DNA1

November 26, 2024

```
[ ]:
```

```
[ ]:
```

```
[82]: import csv
      import pandas as pd
      snpAll = pd.read_csv('gs://ancient-dna-bucket/v62.snp', sep="\s+", header=None)
      snpAll[6] = range(1,1233014)
      display(snpAll)
      #snpA = snp[0].isin(['rs12124819'])
      #print(snpA)
      snpAll[snpAll[0]=='rs12124819']
      snpList = ('1','rs7419119', 'rs12124819')
      print(len(snpList))
      snpChoice = list(range(len(snpList)))
      print(snpChoice)
      #d = [range(1,len(snpList),1)]
      #print(d)
      #print(range(1,len(snpList)))
      #print(snpChoice[1])
      for i in range(0,len(snpList)):
          print(i)
          snpChoice[i] = snpAll[snpAll[0]==snpList[i]]
```

```
                        0   1         2         3  4  5        6
0               rs3094315   1  0.020130    752566  G  A        1
1              rs12124819   1  0.020242    776546  A  G        2
2              rs28765502   1  0.022137    832918  T  C        3
3               rs7419119   1  0.022518    842013  T  G        4
4                rs950122   1  0.022720    846864  G  C        5
…                       …  ..       …         …  .. ..       …
1233008        rs60847530  24  0.000000  59030572  T  A  1233009
1233009  snp_24_59033099  24  0.000000  59033099  T  C  1233010
1233010        rs28628009  24  0.000000  59033110  A  T  1233011
```

```
1233011       rs55686319  24  0.000000  59033139  T  C  1233012
1233012       rs75089321  24  0.000000  59033249  G  T  1233013

[1233013 rows x 7 columns]

3
[0, 1, 2]
0
1
2
```

[92]:

```python
import csv
import pandas as pd
snpAll = pd.read_csv('gs://ancient-dna-bucket/v62.snp', sep="\s+", header=None)
snpAll[6] = range(1,1233014)
display(snpAll)
#snpA = snp[0].isin(['rs12124819'])
#print(snpA)
snpAll[snpAll[0]=='rs12124819']
snpList = ('l','rs7419119', 'rs12124819')
print(len(snpList))
snpChoice = pd.DataFrame()
#snpChoice = []
print(snpChoice)
#d = [range(1,len(snpList),1)]
#print(d)
#print(range(1,len(snpList)))
#print(snpChoice[1])
print("for loop")
for i in range(0,len(snpList)):
    print(i)
#    nRow = snpAll[snpAll[0]==snpList[i]]
    nRow = snpAll.loc[snpAll[0] == snpList[i]]
#    snpChoice[len(snpChoice)] = nRow
    print(nRow)
```

```
                       0   1         2         3  4  5        6
0              rs3094315   1  0.020130    752566  G  A        1
1             rs12124819   1  0.020242    776546  A  G        2
2             rs28765502   1  0.022137    832918  T  C        3
3              rs7419119   1  0.022518    842013  T  G        4
4               rs950122   1  0.022720    846864  G  C        5
…                      …  ..         …         …  .. ..       …
1233008       rs60847530  24  0.000000  59030572  T  A  1233009
1233009  snp_24_59033099  24  0.000000  59033099  T  C  1233010
1233010       rs28628009  24  0.000000  59033110  A  T  1233011
1233011       rs55686319  24  0.000000  59033139  T  C  1233012
```

```
1233012       rs75089321  24  0.000000  59033249  G  T  1233013

[1233013 rows x 7 columns]
```

```
3
Empty DataFrame
Columns: []
Index: []
for loop
0
Empty DataFrame
Columns: [0, 1, 2, 3, 4, 5, 6]
Index: []
1
           0  1         2       3  4  5  6
3  rs7419119  1  0.022518  842013  T  G  4
2
            0  1         2       3  4  5  6
1  rs12124819  1  0.020242  776546  A  G  2
```

[97]:
```python
import csv
import pandas as pd
snpAll = pd.read_csv('gs://ancient-dna-bucket/v62.snp', sep="\s+", header=None)
snpAll[6] = range(1,1233014)
display(snpAll)
#snpA = snp[0].isin(['rs12124819'])
#print(snpA)
#snpAll[snpAll[0]=='rs12124819']
snpList = ('l','rs7419119', 'rs12124819')
print(len(snpList))
snpChoice = pd.DataFrame()
#snpChoice = []
print(snpChoice)
#d = [range(1,len(snpList),1)]
#print(d)
#print(range(1,len(snpList)))
#print(snpChoice[1])
print("for loop")
for i in range(0,len(snpList)):
    print(i)
#    nRow = snpAll[snpAll[0]==snpList[i]]
    nRow = snpAll.loc[snpAll[0] == snpList[i]]
#    snpChoice[len(snpChoice)] = nRow
    print(nRow)
#    snpChoice.append(nRow)
    snpChoice = pd.concat([snpChoice, nRow])
print(snpChoice)
```

```
                       0   1         2         3  4  5        6
```

```
0              rs3094315   1  0.020130    752566  G  A        1
1             rs12124819   1  0.020242    776546  A  G        2
2             rs28765502   1  0.022137    832918  T  C        3
3              rs7419119   1  0.022518    842013  T  G        4
4               rs950122   1  0.022720    846864  G  C        5
…                      …  ..        …         … .. ..        …
1233008       rs60847530  24  0.000000  59030572  T  A  1233009
1233009  snp_24_59033099  24  0.000000  59033099  T  C  1233010
1233010       rs28628009  24  0.000000  59033110  A  T  1233011
1233011       rs55686319  24  0.000000  59033139  T  C  1233012
1233012       rs75089321  24  0.000000  59033249  G  T  1233013

[1233013 rows x 7 columns]
```

```
3
Empty DataFrame
Columns: []
Index: []
for loop
0
Empty DataFrame
Columns: [0, 1, 2, 3, 4, 5, 6]
Index: []
1
           0  1         2       3  4  5  6
3  rs7419119  1  0.022518  842013  T  G  4
2
            0  1         2       3  4  5  6
1  rs12124819  1  0.020242  776546  A  G  2
            0  1         2       3  4  5  6
3   rs7419119  1  0.022518  842013  T  G  4
1  rs12124819  1  0.020242  776546  A  G  2
```

[99]:

```
import csv
import pandas as pd
snpAll = pd.read_csv('gs://ancient-dna-bucket/v62.snp', sep="\s+", header=None)
snpAll[6] = range(1,1233014)
display(snpAll)
#snpA = snp[0].isin(['rs12124819'])
#print(snpA)
#snpAll[snpAll[0]=='rs12124819']
snpList = ('ss820496565', 'rs759157971', 'rs4954490', 'rs56348046', 'rs4954492',
           'rs527991977', 'rs4988233', 'rs182549', 'rs41525747', 'rs4988235',
           'rs41380347', 'rs869051967', 'rs145946881')
print(len(snpList))
snpChoice = pd.DataFrame()
#snpChoice = []
print(snpChoice)
```

```
#d = [range(1,len(snpList),1)]
#print(d)
#print(range(1,len(snpList)))
#print(snpChoice[1])
print("for loop")
for i in range(0,len(snpList)):
    print(i)
#    nRow = snpAll[snpAll[0]==snpList[i]]
    nRow = snpAll.loc[snpAll[0] == snpList[i]]
#    snpChoice[len(snpChoice)] = nRow
    print(nRow)
#    snpChoice.append(nRow)
    snpChoice = pd.concat([snpChoice, nRow])
print(snpChoice)
```

```
                       0   1         2         3  4  5        6
0              rs3094315   1  0.020130    752566  G  A        1
1             rs12124819   1  0.020242    776546  A  G        2
2             rs28765502   1  0.022137    832918  T  C        3
3              rs7419119   1  0.022518    842013  T  G        4
4               rs950122   1  0.022720    846864  G  C        5
…                    …  ..       …       …  ..  ..       …
1233008       rs60847530  24  0.000000  59030572  T  A  1233009
1233009  snp_24_59033099  24  0.000000  59033099  T  C  1233010
1233010       rs28628009  24  0.000000  59033110  A  T  1233011
1233011       rs55686319  24  0.000000  59033139  T  C  1233012
1233012       rs75089321  24  0.000000  59033249  G  T  1233013

[1233013 rows x 7 columns]

13
Empty DataFrame
Columns: []
Index: []
for loop
0
Empty DataFrame
Columns: [0, 1, 2, 3, 4, 5, 6]
Index: []
1
Empty DataFrame
Columns: [0, 1, 2, 3, 4, 5, 6]
Index: []
2
                0  1         2          3  4  5       6
149515  rs4954490  2  1.555436  136608231  G  A  149516
3
Empty DataFrame
```

```
Columns: [0, 1, 2, 3, 4, 5, 6]
Index: []
4
Empty DataFrame
Columns: [0, 1, 2, 3, 4, 5, 6]
Index: []
5
Empty DataFrame
Columns: [0, 1, 2, 3, 4, 5, 6]
Index: []
6
Empty DataFrame
Columns: [0, 1, 2, 3, 4, 5, 6]
Index: []
7
Empty DataFrame
Columns: [0, 1, 2, 3, 4, 5, 6]
Index: []
8
                 0  1         2          3  4  5       6
149518  rs41525747  2  1.555436  136608643  G  C  149519
9
                0  1         2          3  4  5       6
149520  rs4988235  2  1.555436  136608646  G  A  149521
10
                 0  1         2          3  4  5       6
149522  rs41380347  2  1.555436  136608651  A  C  149523
11
Empty DataFrame
Columns: [0, 1, 2, 3, 4, 5, 6]
Index: []
12
                  0  1         2          3  4  5       6
149523  rs145946881  2  1.555436  136608746  C  G  149524
                  0  1         2          3  4  5       6
149515    rs4954490  2  1.555436  136608231  G  A  149516
149518   rs41525747  2  1.555436  136608643  G  C  149519
149520    rs4988235  2  1.555436  136608646  G  A  149521
149522   rs41380347  2  1.555436  136608651  A  C  149523
149523  rs145946881  2  1.555436  136608746  C  G  149524
```

[38]:
```python
import csv
import pandas as pd
snpAll = pd.read_csv('gs://ancient-dna-bucket/v62.snp', sep="\s+", header=None)
snpAll[6] = range(1,1233014)
display(snpAll)
snpList = ('ss820496565', 'rs759157971', 'rs4954490', 'rs56348046', 'rs4954492',
           'rs527991977', 'rs4988233', 'rs182549', 'rs41525747', 'rs4988235',
           'rs41380347', 'rs869051967', 'rs145946881')
snpChoice = pd.DataFrame()
for i in range(0,len(snpList)):
    nRow = snpAll.loc[snpAll[0] == snpList[i]]
    snpChoice = pd.concat([snpChoice, nRow])
print(snpChoice)
```

```
                       0   1         2         3  4  5        6
0              rs3094315   1  0.020130    752566  G  A        1
1             rs12124819   1  0.020242    776546  A  G        2
2             rs28765502   1  0.022137    832918  T  C        3
3              rs7419119   1  0.022518    842013  T  G        4
4               rs950122   1  0.022720    846864  G  C        5
…                    …  ..        …        … .. ..        …
1233008       rs60847530  24  0.000000  59030572  T  A  1233009
1233009  snp_24_59033099  24  0.000000  59033099  T  C  1233010
1233010       rs28628009  24  0.000000  59033110  A  T  1233011
1233011       rs55686319  24  0.000000  59033139  T  C  1233012
1233012       rs75089321  24  0.000000  59033249  G  T  1233013

[1233013 rows x 7 columns]

                  0  1         2          3  4  5       6
149515    rs4954490  2  1.555436  136608231  G  A  149516
149518   rs41525747  2  1.555436  136608643  G  C  149519
149520    rs4988235  2  1.555436  136608646  G  A  149521
149522   rs41380347  2  1.555436  136608651  A  C  149523
149523  rs145946881  2  1.555436  136608746  C  G  149524
```

[ ]:

```
genoAll = pd.read_csv('gs://ancient-dna-bucket/v62.eigenstratgeno', sep="\r")
print(genoAll.head(1))
```

[39]:

```
import csv
import pandas as pd
snpGeno = pd.read_csv('gs://ancient-dna-bucket/snps.geno', sep="\s+",␣
  ↪header=None)
print(snpChoice)
print(snpGeno)
#snpGeno.set_index(0, inplace=True)
snpChoice = snpChoice.reset_index()
print(snpChoice)
print(snpGeno.head(1))
combo = pd.concat([snpChoice, snpGeno], axis=1)
#combo = snpChoice.assign('7'=snpGeno('0'))
snpChoice[7] = snpGeno[0]
snpChoice[8] = snpGeno[1]
print(snpChoice)
```

```
#print(combo.head(1))
#print(combo)
snpChoice.to_csv('snpChoice.tsv', sep='\t', index=False)
```

```
                  0  1         2          3  4  5       6
149515    rs4954490  2  1.555436  136608231  G  A  149516
149518   rs41525747  2  1.555436  136608643  G  C  149519
149520    rs4988235  2  1.555436  136608646  G  A  149521
149522   rs41380347  2  1.555436  136608651  A  C  149523
149523  rs145946881  2  1.555436  136608746  C  G  149524
        0                                                1
0  149516  2222222999999999999920990022292222299209990029…
1  149519  2222222929992999999922992222222222292222222229…
2  149521  2222220929992999999922990020222222292202022029…
3  149523  2222222929992999999922292222222222292222222229…
4  149524  2222222999999999999922222222292222929922222922…
    index            0  1         2          3  4  5       6
0  149515    rs4954490  2  1.555436  136608231  G  A  149516
1  149518   rs41525747  2  1.555436  136608643  G  C  149519
2  149520    rs4988235  2  1.555436  136608646  G  A  149521
3  149522   rs41380347  2  1.555436  136608651  A  C  149523
4  149523  rs145946881  2  1.555436  136608746  C  G  149524
        0                                                1
0  149516  2222222999999999999920990022292222299209990029…
    index            0  1         2          3  4  5       6       7  \
0  149515    rs4954490  2  1.555436  136608231  G  A  149516  149516
1  149518   rs41525747  2  1.555436  136608643  G  C  149519  149519
2  149520    rs4988235  2  1.555436  136608646  G  A  149521  149521
3  149522   rs41380347  2  1.555436  136608651  A  C  149523  149523
4  149523  rs145946881  2  1.555436  136608746  C  G  149524  149524

                                                 8
0  2222222999999999999920990022292222299209990029…
1  2222222929992999999922992222222222292222222229…
2  2222220929992999999922990020222222292202022029…
3  2222222929992999999922292222222222292222222229…
4  2222222999999999999922222222292222929922222922…
```

[40]:
```
import csv
import pandas as pd
snpAll = pd.read_csv('gs://ancient-dna-bucket/v62.snp', sep="\s+", header=None)
snpAll[6] = range(1,1233014)
display(snpAll)
snpList = ('ss820496565', 'rs759157971', 'rs4954490', 'rs56348046', 'rs4954492',
           'rs527991977', 'rs4988233', 'rs182549', 'rs41525747', 'rs4988235',
           'rs41380347', 'rs869051967', 'rs145946881')
snpChoice = pd.DataFrame()
```

```python
for i in range(0,len(snpList)):
    nRow = snpAll.loc[snpAll[0] == snpList[i]]
    snpChoice = pd.concat([snpChoice, nRow])
print(snpChoice)

import csv
import pandas as pd
snpGeno = pd.read_csv('gs://ancient-dna-bucket/snps.geno', sep="\s+",␣
 ↪header=None)
print(snpChoice)
print(snpGeno)
#snpGeno.set_index(0, inplace=True)
snpChoice = snpChoice.reset_index()
print(snpChoice)
print(snpGeno.head(1))
combo = pd.concat([snpChoice, snpGeno], axis=1)
#combo = snpChoice.assign('7'=snpGeno('0'))
snpChoice[7] = snpGeno[0]
snpChoice[8] = snpGeno[1]
print(snpChoice)
#print(combo.head(1))
#print(combo)
snpChoice.to_csv('snpChoice.tsv', sep='\t', index=False)
```

```
                       0   1         2         3  4  5        6
0              rs3094315   1  0.020130    752566  G  A        1
1             rs12124819   1  0.020242    776546  A  G        2
2             rs28765502   1  0.022137    832918  T  C        3
3              rs7419119   1  0.022518    842013  T  G        4
4               rs950122   1  0.022720    846864  G  C        5
…                    …  ..         …       …  .. ..       …
1233008       rs60847530  24  0.000000  59030572  T  A  1233009
1233009  snp_24_59033099  24  0.000000  59033099  T  C  1233010
1233010       rs28628009  24  0.000000  59033110  A  T  1233011
1233011       rs55686319  24  0.000000  59033139  T  C  1233012
1233012       rs75089321  24  0.000000  59033249  G  T  1233013

[1233013 rows x 7 columns]
                  0  1         2          3  4  5       6
149515    rs4954490  2  1.555436  136608231  G  A  149516
149518   rs41525747  2  1.555436  136608643  G  C  149519
149520    rs4988235  2  1.555436  136608646  G  A  149521
149522   rs41380347  2  1.555436  136608651  A  C  149523
149523  rs145946881  2  1.555436  136608746  C  G  149524
                  0  1         2          3  4  5       6
149515    rs4954490  2  1.555436  136608231  G  A  149516
149518   rs41525747  2  1.555436  136608643  G  C  149519
```

```
149520    rs4988235  2  1.555436  136608646  G  A  149521
149522   rs41380347  2  1.555436  136608651  A  C  149523
149523  rs145946881  2  1.555436  136608746  C  G  149524
        0                                                   1
0  149516  2222222999999999999920990022292222299209990029…
1  149519  2222222929992999999922992222222222292222222229…
2  149521  2222220929992999999922990020222222292202022029…
3  149523  2222222929992999999922292222222222292222222229…
4  149524  2222222999999999999922222222292222929922222922…
    index            0  1         2          3  4  5       6
0  149515    rs4954490  2  1.555436  136608231  G  A  149516
1  149518   rs41525747  2  1.555436  136608643  G  C  149519
2  149520    rs4988235  2  1.555436  136608646  G  A  149521
3  149522   rs41380347  2  1.555436  136608651  A  C  149523
4  149523  rs145946881  2  1.555436  136608746  C  G  149524
        0                                                   1
0  149516  2222222999999999999920990022292222299209990029…
    index            0  1         2          3  4  5       6       7  \
0  149515    rs4954490  2  1.555436  136608231  G  A  149516  149516
1  149518   rs41525747  2  1.555436  136608643  G  C  149519  149519
2  149520    rs4988235  2  1.555436  136608646  G  A  149521  149521
3  149522   rs41380347  2  1.555436  136608651  A  C  149523  149523
4  149523  rs145946881  2  1.555436  136608746  C  G  149524  149524

                                                   8
0  2222222999999999999920990022292222299209990029…
1  2222222929992999999922992222222222292222222229…
2  2222220929992999999922990020222222292202022029…
3  2222222929992999999922292222222222292222222229…
4  2222222999999999999922222222292222929922222922…
```