Steps In Order

I. CAMS as Real Hazardous Data

1. Import whole CAMS data

In[1]:=
```
CAMS = Import["/Users/muxiliu/Downloads/CAMS-v3-2010to2016 .xlsx"]
```

Out[1]=
```
{{{CAMS Meteoroid Orbit Database v3.0 (2010-2017), , , , , , , , , , , , , , , , ,
    , , , , , , , , , , , , , , , , , , , , , , , , , , , , , , , , , , , , , , , ,
    , , , , , , , , , , , , , , , , , }, ··· 471 585 ···, {471 582., ··· 80 ···, -16.15}}}
```

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

In[2]:=
```
Length[CAMS[[1]]]
```

Out[2]=
```
471 587
```

2. Selecting relevant columns, only keeping 6 orbital elements and their uncertainties

In[3]:=
```
CAMS0 = Table[{CAMS[[1]][[i]][[60]], CAMS[[1]][[i]][[61]], CAMS[[1]][[i]][[64]],
    CAMS[[1]][[i]][[65]], CAMS[[1]][[i]][[66]], CAMS[[1]][[i]][[67]],
    CAMS[[1]][[i]][[68]], CAMS[[1]][[i]][[69]], CAMS[[1]][[i]][[70]],
    CAMS[[1]][[i]][[71]], CAMS[[1]][[i]][[72]]}, {i, 4, 471587}]
```

Out[3]=
```
{{q, ±, a, e, ±, i, ±, w, ±, Node, ±},
 {(AU), (AU), (AU), , , (°, J2000), , (°), , (°), (°)},
 ··· 471580 ···, {0.5499, 0.0101, 11.59, 0.9525, 0.0195, 137.66,
  0.64, 84.46, 1.58, 100.423, 0.001}, {0.6611, 0.01194, 11.65,
  0.9433, 0.0297, 147.49, 0.89, 71.04, 1.92, 100.429, 0.002}}
```

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

In[•]:=
```
Length[CAMS0]
```

Out[•]=
```
471584
```

3. Change the title

In[4]:=
```
CAMS1 = PrependTo[Table[CAMS0[[i]], {i, 3, 471584}], {"q(AU)", "qerror", "a(AU)", "e",
    "eerror", "i(°,J2000)", "ierror", "w(°)", "werror", "Node(°)", "Nodeerror"}]
```

··· Set: Tag Table in Table[CAMS0[[i]], {i, 3, 471584}] is Protected.

Out[•]=
```
{{q(AU), qerror, a(AU), e, eerror,
  i(°,J2000), ierror, w(°), werror, Node(°), Nodeerror},
 {0.9713, 0.0007, 2.85, 0.65907, 0.007, 22.16, 0.11, 159.82, 0.36, 207.481, 0.},
 {0.2584, 0.0023, 1.85, 0.8601, 0.002, 3.14, 0.24, 306.95, 0.4, 207.497, 0.003},
 {0.7837, 0.0147, 1.69, 0.536, 0.066, 11.23, 2.65, 247.03, 4.44, 207.543, 0.007},
 ··· 471576 ···, {0.21646, 0.00489, 5.79, 0.9626, 0.0041, 152.12,
  0.94, 306.17, 0.65, 280.425, 0.001}, {0.5499, 0.0101, 11.59, 0.9525,
  0.0195, 137.66, 0.64, 84.46, 1.58, 100.423, 0.001}, {0.6611, 0.01194,
  11.65, 0.9433, 0.0297, 147.49, 0.89, 71.04, 1.92, 100.429, 0.002}}
```

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

In[91]:=
```
Mean[Table[CAMS1[[i]][[3]], {i, 2, Length[CAMS1]}]]
```

Out[91]=
```
127.502
```

In[64]:=
```
Length[CAMS1]
```
Out[64]= 471583

10. Stuff for adjusting the amount and feature values of CAMS data to meet those of NEOs. Amount: similar "0"/all ratio with that in NEOs. Feature values: not too divergent from those of NEOs (through mean, max, and standard deviation).

In[•]:=
```
NEOsForTesting1 = {};
i = 1;
While[i < Length[NEOsForTesting] + 1, If[NEOsForTesting[[i]][[12]] == 1,
  NEOsForTesting1 = Append[NEOsForTesting1, NEOsForTesting[[i]]]];
 i++];
Length[NEOsForTesting1]
```
Out[•]= 405

In[•]:=
```
StandardDeviation[Table[NEOs1[[i]][[6]], {i, 2, Length[NEOs1]}]]
```
Out[•]= 12.8141

In[•]:=
```
NEOsStat = {};
i = 1;
While[i < Length[NEOs1] + 1,
 If[NEOs1[[i]][[6]] > 80, NEOsStat = Append[NEOsStat, NEOs1[[i]]]];
 i++];
Length[NEOsStat]
```
Out[•]= 5

In[354]:=
```
N[(Length[NEOsForTesting] - Length[NEOsForTesting1])/Length[NEOsForTesting]]
```
Out[354]= 0.780011

In[71]:=
```
NumberOfNeeded1 =
 Round[Length[NEOsForTesting1]/(Length[NEOsForTesting] - Length[NEOsForTesting1]) * Length[NEOsForTraining1]]
```
Out[71]= 1609

In[330]:=
```
CAMSForTraining0 = RandomSample[Delete[CAMS1, 1], Round[2.2 * NumberOfNeeded1]]
```

Out[330]=
```
{{0.45148, 0.00737, 4.88, 0.9074, 0.0044, 54.4, 0.3, 277.99, 0.98, 253.27, 0.},
 {0.24487, 0.00146, 2.5, 0.9019, 0.0015, 2.43, 0.22, 306.37, 0.21, 24.221, 0.008},
 ··· 3536 ···, {0.70496, 0.03965, 9.06, 0.9222,
  0.1682, 120.11, 2.9, 111.35, 8.48, 132.56, 0.002},
 {0.67591, 0.02243, 1.91, 0.6464, 0.0205, 136.89, 1.57, 279.45, 3.1, 350.57, 0.}}
```

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

In[331]:=
```
CAMSForTrainingNEO = {};
i = 1;
While[i < Length[CAMSForTraining0] + 1, If[CAMSForTraining0[[i]][[3]] < 10,
  CAMSForTrainingNEO = Append[CAMSForTrainingNEO, CAMSForTraining0[[i]]]];
 i++];
Length[CAMSForTrainingNEO]
```

Out[333]=
```
2601
```

In[342]:=
```
StandardDeviation[
 Table[CAMSForTrainingNEO1[[i]][[6]], {i, 1, Length[CAMSForTrainingNEO1]}]]
```

Out[342]=
```
20.0144
```

In[344]:=
```
CAMSForTrainingNEO1
```

Out[344]=
```
{{0.45148, 0.00737, 4.88, 0.9074, 0.0044, 54.4, 0.3, 277.99, 0.98, 253.27, 0.},
 {0.24487, 0.00146, 2.5, 0.9019, 0.0015, 2.43,
  0.22, 306.37, 0.21, 24.221, 0.008}, ··· 1682 ···,
 {0.08904, 0.0027, 2.11, 0.9578, 0.0019, 25.04, 0.66, 149.82, 0.5, 320.95, 0.},
 {0.14865, 0.00097, 1.3, 0.8854, 0.0009, 22.63, 0.19, 323.9, 0.14, 261.318, 0.}}
```

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

In[334]:=
```
CAMSForTrainingNEO1 = {};
i = 1;
While[i < Length[CAMSForTrainingNEO] + 1, If[CAMSForTrainingNEO[[i]][[6]] < 80,
  CAMSForTrainingNEO1 = Append[CAMSForTrainingNEO1, CAMSForTrainingNEO[[i]]]];
 i++];
Length[CAMSForTrainingNEO1]
```

Out[336]=
```
1686
```

11. This is the CAMs data ready to go.

In[345]:=
```
CAMST = Transpose[
  Append[Transpose[CAMSForTrainingNEO1], Table[1, Length[CAMSForTrainingNEO1]]]]
```

Out[345]=
```
{{0.45148, 0.00737, 4.88, 0.9074, 0.0044,
   54.4, 0.3, 277.99, 0.98, 253.27, 0., 1}, ··· 1684 ···,
 {0.14865, 0.00097, 1.3, 0.8854, 0.0009, 22.63, 0.19, 323.9, 0.14, 261.318, 0., 1}}
```

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

II. NEOs for training mixing and for prediction.

4. Import NEOs with PHA-defined and H<=22

In[6]:=
```
NEOs = Import["/Users/muxiliu/Downloads/results (12).csv"]
```

Out[6]=
```
{{id, pha, neo, e, a, q, i, om, w, sigma_e, sigma_q, sigma_i, sigma_om, sigma_w}, ··· 920
```
large output | show less | show more | show all | set size limit...

In[21]:=
```
NEOs[[1]]
```

Out[21]=
```
{id, pha, neo, e, a, q, i, om, w, sigma_e, sigma_q, sigma_i, sigma_om, sigma_w}
```

5. reordering the columns to match the CAMS data.

In[22]:=
```
NEOs0 = Table[
  {NEOs[[i]][[6]], NEOs[[i]][[11]], NEOs[[i]][[5]], NEOs[[i]][[4]], NEOs[[i]][[10]],
   NEOs[[i]][[7]], NEOs[[i]][[12]], NEOs[[i]][[9]], NEOs[[i]][[14]],
   NEOs[[i]][[8]], NEOs[[i]][[13]], NEOs[[i]][[2]]}, {i, 1, Length[NEOs]}]
```

Out[22]=
```
{{q, sigma_q, a, e, sigma_e, i, sigma_i, w, sigma_w, om, sigma_om, pha},
 ··· 9205 ···, {0.950941, 1.6871 × 10^-7, 2.81715, 0.662446, 6.6368 × 10^-8,
  4.67928, 0.000018533, 234.895, 0.000089692, 182.983, 0.000056917, Y}}
```
large output | show less | show more | show all | set size limit...

6. Change Y and N in the ‘pha’ column to 1 and 0.

In[24]:=
```
NEOs05 = NEOs0 /. "Y" → 1
```

Out[•]=
```
{{q, sigma_q, a, e, sigma_e, i, sigma_i, w, sigma_w, om, sigma_om, pha},
 {1.13297, 2.6168 × 10^-10, 1.45805, 0.222951, 1.2856 × 10^-10, 10.8305,
  4.0878 × 10^-8, 178.882, 2.4392 × 10^-7, 304.299, 2.1639 × 10^-7, N},
 ··· 9203 ···, {0.544439, 0.0038723, 1.72447, 0.684286, 0.0044412,
  12.774, 0.059463, 290.259, 0.7191, 316.946, 0.2133, N},
 {0.950941, 1.6871 × 10^-7, 2.81715, 0.662446, 6.6368 × 10^-8, 4.67928,
  0.000018533, 234.895, 0.000089692, 182.983, 0.000056917, 1}}
```

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

In[25]:=
```
NEOs09 = NEOs05 /. "N" → 0
```

Out[•]=
```
{{q, sigma_q, a, e, sigma_e, i, sigma_i, w, sigma_w, om, sigma_om, pha},
 {1.13297, 2.6168 × 10^-10, 1.45805, 0.222951, 1.2856 × 10^-10, 10.8305,
  4.0878 × 10^-8, 178.882, 2.4392 × 10^-7, 304.299, 2.1639 × 10^-7, 0},
 ··· 9203 ···, {0.544439, 0.0038723, 1.72447, 0.684286, 0.0044412,
  12.774, 0.059463, 290.259, 0.7191, 316.946, 0.2133, 0},
 {0.950941, 1.6871 × 10^-7, 2.81715, 0.662446, 6.6368 × 10^-8, 4.67928,
  0.000018533, 234.895, 0.000089692, 182.983, 0.000056917, 1}}
```

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

7. Shuffle

In[52]:=
```
NEOs1 = RandomSample[Delete[NEOs09, 1]]
```

Out[•]=
```
{{0.917364, 0.0029289, 1.55211, 0.408956, 0.00031644,
  51.9561, 0.31745, 14.0655, 0.16341, 36.9336, 0.0063784, 0},
 {1.28547, 1.4921 × 10^-7, 2.63296, 0.511775, 5.6799 × 10^-8, 24.4952,
  0.000016742, 259.886, 0.000031075, 234.322, 0.000016709, 0},
 ··· 9203 ···, {1.08734, 2.19 × 10^-7, 2.43474, 0.553408, 8.4299 × 10^-8,
  8.8995, 0.000012923, 355.094, 0.00007692, 308.935, 0.000080209, 0}}
```

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

In[54]:=
```
Length[NEOs1]
```

Out[54]= 9206

8. splitting the data into 80/20.

In[55]:=
```
NEOsForTraining0 = Table[NEOs1[[i]], {i, 1, Round[0.8 * Length[NEOs1]]}]
```

Out[•]=
```
{{0.917364, 0.0029289, 1.55211, 0.408956, 0.00031644,
  51.9561, 0.31745, 14.0655, 0.16341, 36.9336, 0.0063784, 0},
 {1.28547, 1.4921 × 10^-7, 2.63296, 0.511775, 5.6799 × 10^-8, 24.4952,
  0.000016742, 259.886, 0.000031075, 234.322, 0.000016709, 0},
 {0.740625, 5.6513 × 10^-7, 1.05216, 0.296088, 6.2608 × 10^-7, 7.90666,
  0.000013751, 242.303, 0.000072207, 128.827, 0.000090308, 1},
 {1.0416, 0.086431, 1.88992, 0.448866, 0.066854, 5.85772,
  0.67616, 125.012, 13.813, 179.696, 1.054, 0}, ··· 7358 ···,
 {0.923016, 1.3566 × 10^-6, 2.73496, 0.662511, 0.00001386, 18.6341,
  0.00029696, 227.212, 0.000055958, 207.613, 0.00005802, 1},
 {0.481869, 7.6377 × 10^-7, 1.04021, 0.536756, 7.7467 × 10^-7, 17.462,
  0.000018609, 149.892, 0.000036816, 48.2625, 0.000018833, 0},
 {1.20602, 3.4805 × 10^-6, 1.48899, 0.190039, 9.8539 × 10^-6, 4.89264,
  0.000093718, 138.429, 0.00068154, 115.819, 0.00057095, 0}}
```

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

In[56]:=
```
Length[NEOsForTraining0]
```

Out[56]= 7365

In[65]:=
```
NEOsForTesting = Table[NEOs1[[i]], {i, Length[NEOsForTraining0] + 1, Length[NEOs1]}]
```

Out[•]=
```
{{0.785601, 5.3672 × 10^-7, 1.4678, 0.464776, 3.53 × 10^-7,
  2.34153, 0.000013635, 3.3017, 0.00058664, 271.825, 0.00058532, 1},
 {0.963866, 8.0274 × 10^-8, 2.07328, 0.535102, 3.8658 × 10^-8, 1.43065,
  6.4926 × 10^-6, 123.068, 0.00047759, 235.392, 0.00047768, 1},
 ··· 1838 ···, {1.08734, 2.19 × 10^-7, 2.43474, 0.553408, 8.4299 × 10^-8,
  8.8995, 0.000012923, 355.094, 0.00007692, 308.935, 0.000080209, 0}}
```

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

In[66]:=
```
Length[NEOsForTesting]
```

Out[66]= 1841

9. only keep those with pha=0 to form the really non-hazardous data set

In[62]:=
```
NEOsForTraining1 = {};
i = 1;
While[i < Length[NEOsForTraining0] + 1, If[NEOsForTraining0[[i]][[12]] == 0,
  NEOsForTraining1 = Append[NEOsForTraining1, NEOsForTraining0[[i]]]];
 i++];
Length[NEOsForTraining1]
```

Out[62]= 5706

In[63]:=
```
NEOsForTraining1
```

Out[63]=
```
{{0.917364, 0.0029289, 1.55211, 0.408956, 0.00031644,
  51.9561, 0.31745, 14.0655, 0.16341, 36.9336, 0.0063784, 0},
 ··· 5704 ···, {1.20602, 3.4805 × 10^-6, 1.48899, 0.190039, 9.8539 × 10^-6,
  ··· 2 ···, 138.429, 0.00068154, 115.819, 0.00057095, 0}}
```

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

III. Final Combination

12. Combine CAMS-ready-to-go with non-hazardous NEOs for training

In[346]:=
```
ToGo0 = Join[NEOsForTraining1, CAMST]
```

Out[346]=
```
{{0.917364, 0.0029289, 1.55211, 0.408956, 0.00031644, 51.9561,
  0.31745, 14.0655, 0.16341, 36.9336, 0.0063784, 0}, ··· 7390 ···,
 {0.14865, 0.00097, 1.3, 0.8854, 0.0009, 22.63, 0.19, 323.9, 0.14, 261.318, 0., 1}}
```

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

In[347]:=
```
Length[ToGo0]
```

Out[347]= 7392

13. Add the column names.

In[348]:=
```
ToGo = Prepend[RandomSample[ToGo0], {"q(AU)", "qerror", "a(AU)", "e", "eerror",
   "i(°,J2000)", "ierror", "w(°)", "werror", "Node(°)", "Nodeerror", "danger"}]
```

Out[348]=
```
{{q(AU), qerror, a(AU), e, eerror, i(°,J2000),
  ierror, w(°), werror, Node(°), Nodeerror, danger}, ··· 7391 ···,
 {1.27412, 2.8991 × 10^-6, 1.87435, 0.320232, 2.0461 × 10^-6, 38.8307,
  0.00011544, 161.867, 0.00020762, 253.256, 0.00013577, 0}}
```

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

14. Add the column names for the prediction data set.

In[351]:=
```
Testing = Prepend[RandomSample[NEOsForTesting],
  {"q(AU)", "qerror", "a(AU)", "e", "eerror", "i(°,J2000)",
   "ierror", "w(°)", "werror", "Node(°)", "Nodeerror", "danger"}]
```

Out[351]=
```
{{q(AU), qerror, a(AU), e, eerror, i(°,J2000),
  ierror, w(°), werror, Node(°), Nodeerror, danger}, ··· 1840 ···,
 {1.023, 2.2228 × 10^-7, 2.23032, 0.541322, 9.9818 × 10^-8, 39.6477,
  0.000014079, 4.61201, 0.000015518, 267.205, 0.000013838, 1}}
```

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

In[349]:=
```
Length[ToGo]
```

Out[349]= 7393

15. Finally, export the training and prediction data set.

In[350]:=
```
Export["TrainingToGo.csv", ToGo]
```

Out[350]=
```
TrainingToGo.csv
```

In[352]:=
```
Export["TestingToGo.csv", Testing]
```

Out[352]=
```
TestingToGo.csv
```