange, D., & Campos, E. (2016). The optimal windows for seismically-enhanced gold precipitation in the epithermal environment. Ore Geology Reviews, 79, 463–473. https://doi.org/10.1016/j.oregeorev.2016.06.005
8-Simon, A. C., Frank, M. R., Pettke, T., Candela, P. A., Piccoli, P. M., & Heinrich, C. A. (2005). Gold partitioning in melt-vapor-brine systems. Geochimica et Cosmochimica Acta, 69(13), 3321–3335. https://doi.org/10.1016/j.gca.2005.01.028