```
In[1]:= (* Fits to log stable below. Starting
         with 90d worth of data from WETH/USDC .... *)

In[2]:= (* Import from csv *)

In[3]:= Directory[]

Out[3]= /Users/personal

In[4]:= Module[{directory = SystemDialogInput["Directory"]},
         If[directory =!= $Canceled, SetDirectory[directory]]]

Out[4]= /Users/personal/Desktop/note7/points

In[5]:= (* Go back and download WETH/USDC data from cron. Look at that *)

In[6]:= tblWethUsdc90d = Import["90/data-1625069716_weth-usdc-twap.csv"]
```

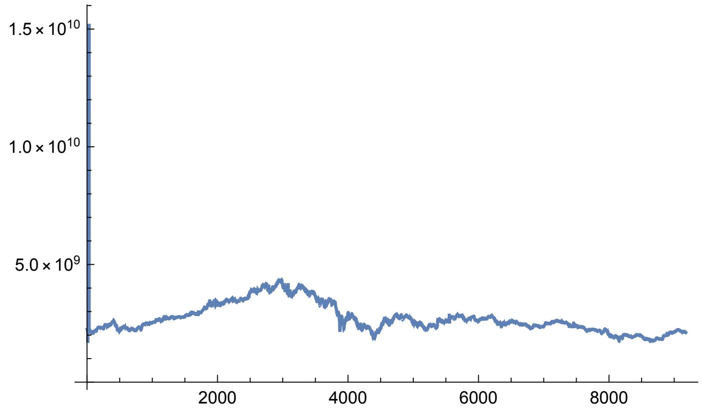
Out[6]= $\{\{, \text{timestamp}, \text{twap}\}, \{0, 1.61878\times10^{9}, 2.23957\times10^{9}\}, \{1, 1.61878\times10^{9}, 2.24054\times10^{9}\}, \{2, 1.61878\times10^{9}, 2.23817\times10^{9}\}, \{3, 1.61878\times10^{9}, 2.25083\times10^{9}\}, \{4, 1.61878\times10^{9}, 2.25656\times10^{9}\}, \{5, 1.61878\times10^{9}, 2.25775\times10^{9}\}, \{6, 1.61878\times10^{9}, 2.25804\times10^{9}\}, \cdots 9164 \cdots, \{9173, 1.62507\times10^{9}, 2.13479\times10^{9}\}, \{9174, 1.62507\times10^{9}, 2.12846\times10^{9}\}, \{9175, 1.62507\times10^{9}, 2.1222\times10^{9}\}, \{9176, 1.62507\times10^{9}, 2.11587\times10^{9}\}, \{9177, 1.62507\times10^{9}, 2.11137\times10^{9}\}, \{9178, 1.62507\times10^{9}, 2.10661\times10^{9}\}, \{9179, 1.62507\times10^{9}, 2.10415\times10^{9}\}\}$

large output | show less | show more | show all | set size limit...

```
In[7]:= Length[tblWethUsdc90d]

Out[7]= 9179

In[8]:= FromUnixTime[tblWethUsdc90d[[2]][[2]]]

Out[8]= Sun 18 Apr 2021 16:50:52 GMT-4.

In[9]:= FromUnixTime[tblWethUsdc90d[[Length[tblWethUsdc90d]]][[2]]]

Out[9]= Wed 30 Jun 2021 12:09:58 GMT-4.

In[10]:= twapsWethUsdc90d = Table[tblWethUsdc90d[[i]][[3]], {i, 2, Length[tblWethUsdc90d]}]
```

In[11]:=
```
ListLinePlot[twapsWethUsdc90d, PlotRange → All]
```

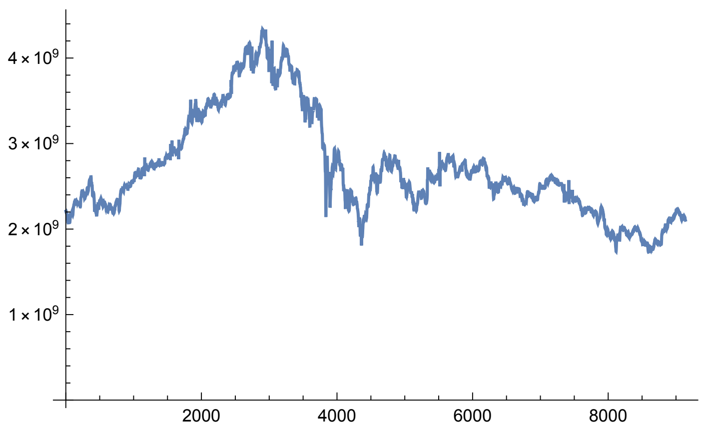
Out[11]=



In[12]:=
```
(* Same issues here for early data as UNI/WETH. Cut off first 40 elements *)
twapsWethUsdc90dFiltered =
 Table[twapsWethUsdc90d[[i]], {i, 40, Length[twapsWethUsdc90d]}]
```

In[13]:=
```
ListLinePlot[twapsWethUsdc90dFiltered, PlotRange → All]
```

Out[13]=



In[14]:=
```
FromUnixTime[tblWethUsdc90d[[40]][[2]]]
```

Out[14]= Mon 19 Apr 2021 17:14:44 GMT−4.

In[15]:=
```
FromUnixTime[tblWethUsdc90d[[Length[twapsWethUsdc90dFiltered]]][[2]]]
```

Out[15]= Wed 30 Jun 2021 04:33:51 GMT−4.

In[16]:=
```
(* Calculate the rs ... *)
```

In[17]:=
```
rsWethUsdc90dFiltered = Differences[Log[twapsWethUsdc90dFiltered]]
```

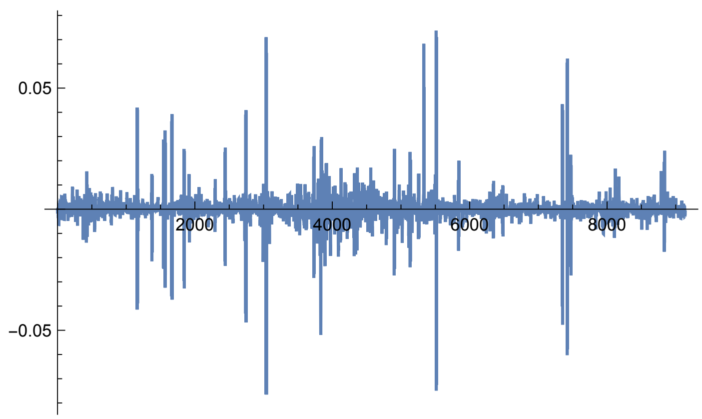
Out[17]=
```
{-0.00120673, 0.000176452, -0.0000603568, -0.000373722,
 -0.000389599, 0.000559858, -0.00092094, -0.00154871, -0.000905547,
 -0.000584311, -0.00172231, -0.00371162, ··· 9115 ···, 0.00150055,
 0.000591025, 0.000421125, -0.000241228, -0.000868266, -0.00297129,
 -0.00294238, -0.00298715, -0.00213101, -0.00225849, -0.00116666}
```

large output | show less | show more | show all | set size limit...

In[18]:=
```
rsWethUsdc90dFiltered[[100]]
```

Out[18]=
```
-0.00443888
```

In[19]:=
```
ListLinePlot[rsWethUsdc90dFiltered, PlotRange → All]
```

Out[19]=


In[20]:=
```
Length[rsWethUsdc90dFiltered]
```

Out[20]=
```
9138
```

In[21]:=
```
edistWethUsdc90dFiltered = EstimatedDistribution[rsWethUsdc90dFiltered,
  StableDistribution[1, aWU90d, bWU90d, locWU90d, scaleWU90d]]
```

Out[21]=
```
StableDistribution[1, 1.46465, -0.0496207, -0.000010553, 0.00154277]
```

In[22]:=
```
(* This seems more reasonable. more data from weth/usdc lead
 to decrease in alpha because included massive run up to $4k *)
```

In[23]:=
```
FromUnixTime[tblWethUsdc90d[[5000]][[2]]]
```

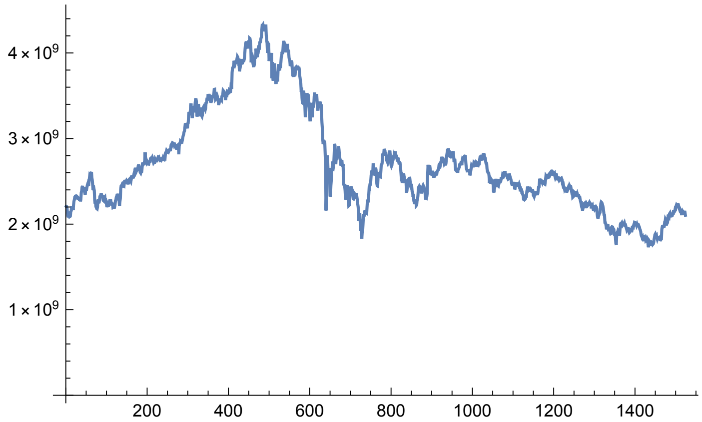
Out[23]= Fri 28 May 2021 06:16:07 GMT−4.

In[24]:=
```
FromUnixTime[tblWethUsdc90d[[Length[tblWethUsdc90d]]][[2]]]
```

Out[24]= Wed 30 Jun 2021 12:09:58 GMT−4.

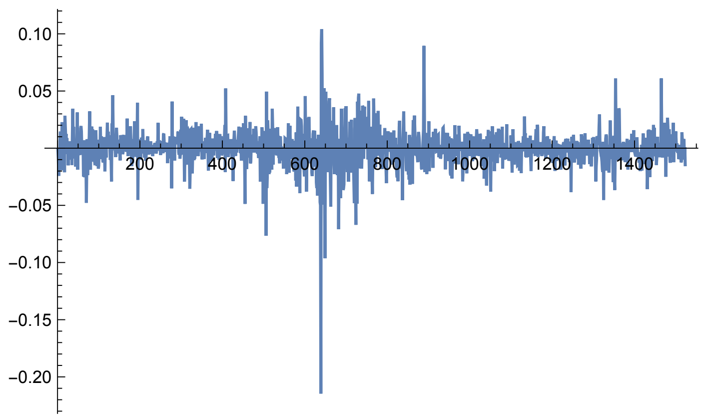
```
In[25]:= (* Seems a good place to estimate k
          values would be 1h candles (from note-7.nb) *)

In[26]:=

In[27]:= (* Some concrete numbers below in terms of funding rate ... *)

In[28]:= (* What does e^{mu * T + sig * (T/a)^(1/a) * F^{-1}(1-alpha)}
          translate to for ETH/USDC fit? *)

In[29]:= (* And if d = f * e^{mu * T + sig * (T/a)^(1/a) * F^{-1}(1-alpha)},
         what f value should we use across the board
           to get k max interest rates on order of 1-10% daily? *)

In[30]:= (* e.x. d = 1.0014649 for T=10m results in 10% per day funding rate *)

In[31]:= edistWethUsdc90dFiltered

Out[31]= StableDistribution[1, 1.46465, -0.0496207, -0.000010553, 0.00154277]

In[32]:= InverseCDF[edistWethUsdc90dFiltered, 0.99]

Out[32]= 0.0124991

In[33]:= InverseCDF[edistWethUsdc90dFiltered, 0.95]

Out[33]= 0.00479452

In[34]:= InverseCDF[edistWethUsdc90dFiltered, 0.90]

Out[34]= 0.0032239

In[35]:= Exp[InverseCDF[edistWethUsdc90dFiltered, 0.95]]

Out[35]= 1.00481

In[36]:= Exp[InverseCDF[edistWethUsdc90dFiltered, 0.99]]

Out[36]= 1.01258

In[37]:= (* Produce 1 hour candles from 10min cron data to use in k analysis ... *)

In[38]:= twapsWethUsdc90dFiltered1HourCandle =
          Table[twapsWethUsdc90dFiltered[[i]], {i, 1, Length[twapsWethUsdc90dFiltered], 6}]
```

In[39]:=
```
ListLinePlot[twapsWethUsdc90dFiltered1HourCandle]
```

Out[39]=


In[40]:=
```
rsWethUsdc90dFiltered1HourCandle =
  Differences[Log[twapsWethUsdc90dFiltered1HourCandle]]
```

In[41]:=
```
ListLinePlot[rsWethUsdc90dFiltered1HourCandle, PlotRange → All]
```

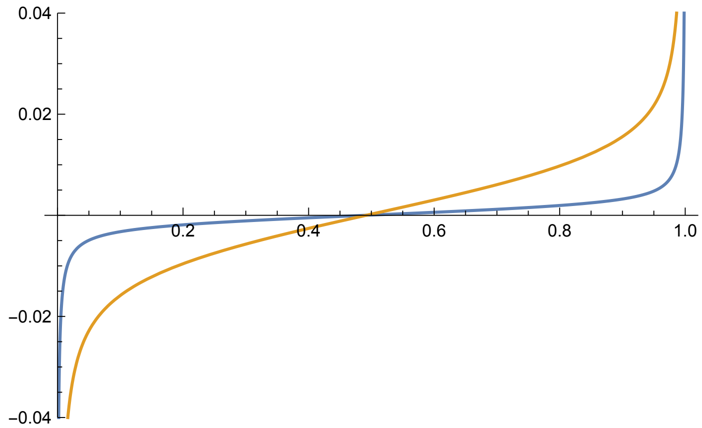
Out[41]=


In[42]:=
```
edistWethUsdc90dFiltered1HourCandle =
  EstimatedDistribution[rsWethUsdc90dFiltered1HourCandle,
   StableDistribution[1, aWU90dC1h, bWU90dC1h, locWU90dC1h, scaleWU90dC1h]]
```

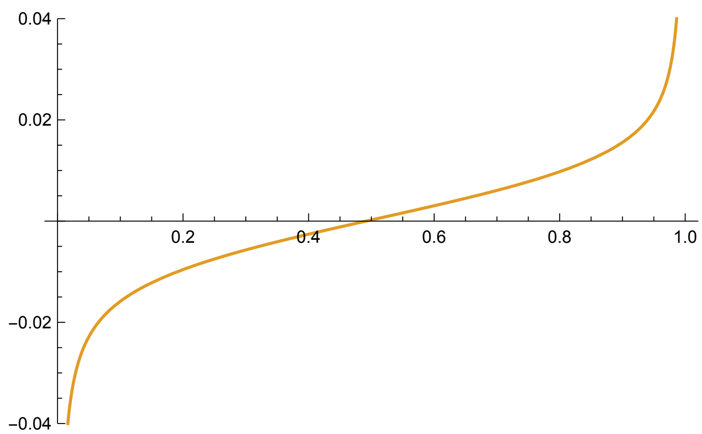
Out[42]=
```
StableDistribution[1, 1.59768, -0.0971292, -0.000157254, 0.00790013]
```

In[43]:=
```
(* Interesting ... compound less on the funding payments,
have more leeway. How does that make sense wrt inverse cdf exponentiation? *)
```

In[44]:=
```
Plot[{InverseCDF[edistWethUsdc90dFiltered, x],
  InverseCDF[edistWethUsdc90dFiltered1HourCandle, x],
  InverseCDF[edistWethUsdc90dFiltered4HourCandle, x]},
 {x, 0, 1.0}, PlotRange → {-0.04, 0.04}]
```

Out[44]=


In[45]:=
```
Plot[{, InverseCDF[edistWethUsdc90dFiltered1HourCandle, x]},
 {x, 0, 1.0},  PlotRange → {-0.04, 0.04}]
```

Out[45]=


In[46]:=
```
InverseCDF[edistWethUsdc90dFiltered1HourCandle, 0.99]
```

Out[46]= 0.0472398

In[47]:=
```
InverseCDF[edistWethUsdc90dFiltered, 0.99]
```

Out[47]= 0.0124991

In[48]:=
```
Exp[0.012499142980299232`]
```

Out[48]= 1.01258

In[49]:=
```
Exp[0.1072270317901139`]
```

Out[49]= 1.11319

In[50]:=
```
(* Hmmm, so maybe plot e^(mu * T + sig * (T/a)^(1/a) * F^{-1}(1-alpha)) as a
 function of update times T. What does this tell us vs d we're looking for? *)
```

```
In[51]:= edistWethUsdc90dFiltered1HourCandle

Out[51]= StableDistribution[1, 1.59768, -0.0971292, -0.000157254, 0.00790013]

In[52]:= (* Define mu, sig functions for exponential term in d *)

In[53]:= mu[a_, loc_] := loc

In[54]:= sig[a_, scale_] := scale / (1 / a) ^ (1 / a)

In[55]:= (* Apply to our case *)

In[56]:= muWethUsdc90dFiltered1HourCandle =
          mu[1.597679462042982`, -0.00015725436055898914`]

Out[56]= -0.000157254

In[57]:= sigWethUsdc90dFiltered1HourCandle = sig[1.597679462042982`, 0.007900125269736371`]

Out[57]= 0.0105925

In[58]:= aWethUsdc90dFiltered1HourCandle = 1.597679462042982`

Out[58]= 1.59768

In[59]:= edistWethUsdc90dFiltered1HourCandleNormalized =
          StableDistribution[1, aWethUsdc90dFiltered1HourCandle, 0, 0, 1]

Out[59]= StableDistribution[1, 1.59768, 0, 0, 1]

In[60]:= InverseCDF[edistWethUsdc90dFiltered1HourCandleNormalized, 0.90]

Out[60]= 1.98681

In[61]:= InverseCDF[edistWethUsdc90dFiltered1HourCandleNormalized, 0.95]

Out[61]= 2.81905

In[62]:= InverseCDF[edistWethUsdc90dFiltered1HourCandleNormalized, 0.99]

Out[62]= 6.31376

In[63]:= factorWethUsdc90dFiltered1HourCandle[t_, alpha_] :=
          Exp[muWethUsdc90dFiltered1HourCandle * t + sigWethUsdc90dFiltered1HourCandle *
              (t / aWethUsdc90dFiltered1HourCandle) ^ (1 / aWethUsdc90dFiltered1HourCandle) *
              InverseCDF[edistWethUsdc90dFiltered1HourCandleNormalized, alpha]]

In[64]:= factorWethUsdc90dFiltered1HourCandle[1, 0.99]

Out[64]= 1.05098

In[65]:= (* TODO: Redo this analysis! ... Below is in line with 4 hour candle above :) *)

In[66]:= factorWethUsdc90dFiltered1HourCandle[4, 0.99]

Out[66]= 1.12542
```

In[67]:=
```
factorWethUsdc90dFiltered1HourCandle[8, 0.99]
```
Out[67]= 1.19967

In[68]:=
```
(* Over 24 hours you can see the difference
  substantially at different confidence levels ... *)
```

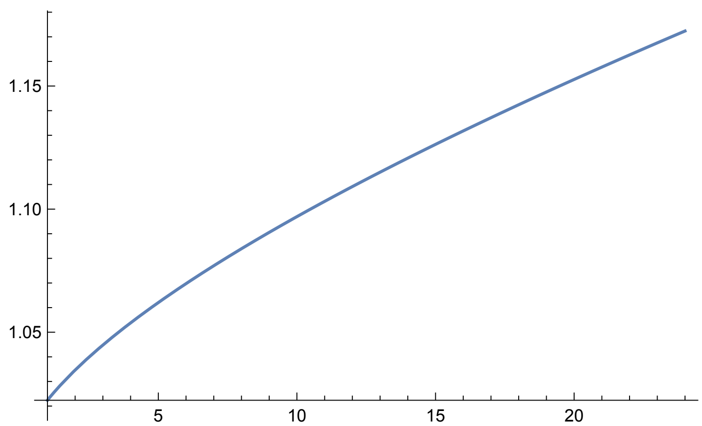
In[69]:=
```
factorWethUsdc90dFiltered1HourCandle[24, 0.99]
```
Out[69]= 1.4345

In[70]:=
```
factorWethUsdc90dFiltered1HourCandle[24, 0.90]
```
Out[70]= 1.11735

In[71]:=
```
factorWethUsdc90dFiltered1HourCandle[24, 0.95]
```
Out[71]= 1.17235

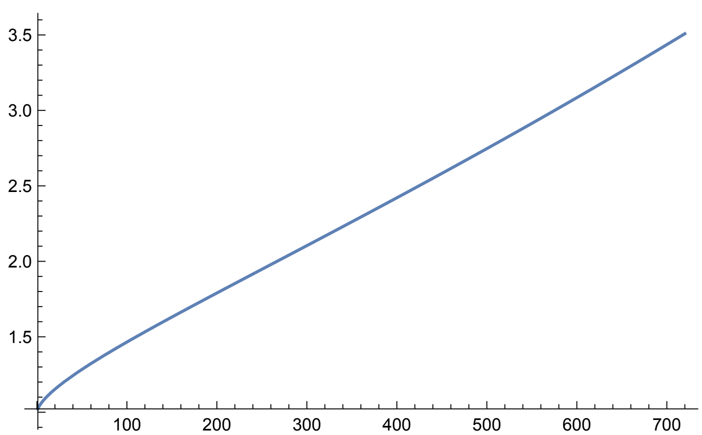
In[72]:=
```
(* VaR at 95% seems good here in terms of usability for trading. 15% draw down in
 a day worst case on ETH/USDC is not terrible in terms of funding rate max *)
```

In[73]:=
```
Plot[factorWethUsdc90dFiltered1HourCandle[t, 0.95], {t, 1, 24}]
```
Out[73]=

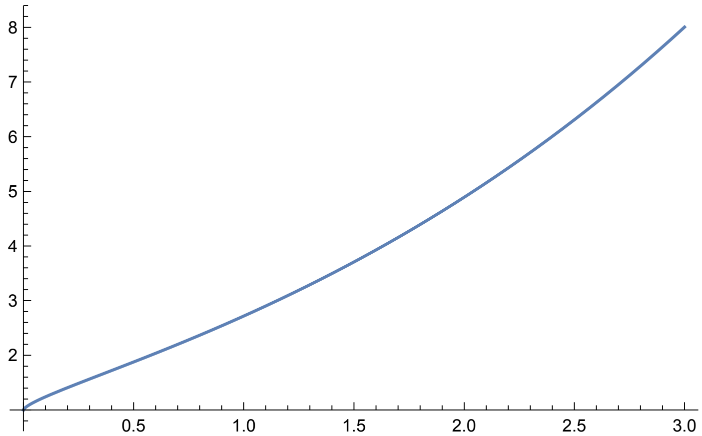


In[74]:=
```
(* Interesting S shape here ... Due to the Exp[t^(1/1.5)] term *)
```

In[75]:=
```
Plot[factorWethUsdc90dFiltered1HourCandle[t, 0.95], {t, 1, 24*30}]
```
Out[75]=

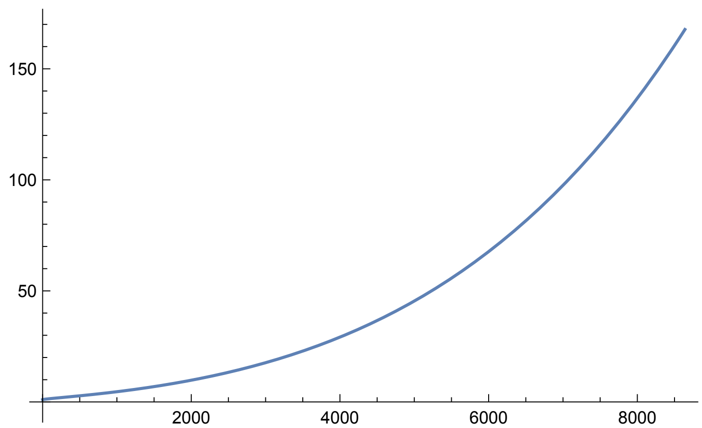


In[76]:= `Plot[Exp[(t)^(1/1.5)], {t, 0, 3}]`

Out[76]=


In[77]:= `(* This is where caps become very good :) *)`

In[78]:= `Plot[factorWethUsdc90dFiltered1HourCandle[t, 0.95], {t, 1, 24*30*12}]`

Out[78]=


In[79]:=

In[80]:= `(* Plot k AND (1-k)^m where m is diff depending on number of times we compound *)`

In[81]:=
```
(* Assume k min is determined by factor value
 at 95% confidence for different number of periods T *)
```

In[82]:=
```
dmin95WethUsdc90dFiltered1HourCandle[t_] :=
 factorWethUsdc90dFiltered1HourCandle[t, 0.95]
```

In[83]:=
```
kmin95WethUsdc90dFiltered1HourCandle[t_] :=
 (1 - 1 / dmin95WethUsdc90dFiltered1HourCandle[t]) / 2
```

In[84]:=
```
(* Look at different time periods ... 1 (1h),
4 (4h), 6 (6h), 8 (8h), 12 (12h), 24 (1d), 168 (7d) *)
```

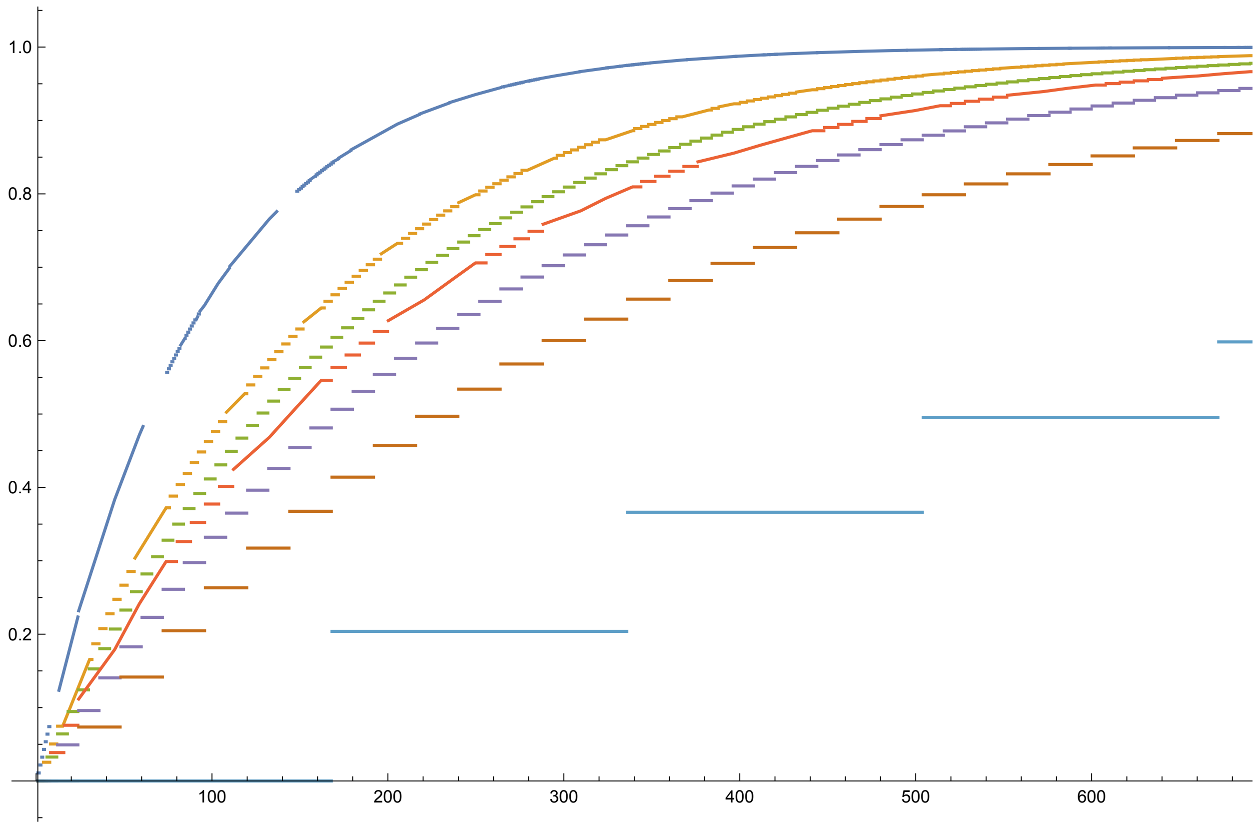
In[85]:= `kmin95T1hWethUsdc90dFiltered1HourCandle = kmin95WethUsdc90dFiltered1HourCandle[1]`

Out[85]= 0.0109354

```
In[86]:= kmin95T4hWethUsdc90dFiltered1HourCandle = kmin95WethUsdc90dFiltered1HourCandle[4]
Out[86]= 0.0255287

In[87]:= kmin95T6hWethUsdc90dFiltered1HourCandle = kmin95WethUsdc90dFiltered1HourCandle[6]
Out[87]= 0.0325961

In[88]:= kmin95T8hWethUsdc90dFiltered1HourCandle = kmin95WethUsdc90dFiltered1HourCandle[8]
Out[88]= 0.0387123

In[89]:= kmin95T12hWethUsdc90dFiltered1HourCandle =
          kmin95WethUsdc90dFiltered1HourCandle[12]
Out[89]= 0.0492077

In[90]:= kmin95T1dWethUsdc90dFiltered1HourCandle = kmin95WethUsdc90dFiltered1HourCandle[24]
Out[90]= 0.0735077

In[91]:= kmin95T7dWethUsdc90dFiltered1HourCandle =
          kmin95WethUsdc90dFiltered1HourCandle[168]
Out[91]= 0.203882

In[92]:= (* Interesting ... around minimum of 3.28% at every 8 hr
          compounding. Can work with that to have some buffer in event bad fit *)

In[93]:= drawDownMin95WethUsdc90dFiltered1HourCandle[t_, m_] :=
          1 - (1 - kmin95WethUsdc90dFiltered1HourCandle[m]) ^ (Floor[t / m])
```

In[94]:=
```
Plot[{drawDownMin95WethUsdc90dFiltered1HourCandle[t, 1],
  drawDownMin95WethUsdc90dFiltered1HourCandle[t, 4],
  drawDownMin95WethUsdc90dFiltered1HourCandle[t, 6],
  drawDownMin95WethUsdc90dFiltered1HourCandle[t, 8],
  drawDownMin95WethUsdc90dFiltered1HourCandle[t, 12],
  drawDownMin95WethUsdc90dFiltered1HourCandle[t, 24],
  drawDownMin95WethUsdc90dFiltered1HourCandle[t, 168]}, {t, 1, 24 * 30}]
```

Out[94]=

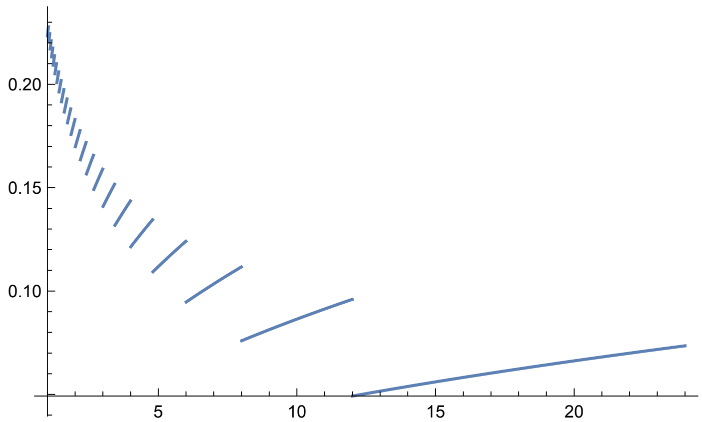


In[95]:=
```
(* Should like compare ratio of price bracket to d^(t/m). When set m for d,
setting point at which d^m =
 price bracket[m].. But d is NOT an interest rate... k = (1-1/d)/2 makes this
   interesting in terms of traders view on draw down to their OI ... *)
```

In[96]:=
```
(* Plot the actual interest rates seen per day for different values of m *)
```

In[97]:=
```
Plot[drawDownMin95WethUsdc90dFiltered1HourCandle[24, m], {m, 1, 24}]
```

Out[97]=



In[111]:=
```
drawDownMin95WethUsdc90dFiltered1HourCandle[24, 1]
```

Out[111]= 0.231947

In[112]:=
```
drawDownMin95WethUsdc90dFiltered1HourCandle[24, 4]
```

Out[112]= 0.143723

In[113]:=
```
drawDownMin95WethUsdc90dFiltered1HourCandle[24, 8]
```

Out[113]= 0.111699

In[114]:=
```
drawDownMin95WethUsdc90dFiltered1HourCandle[24, 12]
```

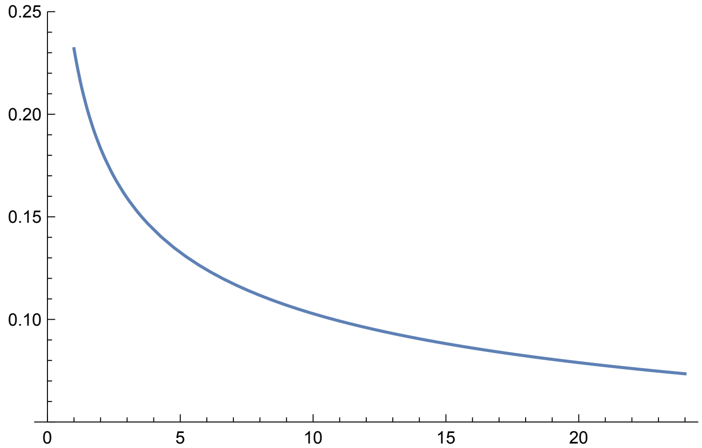
Out[114]= 0.095994

In[98]:=
```
(* Floor is making this visual
  difficult. Make continuous for plot purposes ... *)
```

In[99]:=
```
drawDownMin95WethUsdc90dFiltered1HourCandleContinuous[t_, m_] :=
 1 - (1 - kmin95WethUsdc90dFiltered1HourCandle[m]) ^ (t / m)

(* Below is min funding rate required extended to per day rate,
if apply funding payment every m hours  *)
```

In[131]:=
```
Plot[drawDownMin95WethUsdc90dFiltered1HourCandleContinuous[24, m],
 {m, 1, 24}, PlotRange → {0.05, 0.25}]
```

Out[131]=



In[101]:=
```
(* Less we compound, smaller max DAILY funding rate needed to overcome
 price bracket changes. But more short term (<1d) we likely take *)
```

In[102]:=
```
(* Generalize this for any confidence level (not just 95%) *)
```

In[103]:=
```
dminWethUsdc90dFiltered1HourCandle[t_, alpha_] :=
 factorWethUsdc90dFiltered1HourCandle[t, alpha]
```

In[104]:=
```
kminWethUsdc90dFiltered1HourCandle[t_, alpha_] :=
 (1 - 1 / dminWethUsdc90dFiltered1HourCandle[t, alpha]) / 2
```

In[105]:=
```
drawDownMinWethUsdc90dFiltered1HourCandle[t_, alpha_, m_] :=
 1 - (1 - kminWethUsdc90dFiltered1HourCandle[m, alpha]) ^ (Floor[t / m])
```

In[119]:=
```
drawDownMinWethUsdc90dFiltered1HourCandleContinuous[t_, alpha_, m_] :=
 1 - (1 - kminWethUsdc90dFiltered1HourCandle[m, alpha]) ^ (t / m)
```

In[107]:=
```
(* Plot price bracket * d^(t/m) over time t *)
```

```
(* Look at funding rates needed over different confidence levels, to compare *)
```

In[120]:=
```
drawDownMinWethUsdc90dFiltered1HourCandle[24, 0.99, 8]
```

Out[120]= 0.229456

In[121]:=
```
drawDownMinWethUsdc90dFiltered1HourCandle[24, 0.95, 8]
```

Out[121]= 0.111699

In[122]:=
```
drawDownMinWethUsdc90dFiltered1HourCandle[24, 0.90, 8]
```

Out[122]= 0.0800559

```
(* Last 95 → 99% is what ramps it up significantly *)
```

In[124]:=
```
drawDownMinWethUsdc90dFiltered1HourCandle[24, 0.99, 1]
```

Out[124]= 0.445257

In[123]:=
```
drawDownMinWethUsdc90dFiltered1HourCandle[24, 0.95, 1]
```
Out[123]=
```
0.231947
```

In[125]:=
```
drawDownMinWethUsdc90dFiltered1HourCandle[24, 0.90, 1]
```
Out[125]=
```
0.169511
```

In[128]:=
```
drawDownMinWethUsdc90dFiltered1HourCandle[24, 0.95, 24]
```
Out[128]=
```
0.0735077
```