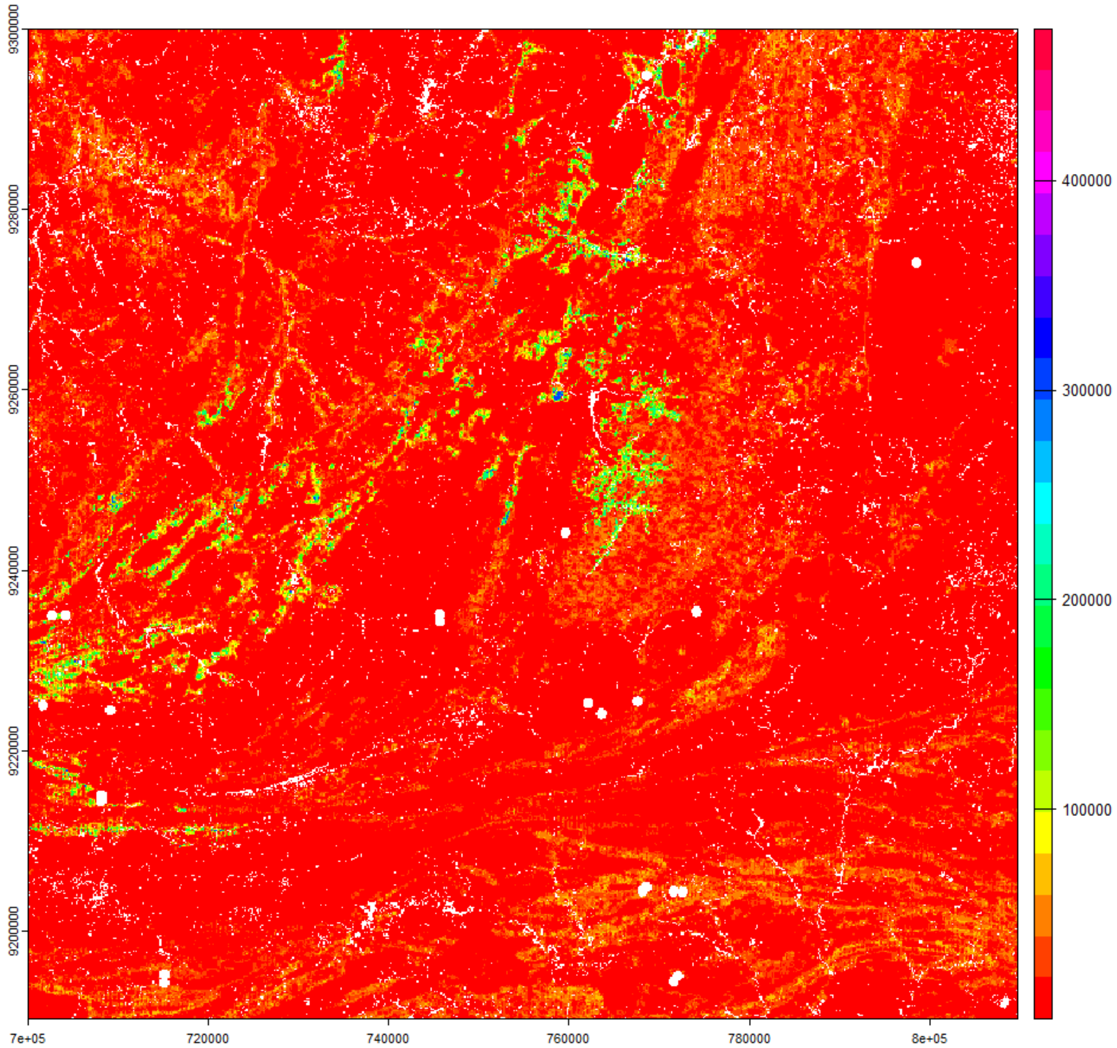
Mineral Exploration Targeting

PART 3 – Preliminary Target Areas Identification



André Luiz Costa, M.Sc, P.Geo., FAIG

Visit https://gdatasystems.com

PART 3

All we covered until now was data gathering and preparation, now the targeting process starts with the analysis and correlation. Aimed in quantifying a signature for a certain mineral commodity.

After preparing our base, it is time quantify on each layer of the prepared stack based on the presence of a certain occurrence. This is an important point, and a series of premises have to be taken into consideration. A good understanding of these premises will lead to a successful result.

Resolution (size) of mineralized target

The size of the expected mineralized zone is important to be evaluated. The resolution of 20 metres will resolve targets with at least 40 metres in size. Some layers have been interpolated to 20 metres from 500 metres linespacing (mag and gamma data). Keep always the resolution in mind when defining what size of deposit which you are willing to identify.

Vegetation

Although this is not the case in our example, some areas with heavy forested zones will affect drastically the spectral response from soil and rocks acting as a green shield. The use of airborne geophysics such as magnetometry and, in some cases, gammaspectrometry can assist in this situation.

Geological Characteristics of a mineral deposit

A fair understanding of what you are aiming for in the targeting process can only be given by the geological nature of the expected deposit. How do they occur? How deep? Is it primary, supergenic or epigenic? How big they are expected to be? Are they associated with fractures and faulting? Can its signature be detected by multi spectrometric response or geophysics? are they associated to a certain distance from a specific feature?

All the answers above will assist in choosing the right layer from the available targeting stack.

Sources of known data (training data)

Mineral occurrences data

Some geological surveys have a database available for downloading and they are the main source of evaluation/training data.

Existing mines and mining claims

Existing mines are the more reliable source, but they may be limited in some areas.

Soil-rock Geochemical data

This data may be available from some geological surveys or from proprietary exploration databases.

Web scraping

Web scraping is one of the most popular methods used to extract data from the world wide web. This can be achieved by using simple scripts directed to specific sites. Mostly publication sites and geological society sites.

Data Spatial Precision

Some geological surveys data are old and the error on the occurrence position can be significantly higher than what we expected. Keep in mind this fact and avoid low spatial precision.

Reliability of data

We must implement procedures like QAQC used for geochemistry and geological data. The quality of the resulting targeting map is directly corelated to the quality of training data used (crap-in/crap-out or CICO). This effect can be minimized by using statistical analysis and it will be shown in the next section.

What if no training data is available?

In some cases, we may get into a region where little or no data is available. Thus, we will have to interpret what range of values would be expected for each targeting layer based on geology and physical-chemical properties. This is not precise but can give us some hints about where to check first.

Targeting Layers Evaluation and Classification

Evaluate each layer from our stack and select those with a reasonable correlation with the training data. In this exercise we will stick with the **tungsten** mineral occurrences only. You can experiment later using other mineralization types from the mineral occurrence dataset or other dataset.

Tungsten in the region it is associated with Skarn deposits and the main mineral containing it is Scheelite **$Ca(WO_4)$**. The mineralization often occurs in the contact between a granitic intrusion and the host rock.

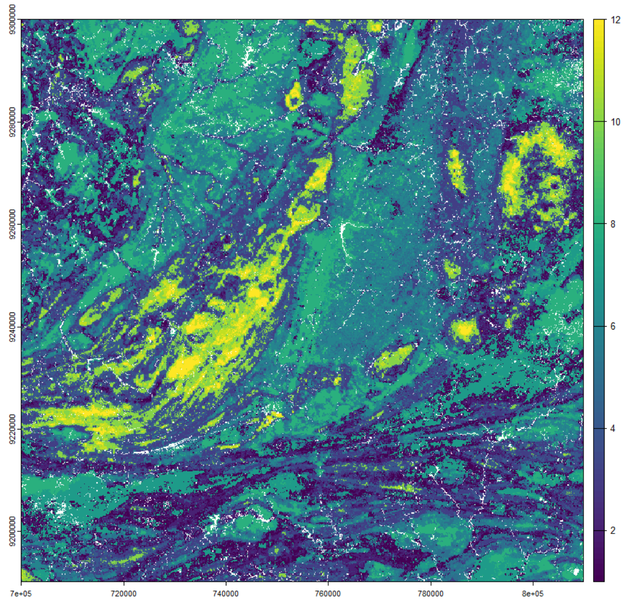
The statistical evaluation is one of the most effective procedures considering that we have a reasonable large sample of population based on the number of points for a certain mineral occurrence.

Extract the values from each layer and visually, from the histogram, select the most frequent values (or range of values). Reclassify each layer value as High (3), moderate (2) and low (1) correlation. When a layer is not conclusive, disregard it.

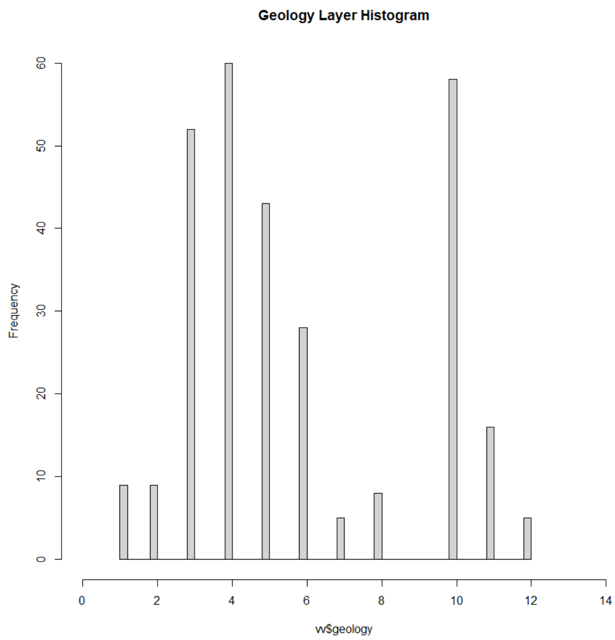
Geology

This geology layer represents the classification by quality and not quantity, thus we will select here the classes with higher frequency disregarding the data distribution.

```
library(terra)
#Working directory where the images and data are located
wd<-'C:/Users/User/Desktop/R algo/WDIR_P3'
setwd(wd)
train<-vect('tungst_occu.shp')
targ<-rast('targ_stack.tif')
geo1<-targ[[1]]
v <- na.omit(terra::extract(geo1, train,xy=T, method = "simple", ID=F))
vv<-vect(v,geom=c('x','y'))
plot(geo1)
```

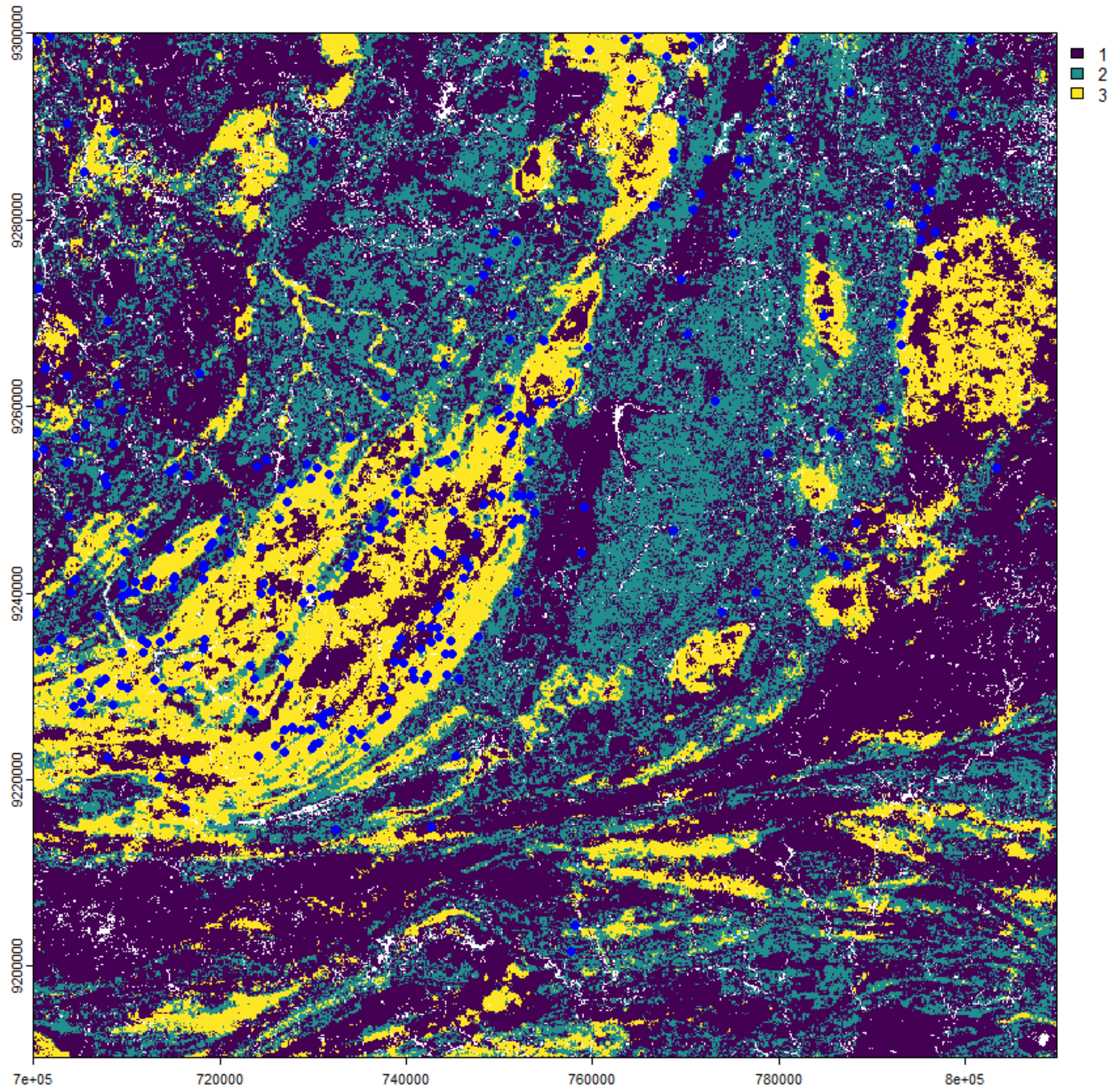



```
hist(vv$geology,xlim=c(0,14),n=70,main= 'Geology Layer Histogram')
```



The values 3, 4 and 10 will be considered **High** and a value of 3 will be assigned to them. Values 4 and 6 will be considered **Moderate** and a value of 2 will be assigned to them. The other values will be considered **Low** and the value of 1 will be assigned to them.

```
geoClass<-geo1
geoClass[geoClass==1 | geoClass ==2 | geoClass ==5 | geoClass ==7 | geoClass ==8 |
geoClass ==11 | geoClass ==12]<-1
geoClass[geoClass==3 | geoClass ==6 ]<-2
geoClass[geoClass ==4 | geoClass ==10]<-3
plot(geoClass)
plot(vv,add=T,col='blue')
```



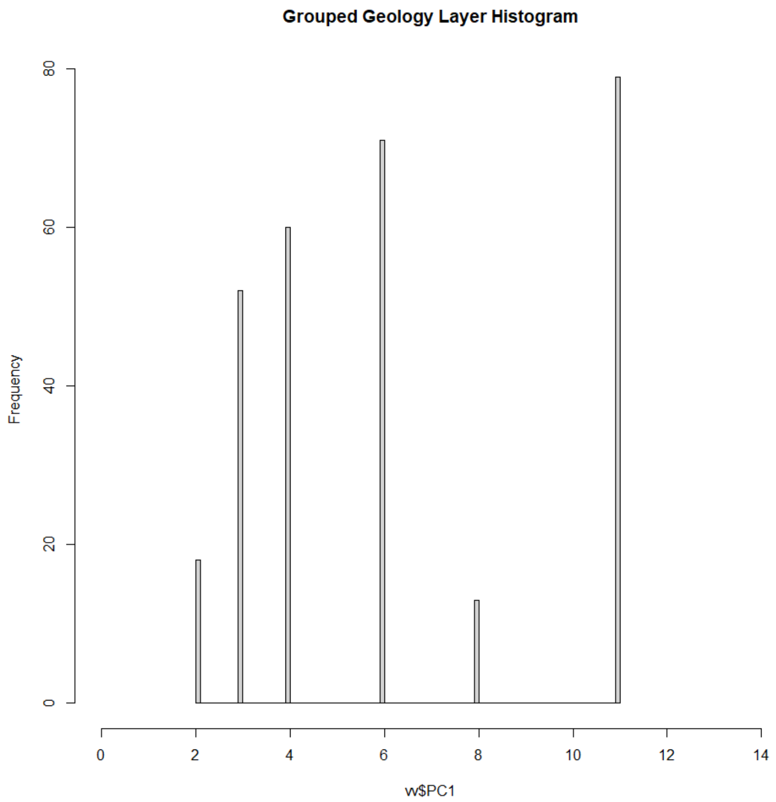

```
writeRaster(geoClass,'geoClass.tif', overwrite=TRUE)
```

Geology2

This second geology layer is a class grouping of closely related categories simplifying the layer above.

```
geo2<-targ[[2]]
```

```
v <- na.omit(terra::extract(geo2, train,xy=T, method = "simple", ID=F))
vv<-vect(v,geom=c('x','y'))
hist(vv$PC1,xlim=c(0,14),n=70 ,main='Grouped Geology Layer Histogram')
```

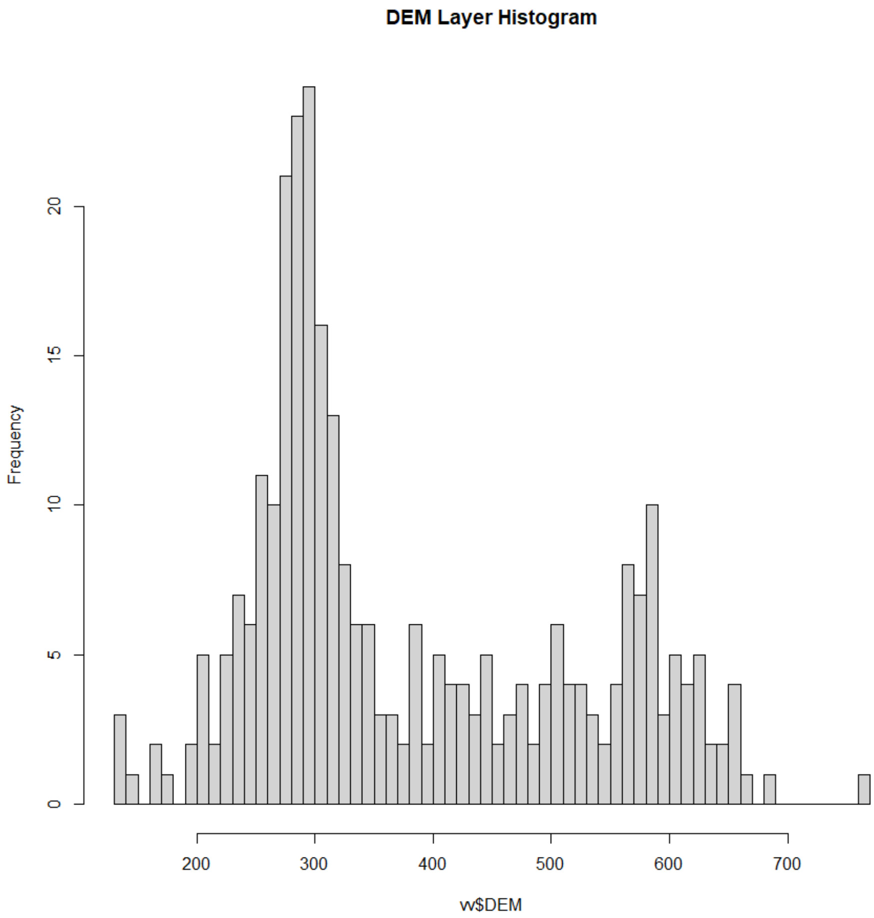


Compared to the geology layer above, the grouped Geology did not present a good correlation for Tungsten occurrences, and it will be disregarded.

DEM

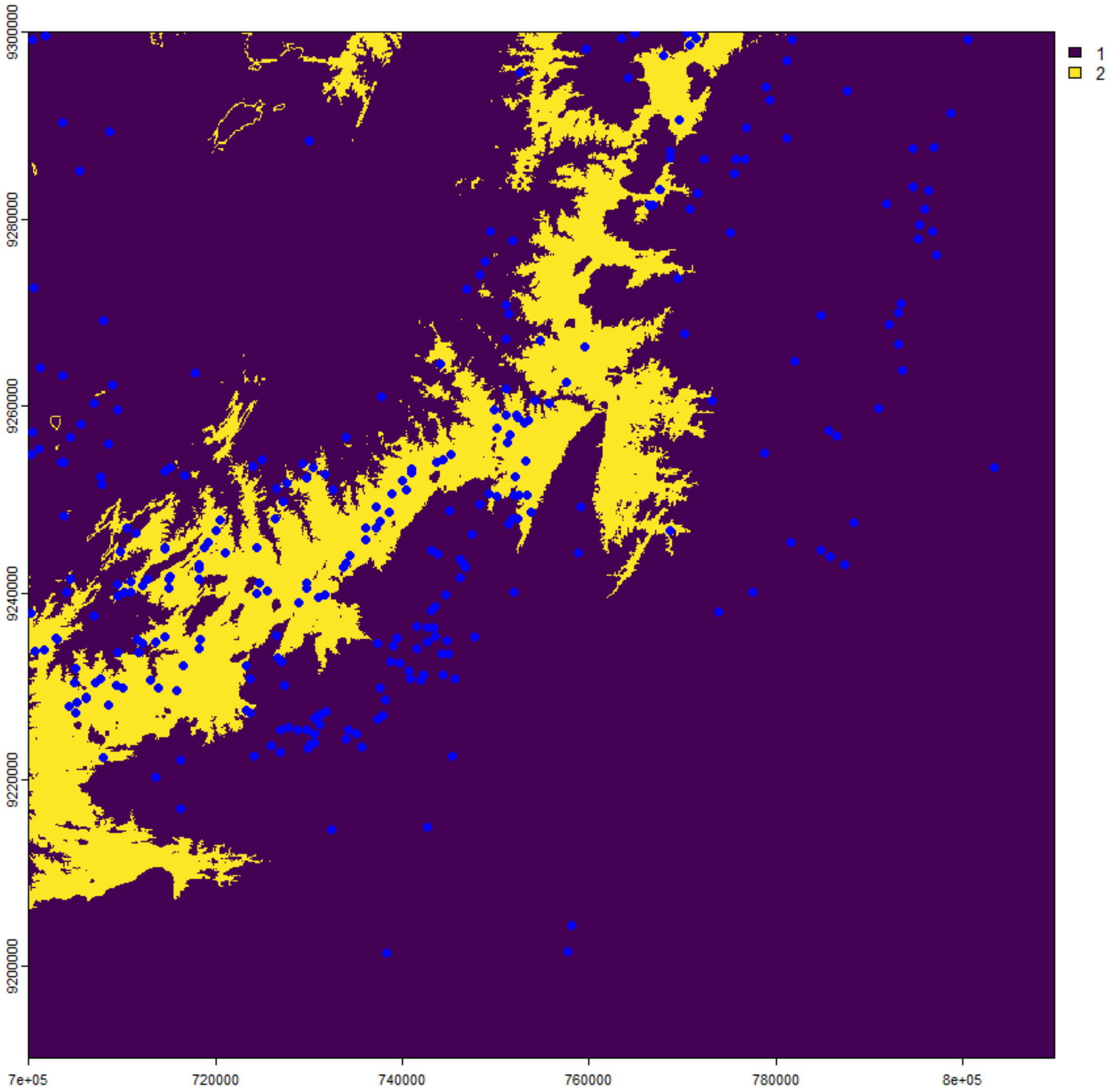
The next layer being evaluated is the Digital elevation model to check if we have a strong correlation between elevation and the tungsten occurrences.

```
dem<-targ[[3]]
v <- na.omit(terra::extract(dem, train,xy=T, method = "simple", ID=F))
vv<-vect(v,geom=c('x','y'))
hist(vv$DEM,n=70, main='DEM Layer Histogram')
```



Usually there is no correlation between the tungsten deposit and elevation but here we notice that between 270 and 330 we have a reasonable frequency of data, therefore we will assign this range as **Moderate** and the rest as **Low**.

```
demClass<-dem
demClass[demClass <=330 & demClass > 270 ]<-NA
demClass[!is.na(demClass)]<-1
demClass[is.na(demClass)]<-2
plot(demClass)
plot(vv,add=T,col='blue')
```

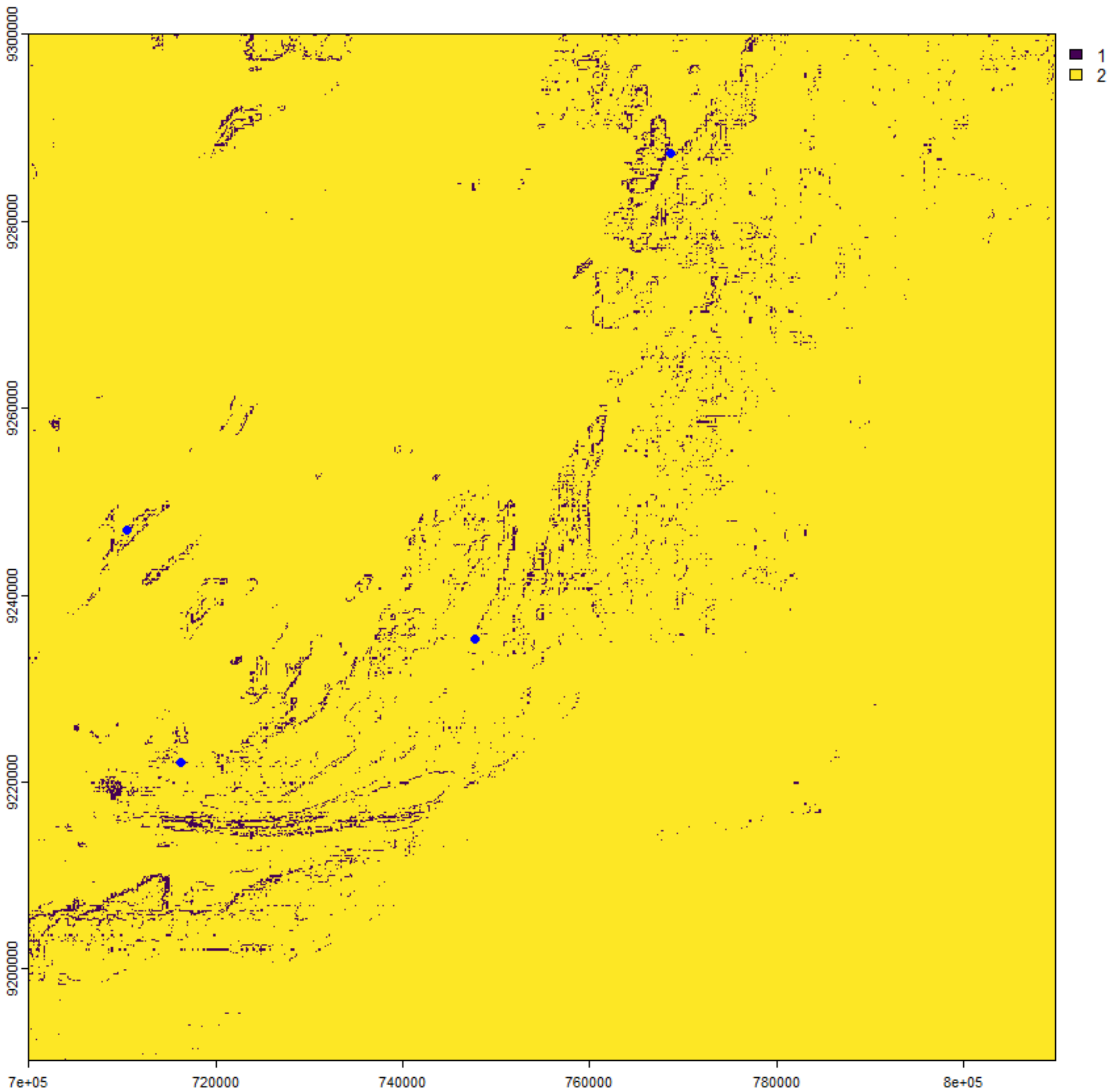



```
writeRaster(demClass,'demClass.tif', overwrite=TRUE)
```

Crests

The areas with higher angle slopes are evaluated in this layer. The result is relative to difficulty of accessing these zones, including the viability of implementing an extraction plant. Thus, the result here does not necessarily indicate that no tungsten is present but indicates that on slopes it is hard to implement. More an engineering condition than a geological one but it should be taken in consideration for targeting.

```
crests<-targ[[4]]
v <- na.omit(terra::extract(crests, train,xy=T, method = "simple", ID=F))
vv<-vect(v,geom=c('x','y'))
crestClass<-crests
crestClass[is.na(crestClass)]<-2
plot(crestClass)
plot(vv,add=T,col='blue')
```



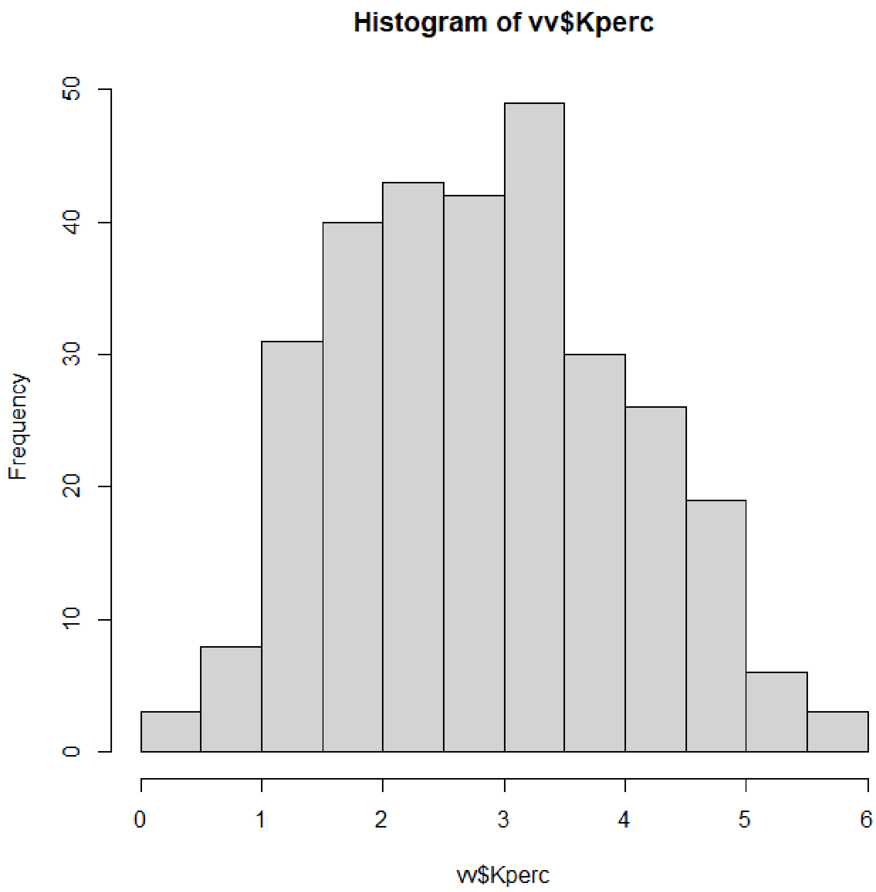

```
writeRaster(crestClass,'crestClass.tif', overwrite=TRUE)
```

Lines

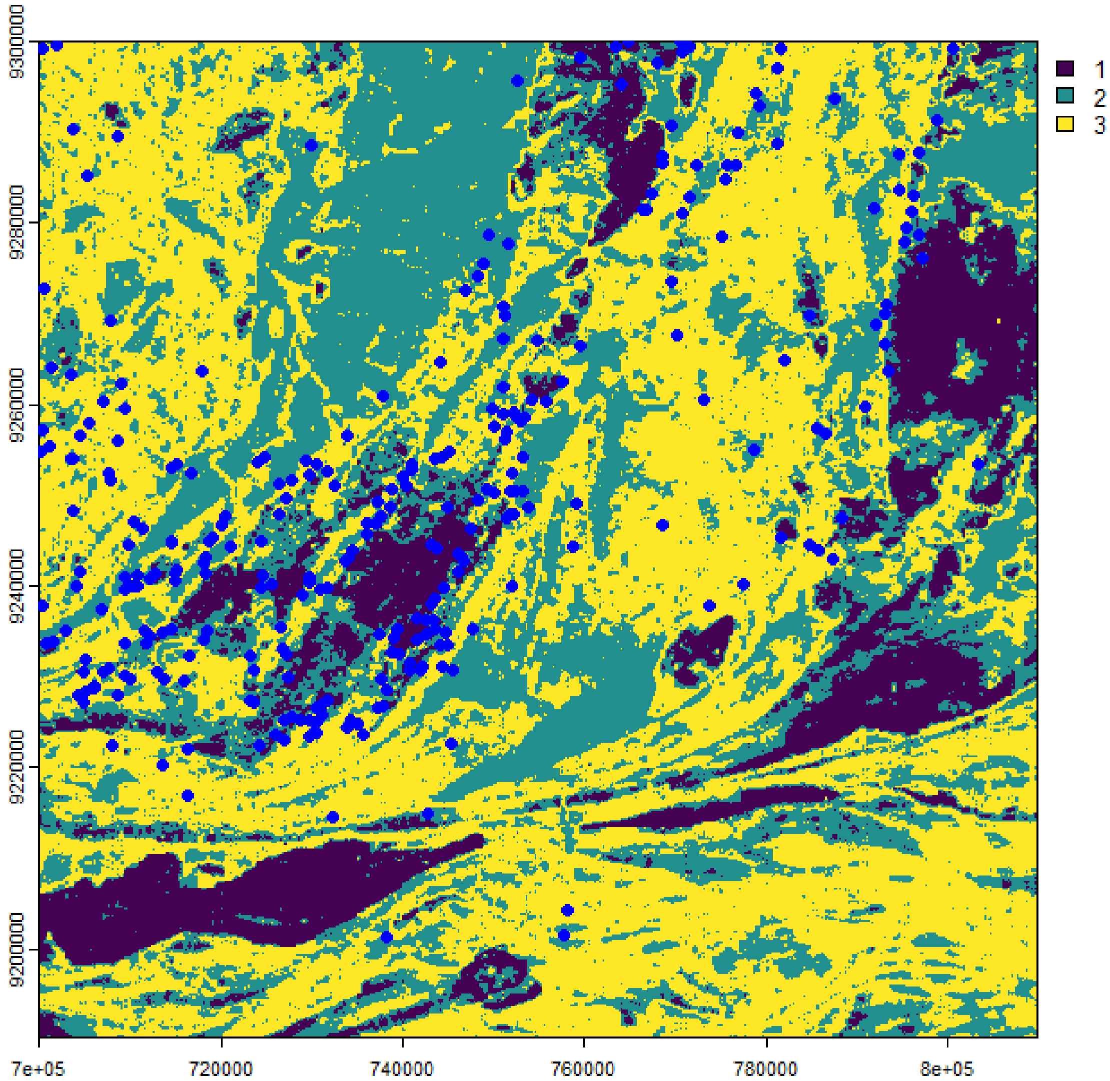
This layer could not indicate a correlation for tungsten occurrences. It does not state that there is no correlation between structural framework and mineralization, just indicates that there is no direct meaning in the way it is presented.

K

```
k<-targ[[6]]
v <- na.omit(terra::extract(k, train,xy=T, method = "simple", ID=F))
vv<-vect(v,geom=c('x','y'))
hist(vv$Kperc,n=10)# 1.5 to 3.5 good, 1 to 1.5 and 3.5 to 4.5 moderate
```



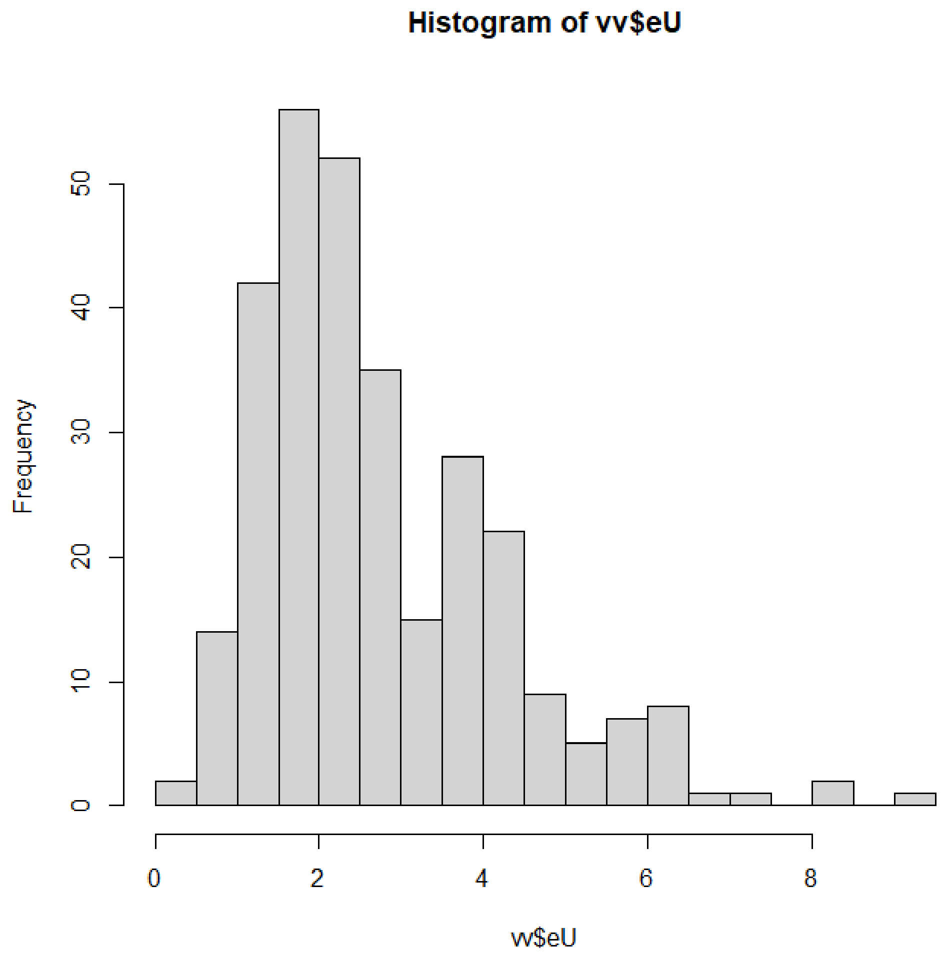

```
kClass <-k
kClass[kClass$Kperc >=1.5 & kClass$Kperc < 3.5]<-3000
kClass[kClass$Kperc < 1.5]<-2000
kClass[kClass$Kperc >=1 & kClass$Kperc < 1.5]<-2000
kClass[kClass$Kperc >=3.5 & kClass$Kperc < 4.5]<-2000
kClass[kClass$Kperc < 1]<-1000
kClass[kClass$Kperc >= 4.5 & kClass$Kperc<2000]<-1000
kClass[kClass$Kperc == 1000]<-1
kClass[kClass$Kperc == 2000]<-2
kClass[kClass$Kperc == 3000]<-3
plot(kClass)
plot(vv,add=T,col='blue')
```



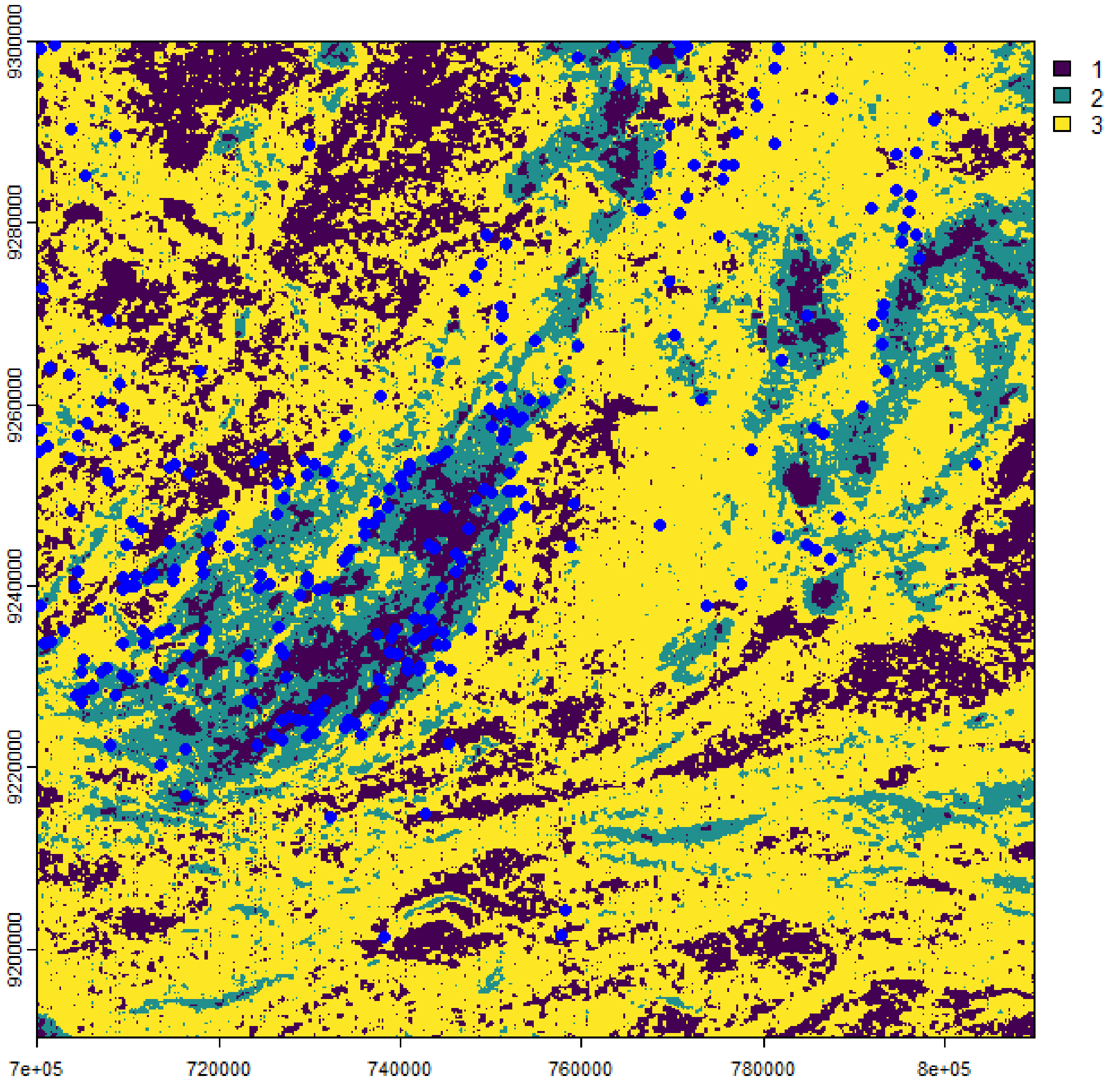

```
writeRaster(kClass,'kClass.tif', overwrite=TRUE)
```

U

```
u<-targ[[7]]
v <- na.omit(terra::extract(u, train,xy=T, method = "simple", ID=F))
vv<-vect(v,geom=c('x','y'))
hist(vv$eU,n=15)# 1 to 2.5 good, 2.5 to 4.5 moderate
```



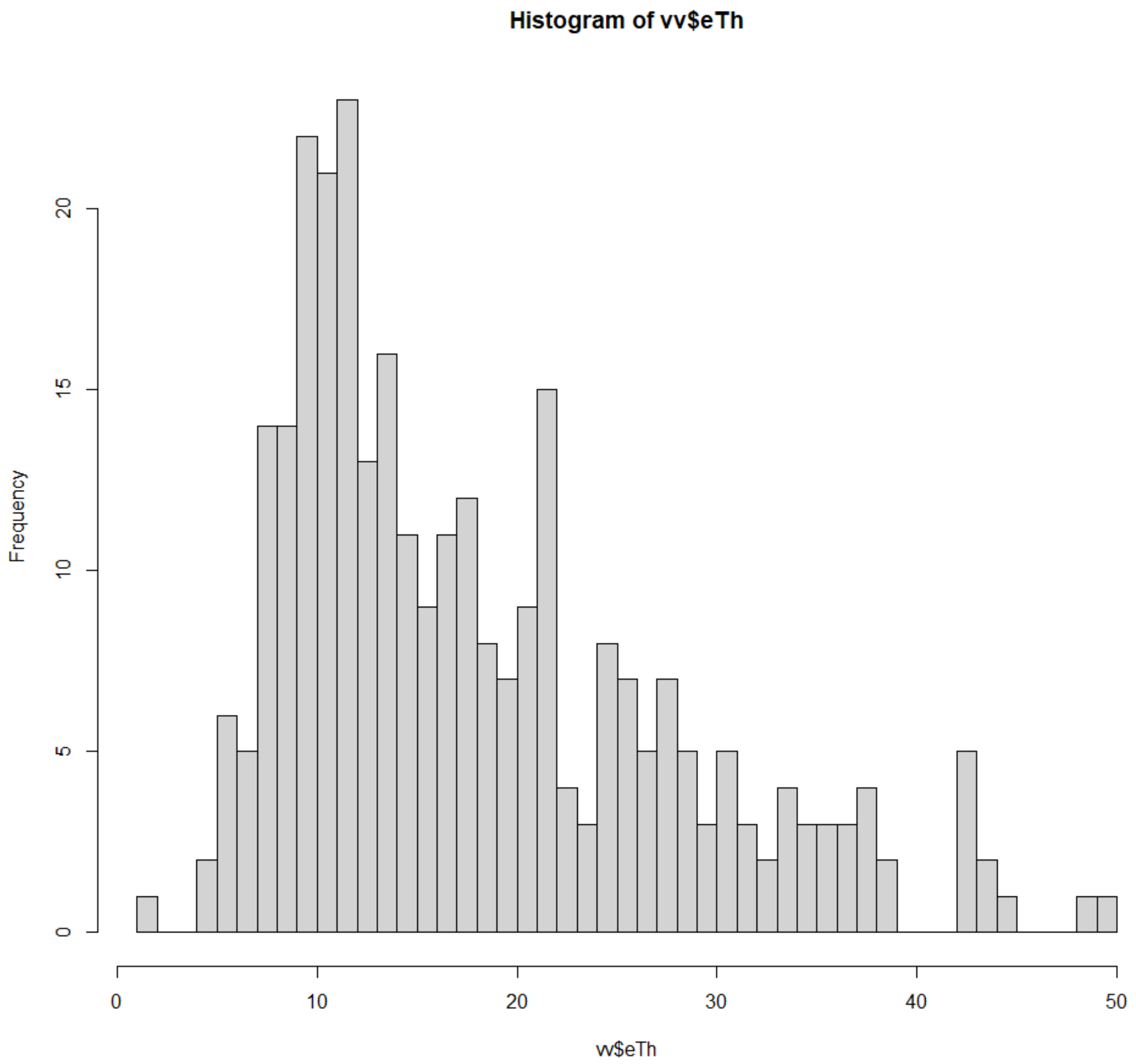

```
uClass<-u
uClass[uClass$eU >=1 & uClass$eU < 2.5]<-3000
uClass[uClass$eU >=2.5 & uClass$eU < 4.5]<-2000
uClass[uClass$eU < 1]<-1000
uClass[uClass$eU >= 4.5 & uClass$eU<2000]<-1000
uClass[uClass$eU == 1000]<-1
uClass[uClass$eU == 2000]<-2
uClass[uClass$eU == 3000]<-3
plot(uClass)
plot(vv,add=T,col='blue')
```



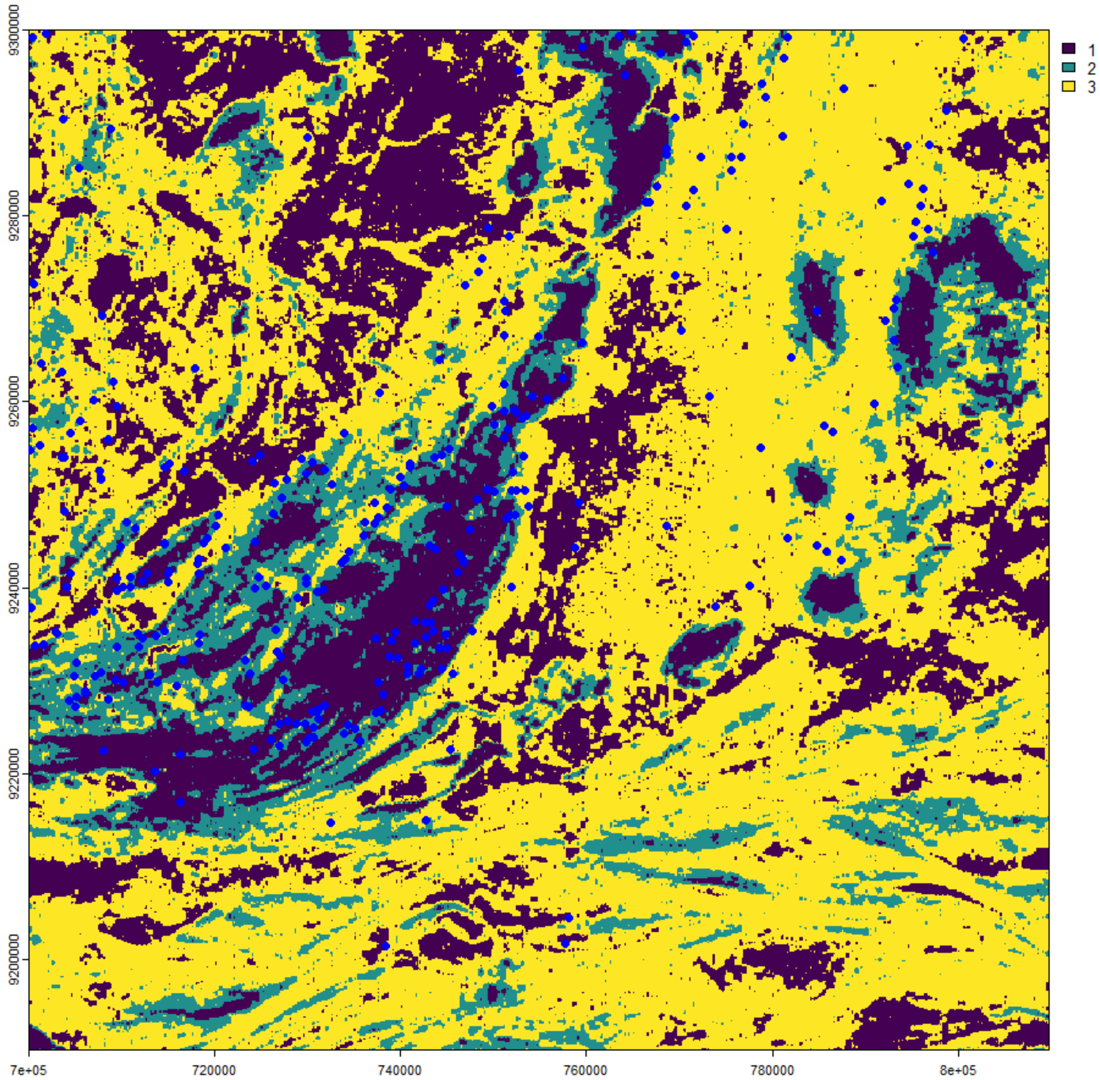

```
writeRaster(uClass, 'uClass.tif', overwrite=TRUE)
```

Th

```
th<-targ[[8]]
v <- na.omit(terra::extract(th, train,xy=T, method = "simple", ID=F))
vv<-vect(v,geom=c('x','y'))
hist(vv$eTh,n=50)# 7 to 14 good, 15 to 22 moderate
```



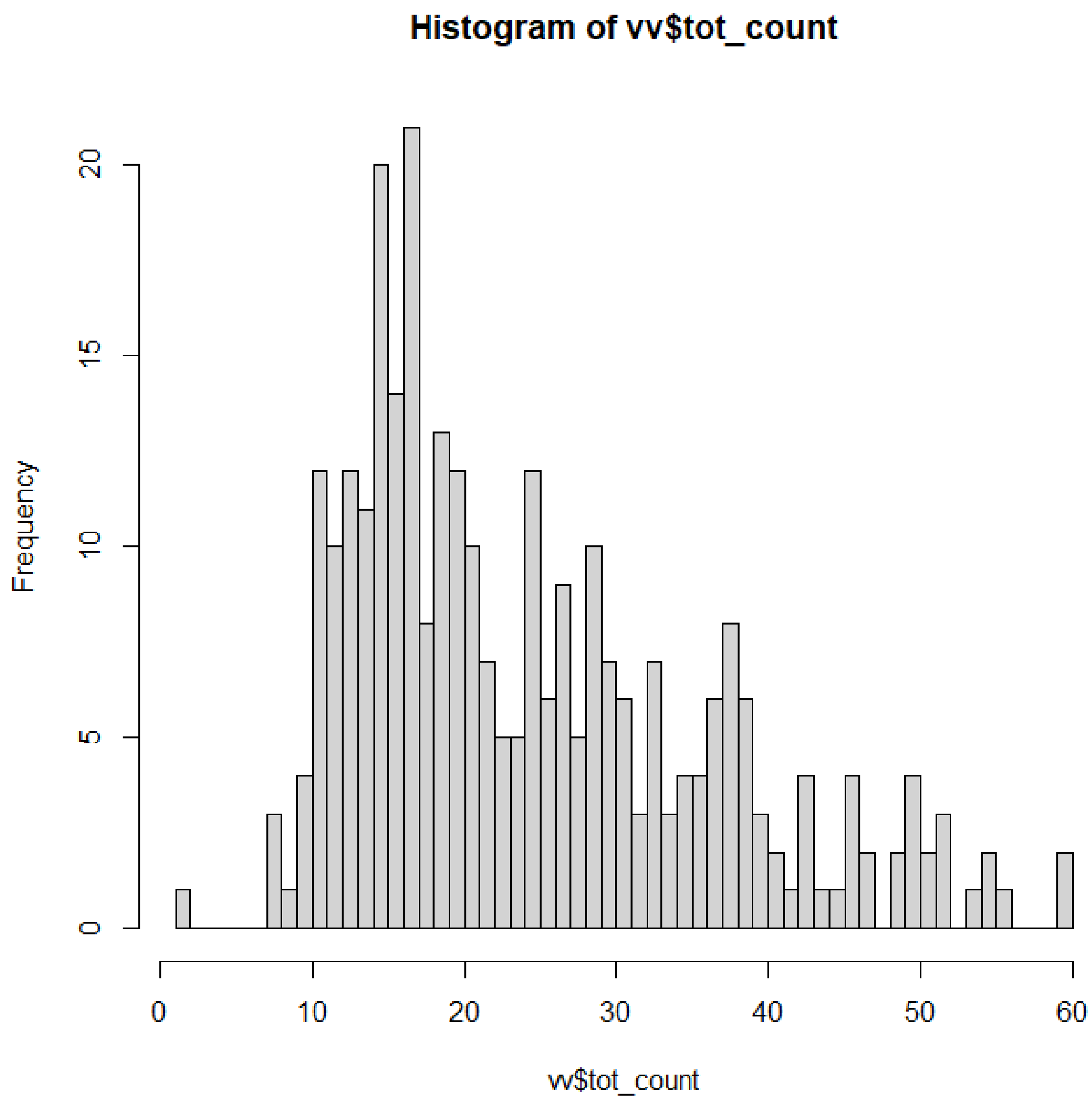

```
classTh<-th
classTh[classTh$eTh >=7 & classTh$eTh < 14]<-3000
classTh[classTh$eTh >=14 & classTh$eTh < 22]<-2000
classTh[classTh$eTh < 7]<-1000
classTh[classTh$eTh >= 22 & classTh$eTh<2000]<-1000
classTh[classTh$eTh == 1000]<-1
classTh[classTh$eTh == 2000]<-2
classTh[classTh$eTh == 3000]<-3
plot(classTh)
plot(vv,add=T,col='blue')
```



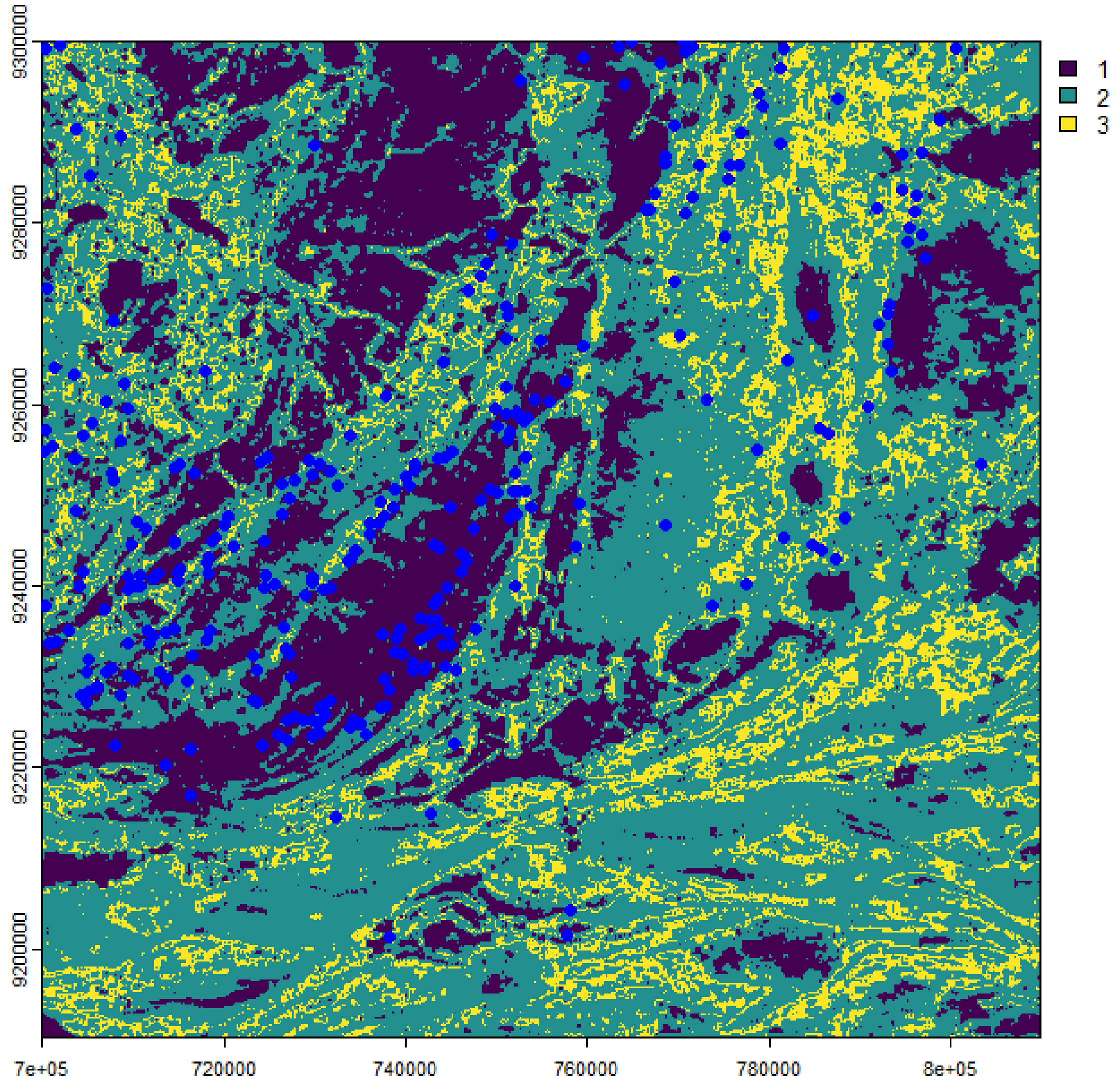

```
writeRaster(classTh, 'thClass.tif', overwrite=TRUE)
```

Total Count

```
tc<-targ[[9]]
v <- na.omit(terra::extract(tc, train,xy=T, method = "simple", ID=F))
vv<-vect(v,geom=c('x','y'))
hist(vv$tot_count,n=50)# 14 to 16 good, 10 to 13 and 17 to 28 moderate
```



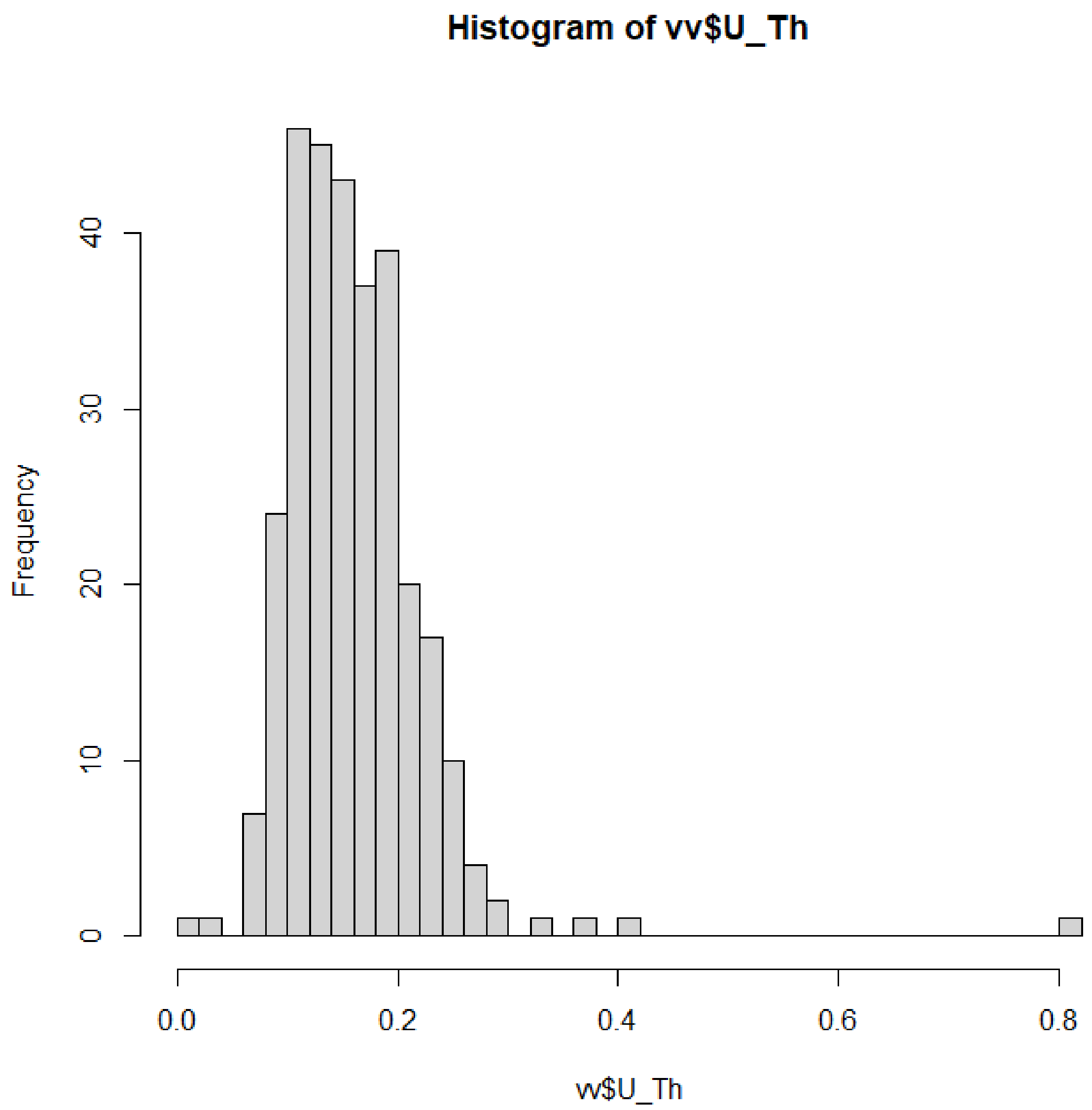

```
classTc<-tc
classTc[classTc$tot_count >=14 & classTc$tot_count < 16]<-3000
classTc[classTc$tot_count >=10 & classTc$tot_count < 14]<-2000
classTc[classTc$tot_count >=16 & classTc$tot_count < 28]<-2000
classTc[classTc$tot_count < 10]<-1000
classTc[classTc$tot_count >= 28 & classTc$tot_count<2000]<-1000
classTc[classTc$tot_count == 1000]<-1
classTc[classTc$tot_count == 2000]<-2
classTc[classTc$tot_count == 3000]<-3
plot(classTc)
plot(vv,add=T,col='blue')
```



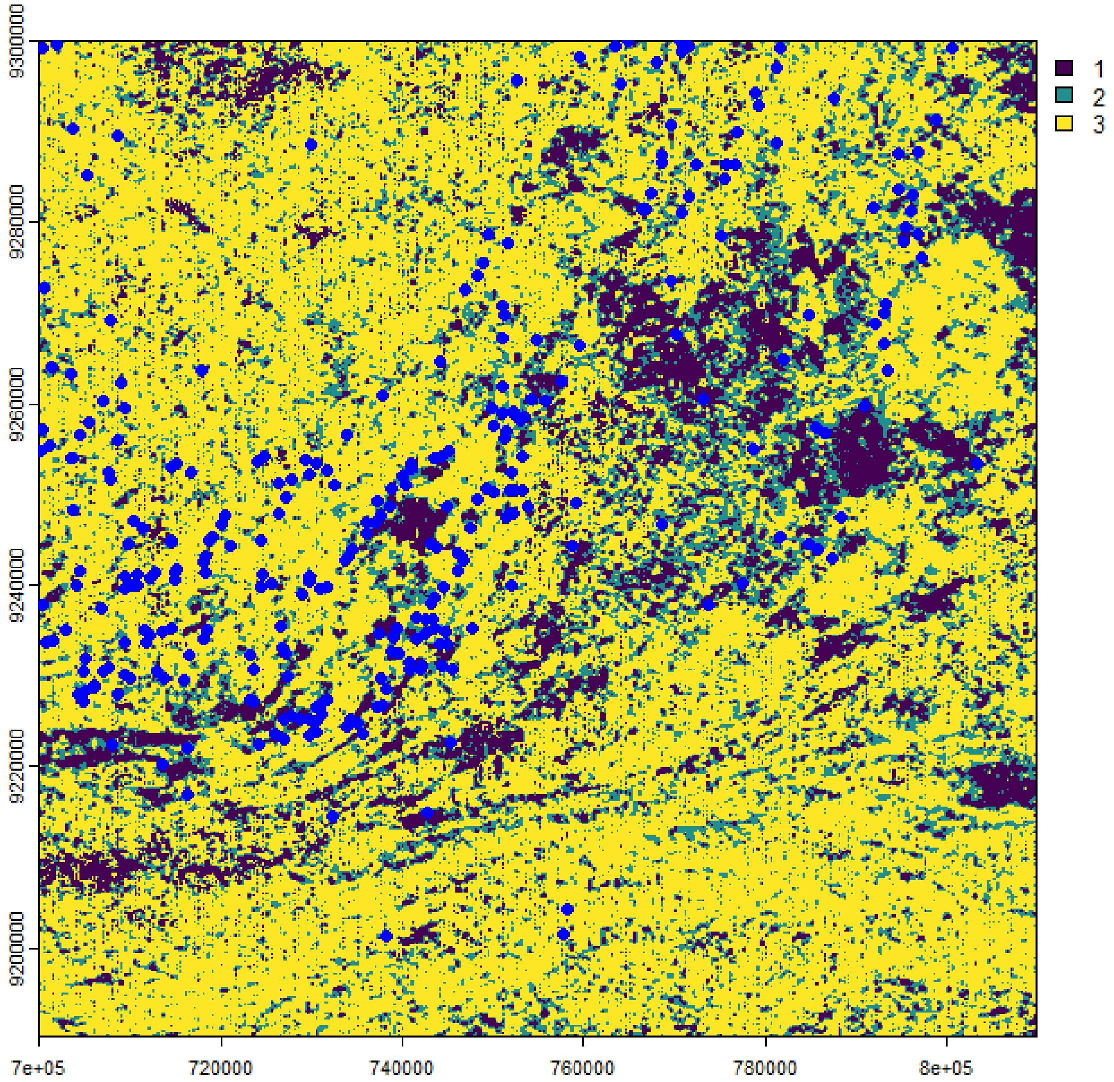

```
writeRaster(classTc, 'tcClass.tif', overwrite=TRUE)
```

U/Th

```
rtUTh<-targ[[10]]
v <- na.omit(terra::extract(rtUTh, train,xy=T, method = "simple", ID=F))
vv<-vect(v,geom=c('x','y'))
hist(vv$U_Th,n=50)# 0.1 to 0.2 good, 0.08 to 0.1 and 0.2 to 0.24 moderate
```



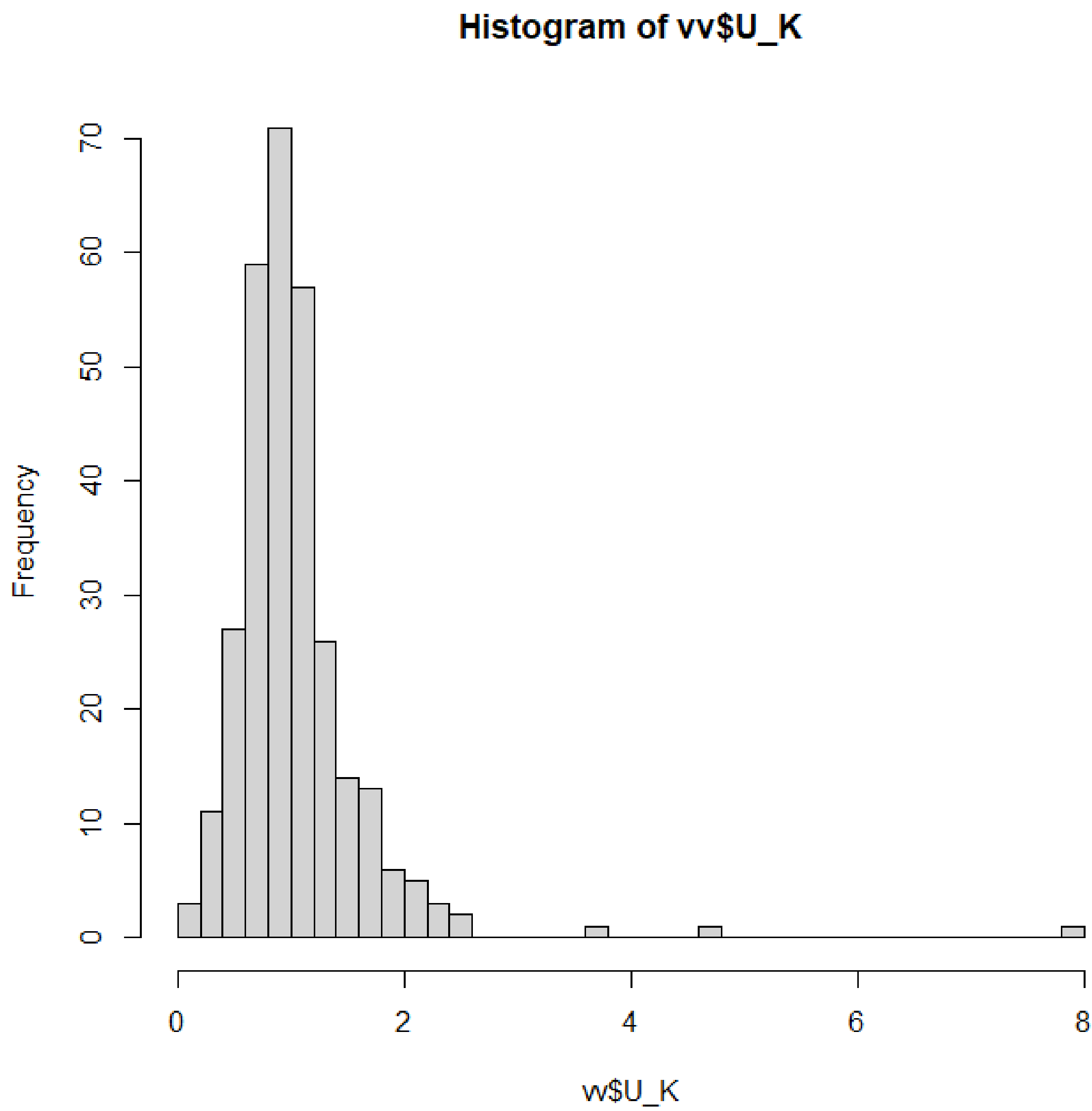

```
classUth<-rtUTh
classUth[classUth$U_Th >=0.1 & classUth$U_Th < 0.2]<-3000
classUth[classUth$U_Th >=0.08 & classUth$U_Th < 0.1]<-2000
classUth[classUth$U_Th >=0.2 & classUth$U_Th < 0.24]<-2000
classUth[classUth$U_Th < 0.08]<-1000
classUth[classUth$U_Th >= 0.24 & classUth$U_Th<2000]<-1000
classUth[classUth$U_Th == 1000]<-1
classUth[classUth$U_Th == 2000]<-2
classUth[classUth$U_Th == 3000]<-3
plot(classUth)
plot(vv,add=T,col='blue')
```



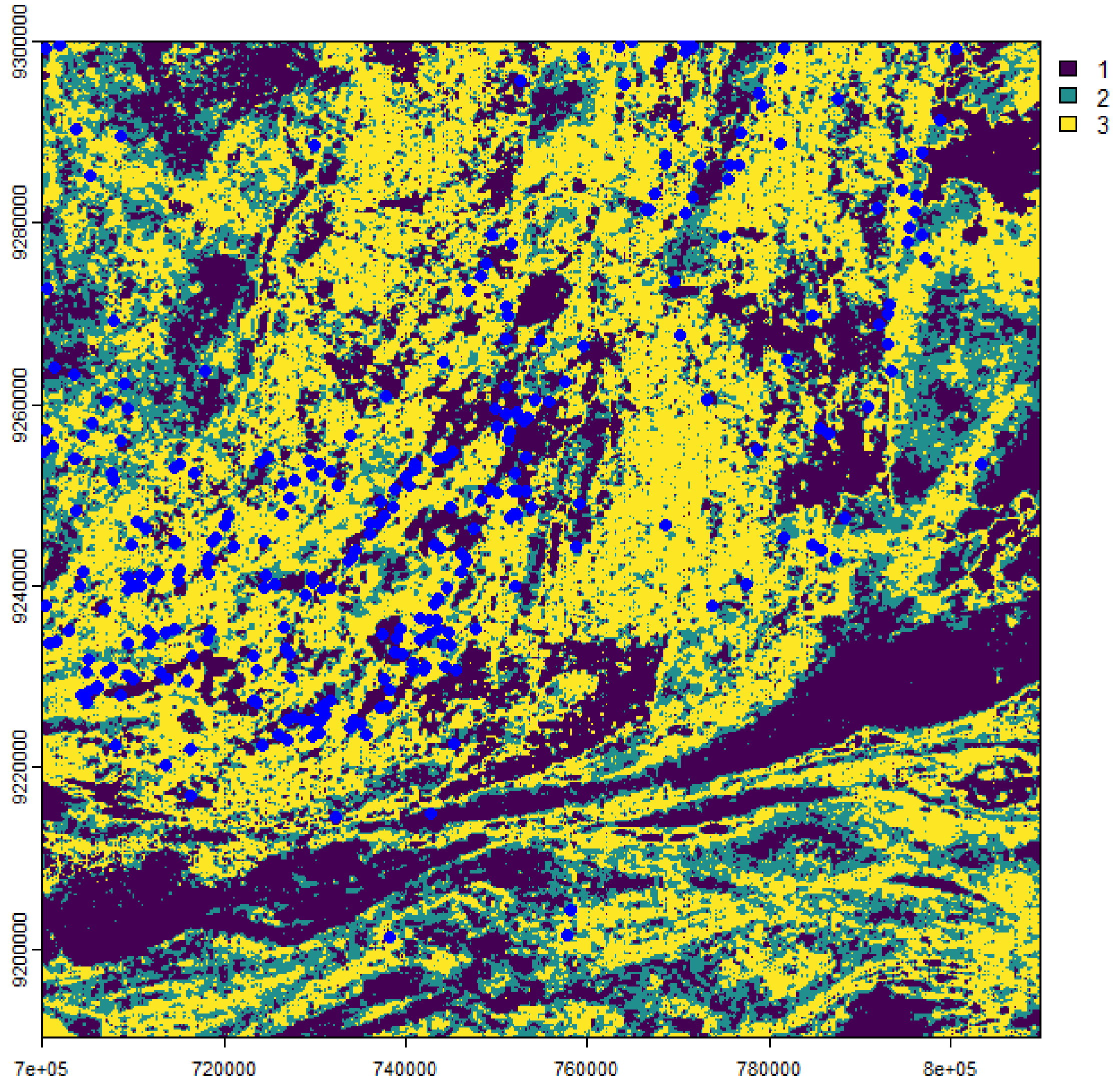

```
writeRaster(classUth, 'uthClass.tif', overwrite=TRUE)
```

U/K

```
rtUK<-targ[[11]]
v <- na.omit(terra::extract(rtUK, train,xy=T, method = "simple", ID=F))
vv<-vect(v,geom=c('x','y'))
hist(vv$U_K,n=50)# 0.6 to 1.2 good,0.4 to 0.6 and 1,2 to 1.4 moderate
```



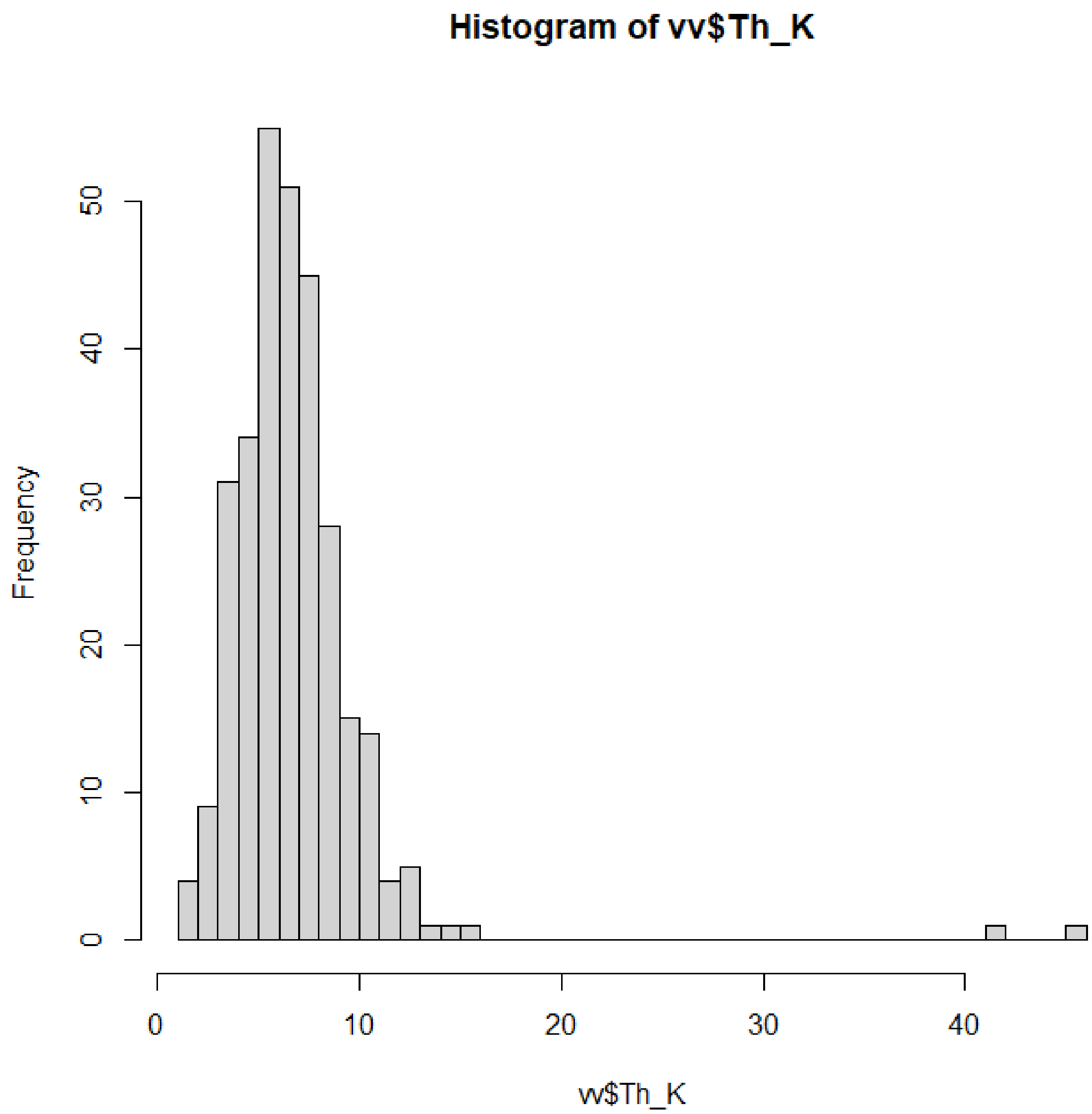

```
classUk<-rtUK
classUk[classUk$U_K >=0.6 & classUk$U_K < 1.2]<-3000
classUk[classUk$U_K >=0.4 & classUk$U_K < 0.6]<-2000
classUk[classUk$U_K >=1.2 & classUk$U_K < 1.4]<-2000
classUk[classUk$U_K < 0.4]<-1000
classUk[classUk$U_K >= 1.4 & classUk$U_K<2000]<-1000
classUk[classUk$U_K == 1000]<-1
classUk[classUk$U_K == 2000]<-2
classUk[classUk$U_K == 3000]<-3
plot(classUk)
plot(vv,add=T,col='blue')
```



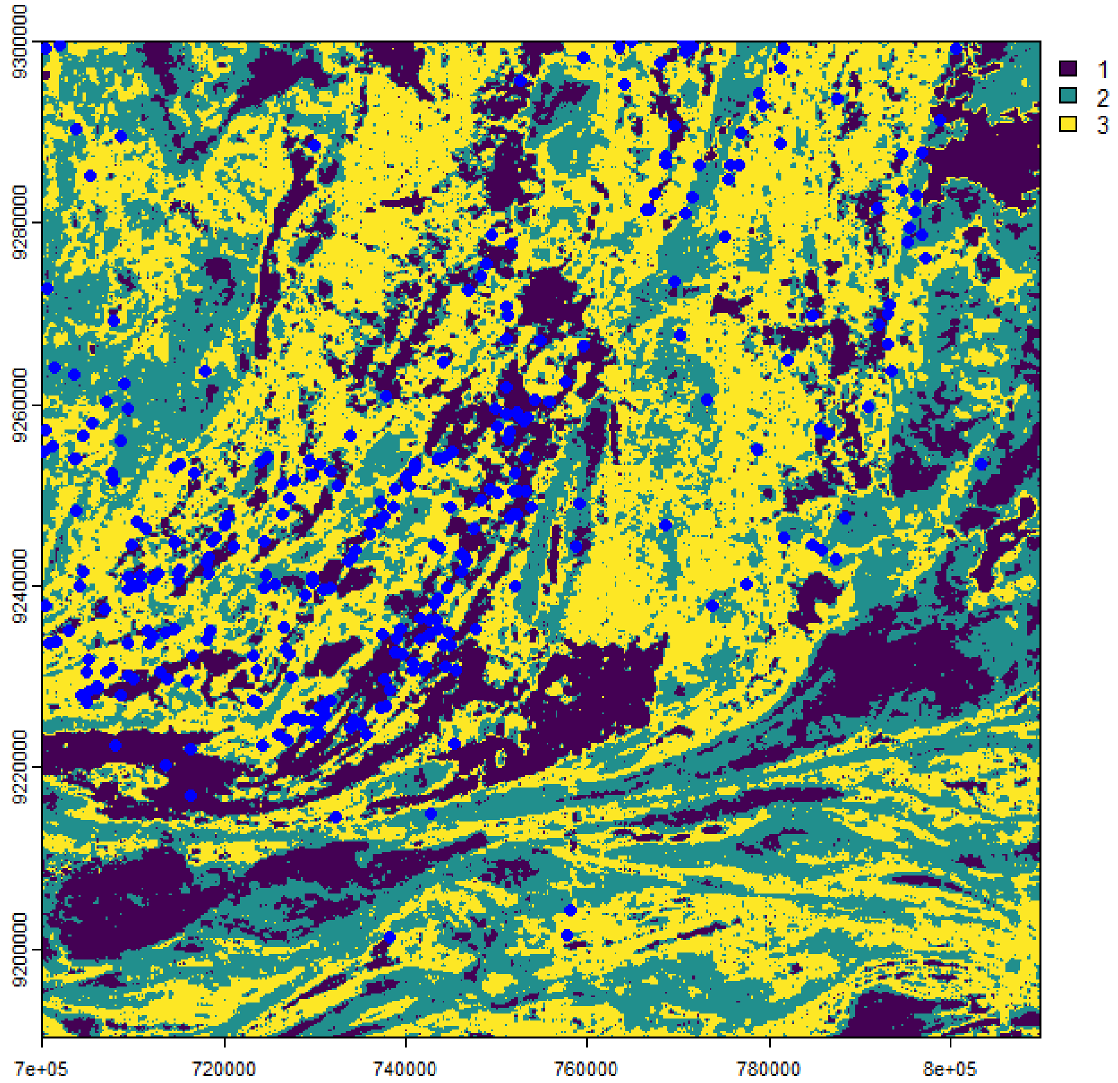

```
writeRaster(classUk, 'ukClass.tif', overwrite=TRUE)
```

Th/K

```
rtThK<-targ[[12]]
v <- na.omit(terra::extract(rtThK, train,xy=T, method = "simple", ID=F))
vv<-vect(v,geom=c('x','y'))
hist(vv$Th_K,n=50)# 4 to 7 good,2 to 4 and 7 to 8 moderate
```



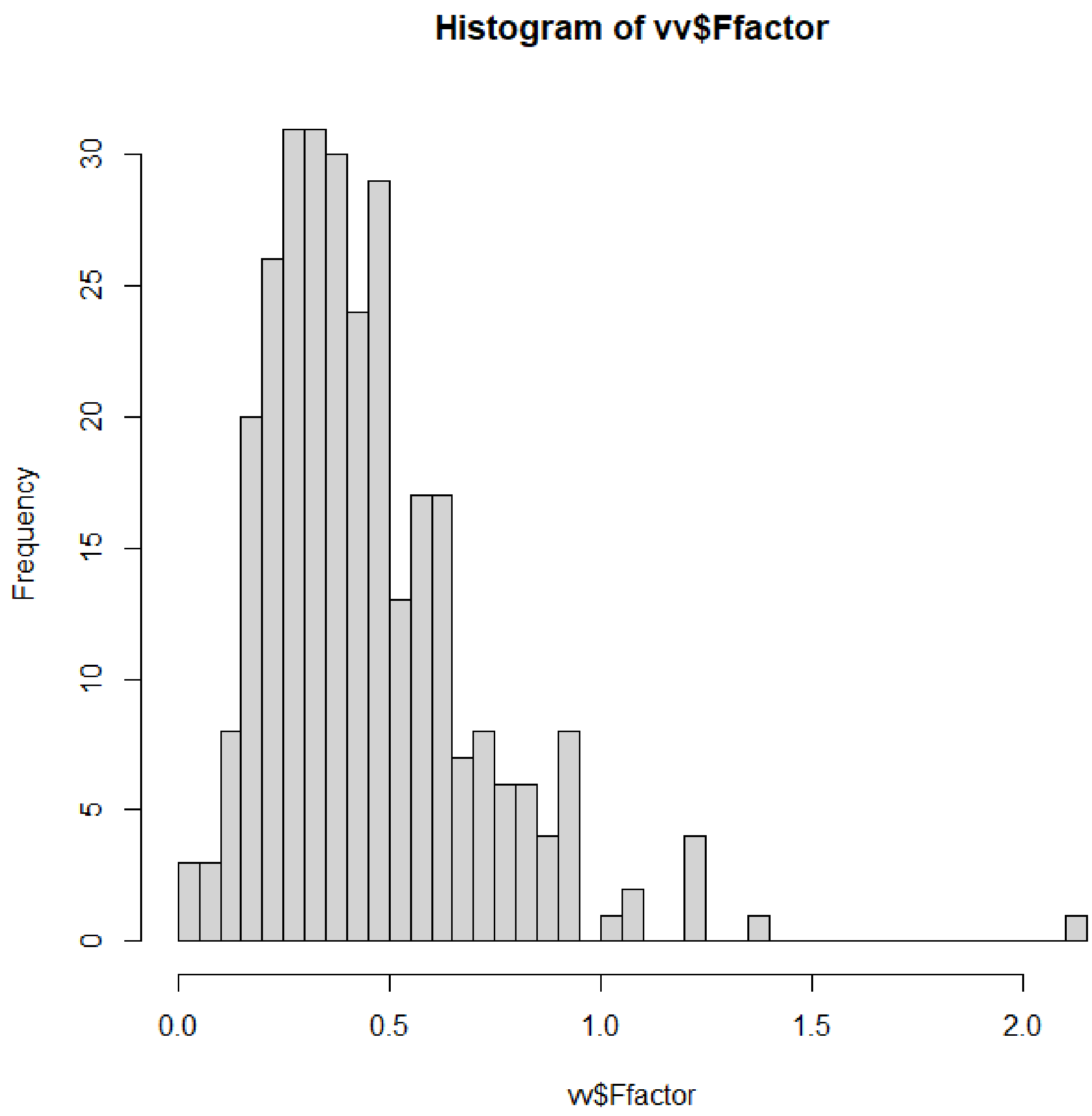

```
classThk<-rtThK
classThk[classThk$Th_K >=4 & classThk$Th_K < 7]<-3000
classThk[classThk$Th_K >=2 & classThk$Th_K < 4]<-2000
classThk[classThk$Th_K >=7 & classThk$Th_K < 8]<-2000
classThk[classThk$Th_K < 2]<-1000
classThk[classThk$Th_K >= 8 & classThk$Th_K<2000]<-1000
classThk[classThk$Th_K == 1000]<-1
classThk[classThk$Th_K == 2000]<-2
classThk[classThk$Th_K == 3000]<-3
plot(classThk)
plot(vv,add=T,col='blue')
```



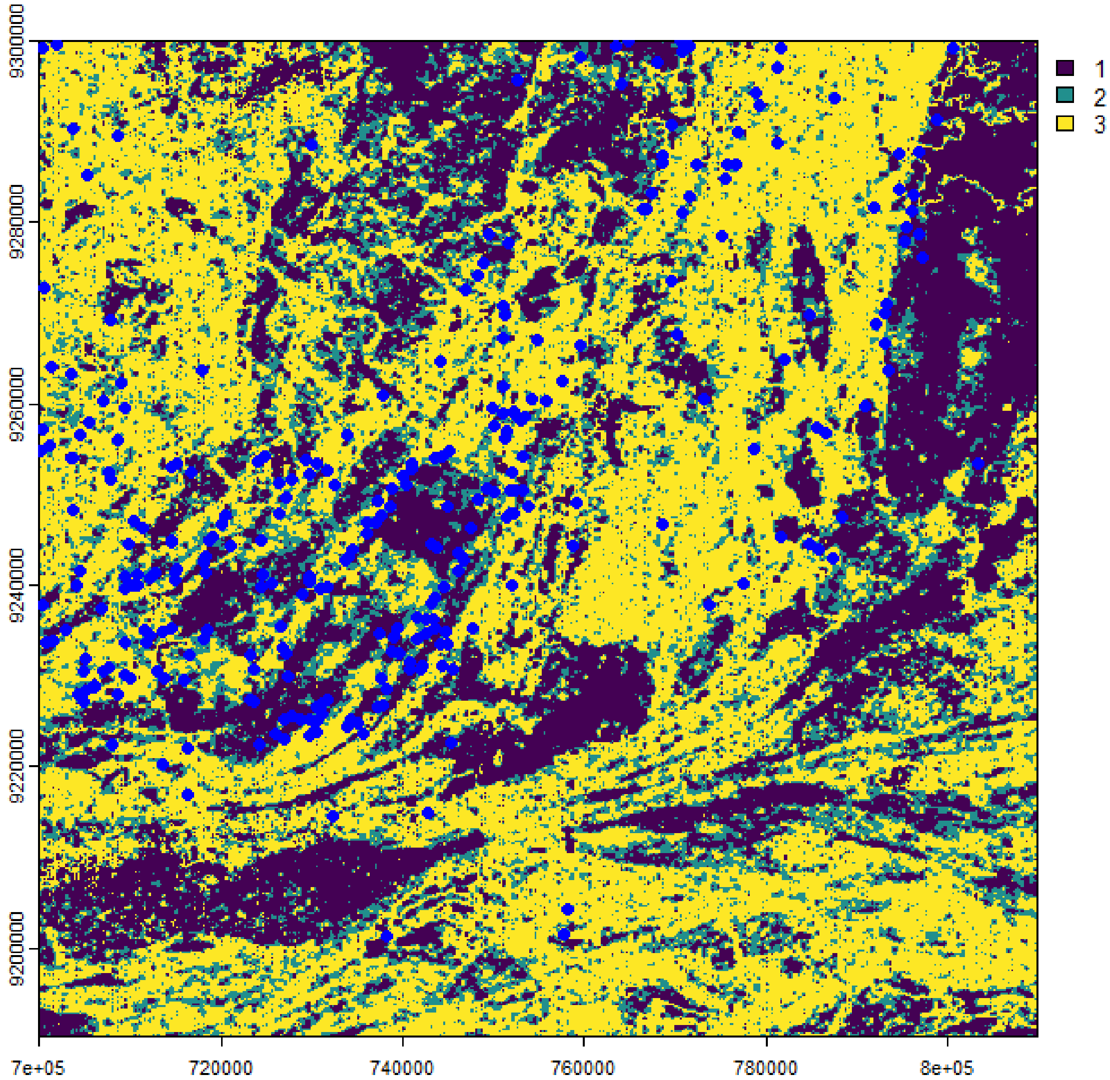

```
writeRaster(classThk, 'thkClass.tif', overwrite=TRUE)
```

F Factor

```
ff<-targ[[13]]
v <- na.omit(terra::extract(ff, train,xy=T, method = "simple", ID=F))
vv<-vect(v,geom=c('x','y'))
hist(vv$Ffactor,n=50)# 0.2 to 0.5 good,0.5 to 0.65 and 0,15 to 0.2 moderate
```



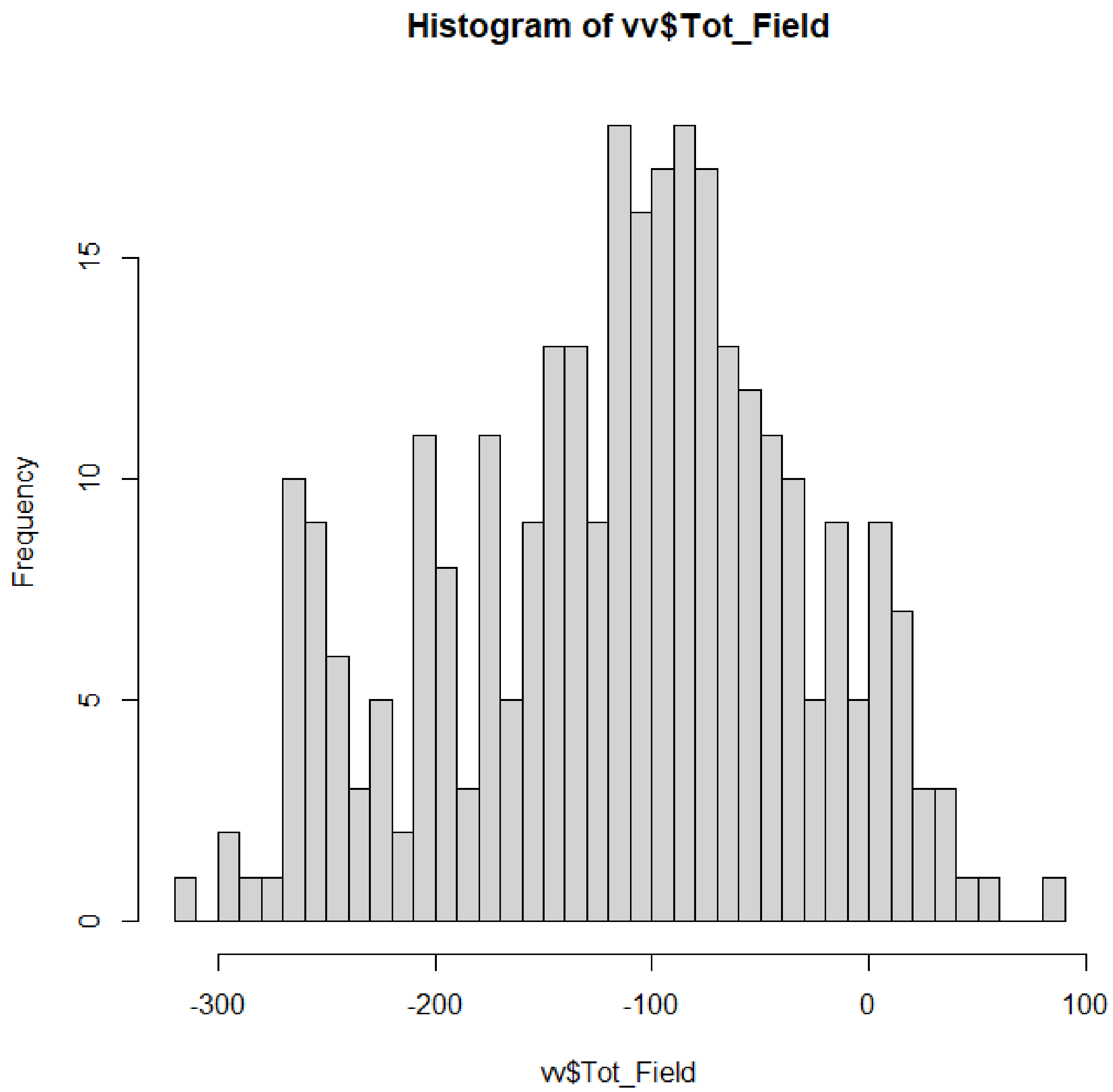

```
classFF<-ff
classFF[classFF$Ffactor >=0.2 & classFF$Ffactor < 0.5]<-3000
classFF[classFF$Ffactor >=0.5 & classFF$Ffactor < 0.65]<-2000
classFF[classFF$Ffactor >=0.15 & classFF$Ffactor < 0.2]<-2000
classFF[classFF$Ffactor < 0.15]<-1000
classFF[classFF$Ffactor >= 0.65 & classFF$Ffactor<2000]<-1000
classFF[classFF$Ffactor == 1000]<-1
classFF[classFF$Ffactor == 2000]<-2
classFF[classFF$Ffactor == 3000]<-3
plot(classFF)
plot(vv,add=T,col='blue')
```



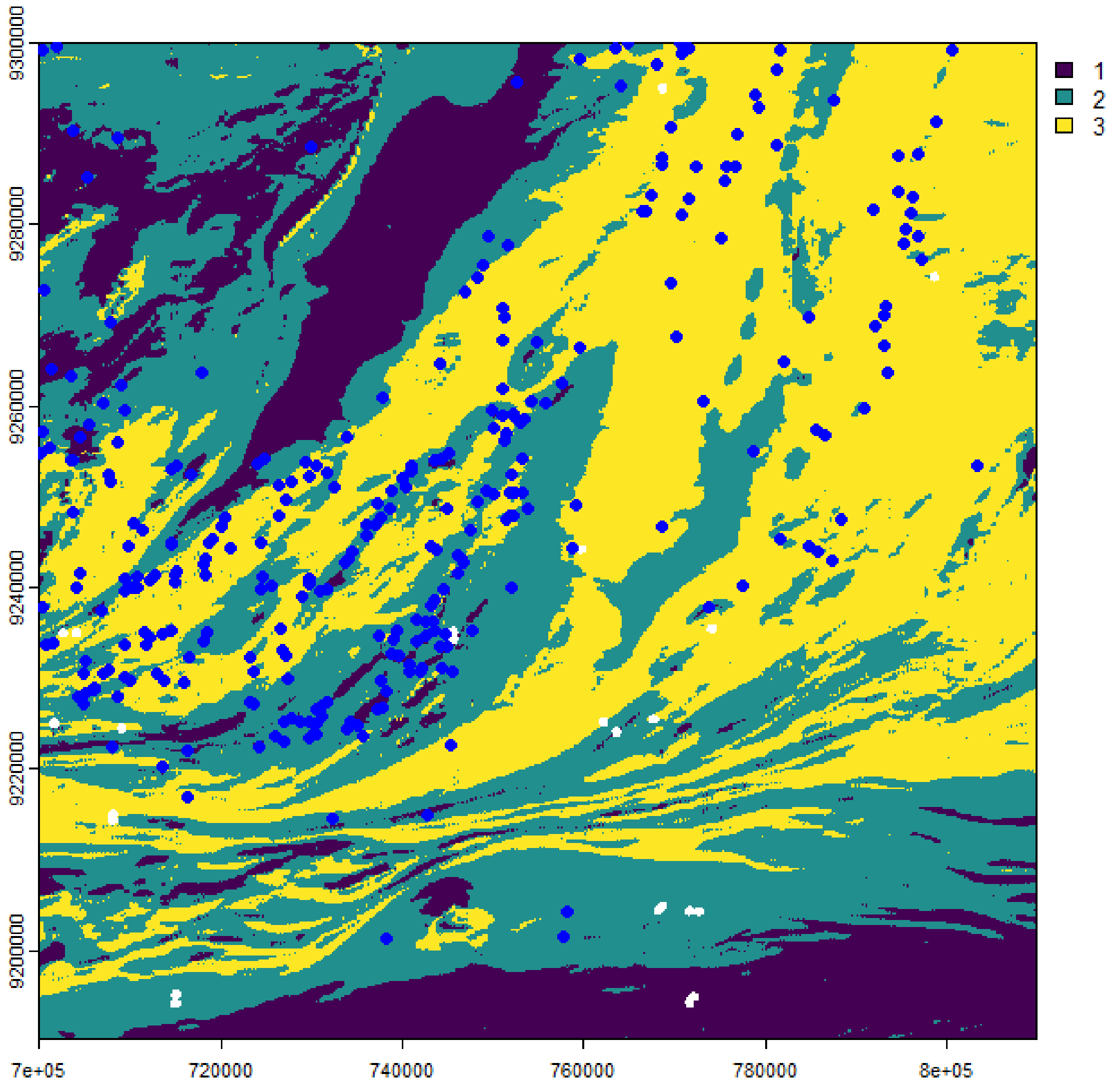

```
writeRaster(classFF, 'ffClass.tif', overwrite=TRUE)
```

Total Magnetic Field

```
mtf<-targ[[14]]
v <- na.omit(terra::extract(mtf, train,xy=T, method = "simple", ID=F))
vv<-vect(v,geom=c('x','y'))
hist(vv$Tot_Field,n=50)#-150 to -50 good,-270 to -150 and -50 to 20 moderate
```



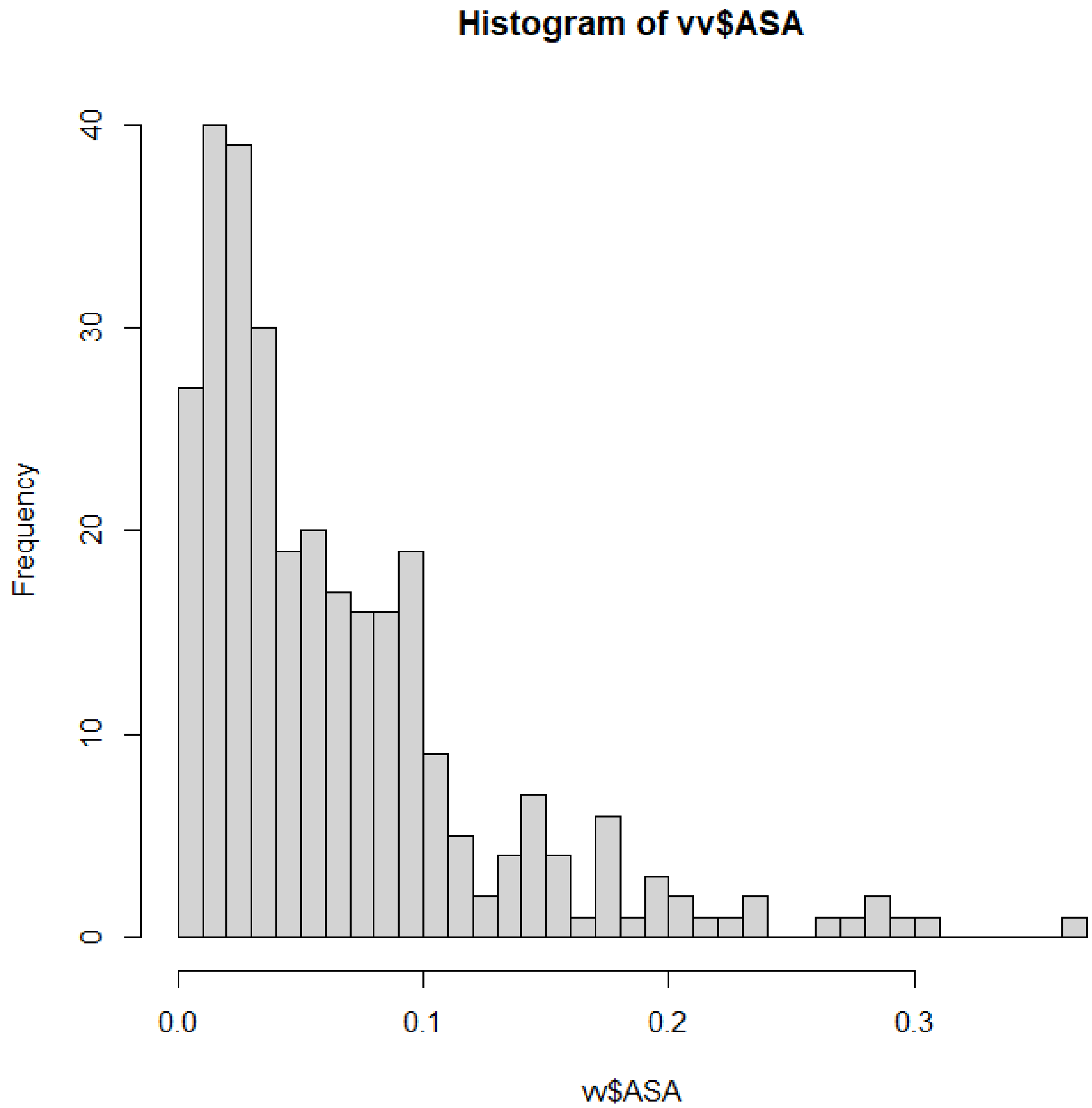

```
classTF<-mtf
classTF[classTF$Tot_Field >=-150 & classTF$Tot_Field < -50]<-3000
classTF[classTF$Tot_Field >=-270 & classTF$Tot_Field < -150]<-2000
classTF[classTF$Tot_Field >=-50 & classTF$Tot_Field < 20]<-2000
classTF[classTF$Tot_Field < -270]<-1000
classTF[classTF$Tot_Field >= 20 & classTF$Tot_Field<2000]<-1000
classTF[classTF$Tot_Field == 1000]<-1
classTF[classTF$Tot_Field == 2000]<-2
classTF[classTF$Tot_Field == 3000]<-3
plot(classTF)
plot(vv,add=T,col='blue')
```



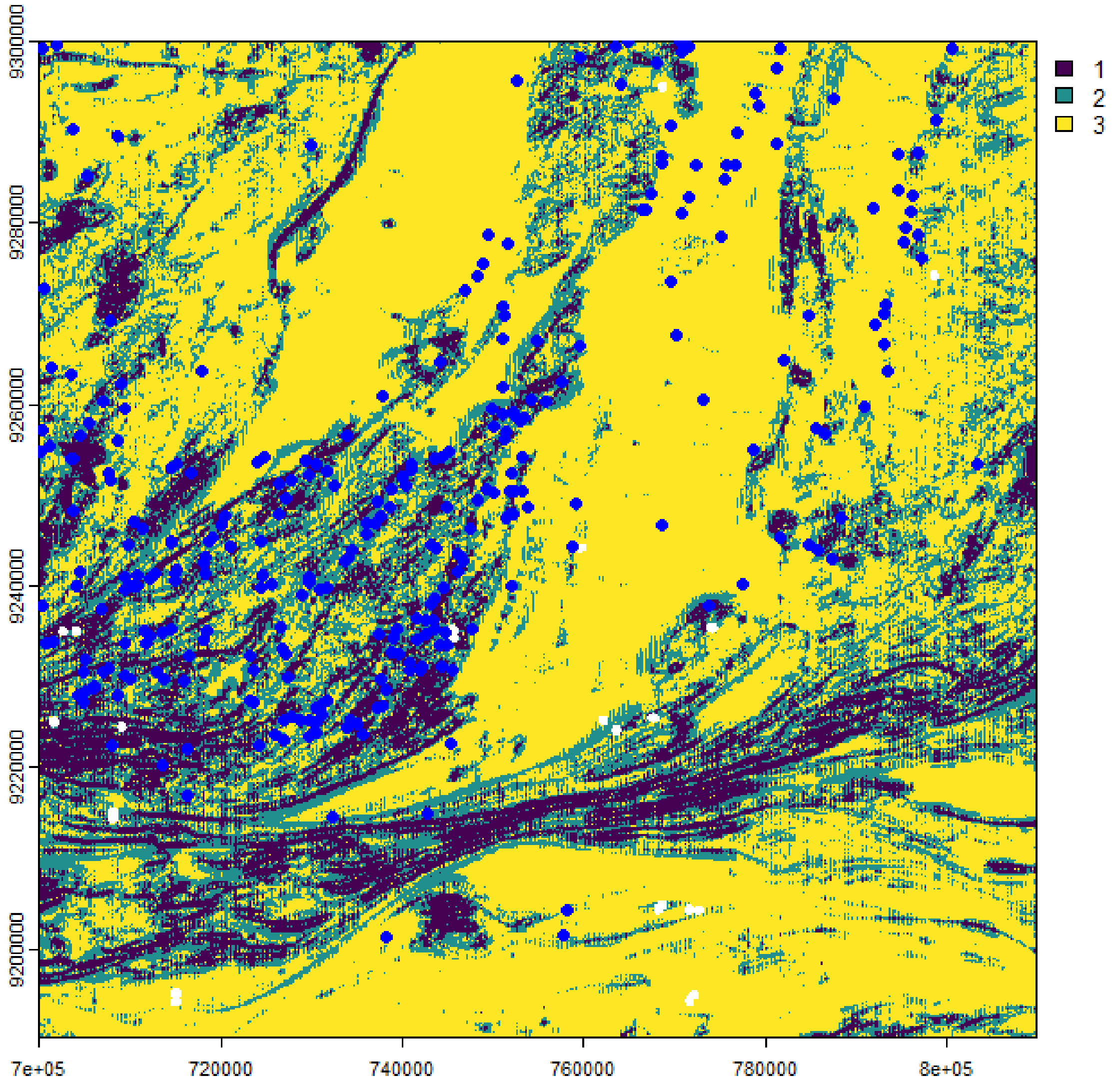

```
writeRaster(classTF, 'tmfClass.tif', overwrite=TRUE)
```

ASA

```
asa<-targ[[15]]
v <- na.omit(terra::extract(asa, train,xy=T, method = "simple", ID=F))
vv<-vect(v,geom=c('x','y'))
hist(vv$ASA,n=50)#0 to 0.04 good,0.04 to 0.1 moderate
```



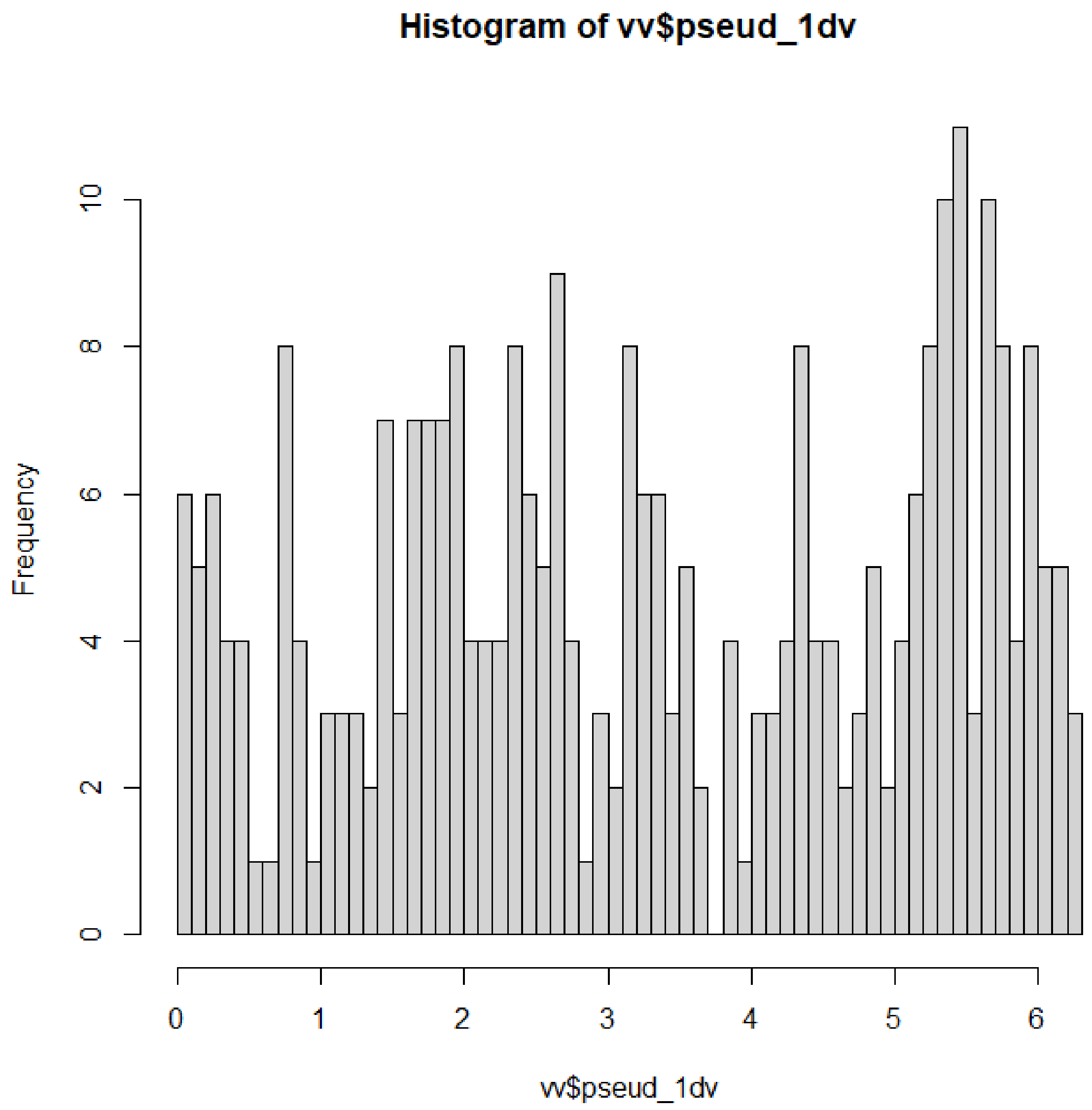

```
classASA<-asa
classASA[classASA$ASA >= 0 & classASA$ASA < 0.04]<-3000
classASA[classASA$ASA >= 0.04 & classASA$ASA < 0.1]<-2000
classASA[classASA$ASA >= 0.1 & classASA$ASA < 2000]<-1000
classASA[classASA$ASA == 1000]<-1
classASA[classASA$ASA == 2000]<-2
classASA[classASA$ASA == 3000]<-3
plot(classASA)
plot(vv,add=T,col='blue')
```




```
writeRaster(classASA, 'asaClass.tif', overwrite=TRUE)
```

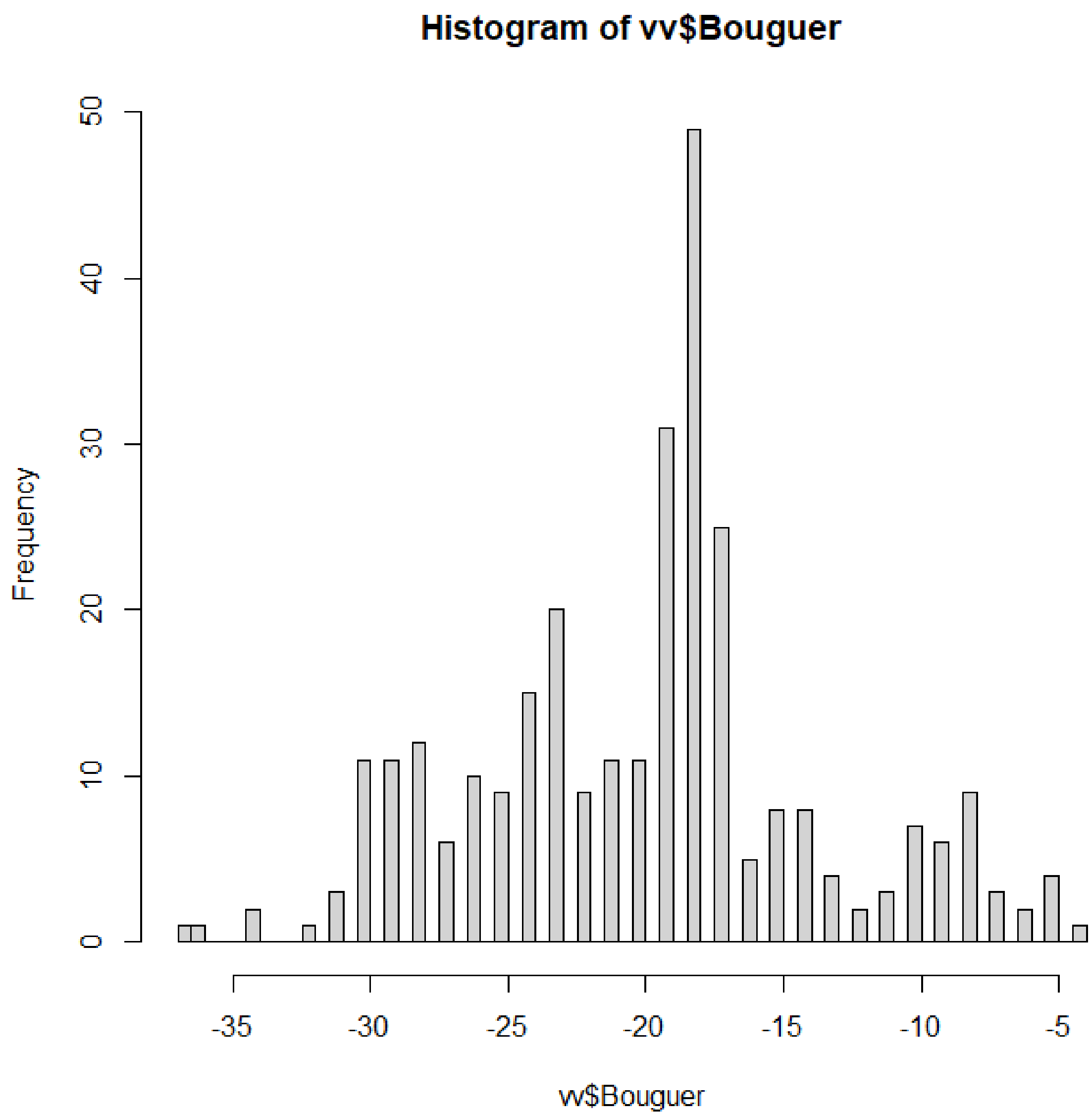
<u>**Pseudo first Derivative**</u>

```
dv1<-targ[[16]]
v <- na.omit(terra::extract(dv1, train,xy=T, method = "simple", ID=F))
vv<-vect(v,geom=c('x','y'))
hist(vv$pseud_1dv,n=50)# not resolving
```



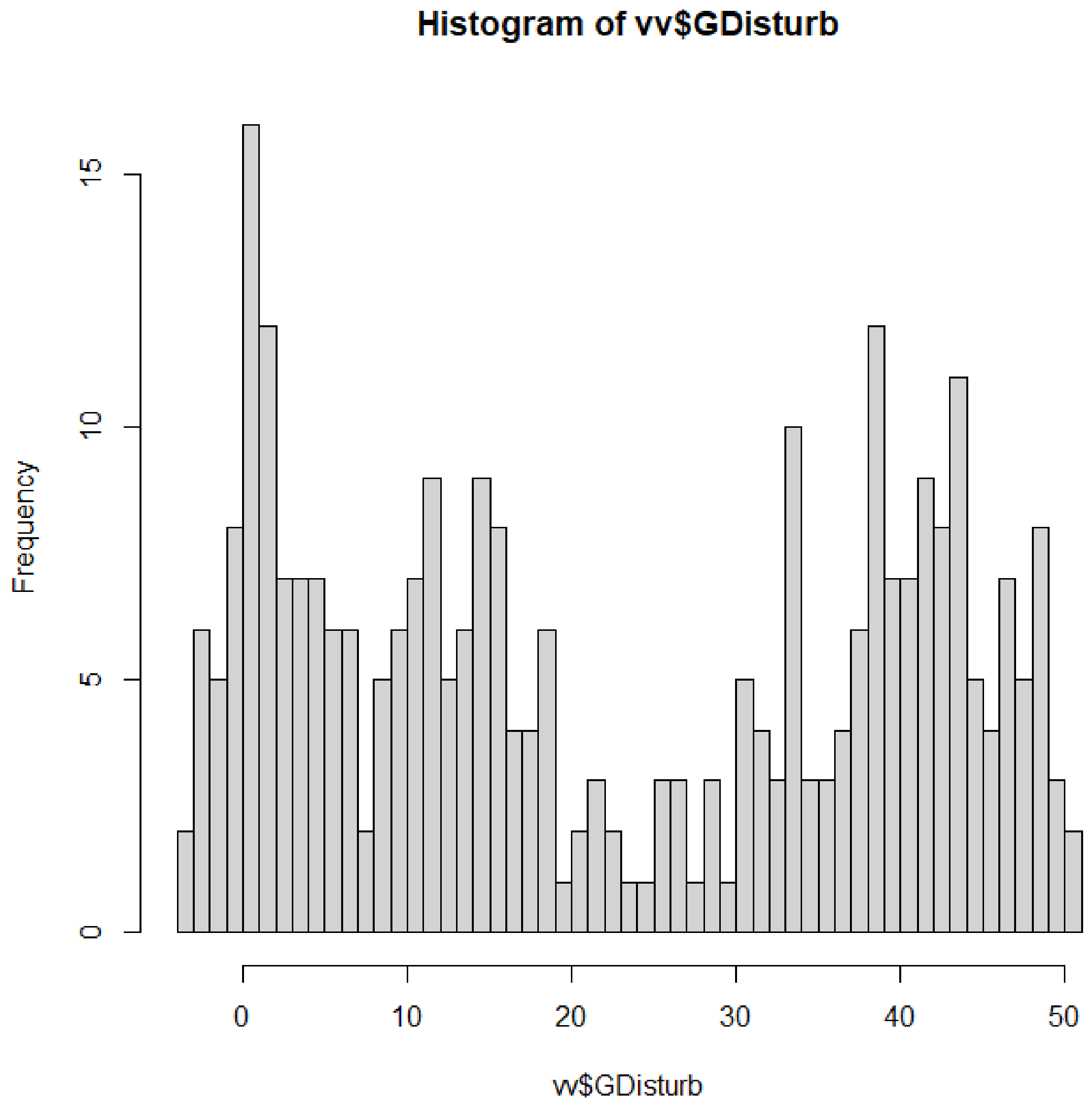
Bouguer

```
bg<-targ[[17]]
v <- na.omit(terra::extract(bg, train,xy=T, method = "simple", ID=F))
vv<-vect(v,geom=c('x','y'))
hist(vv$Bouguer,n=50)# not resolving
```



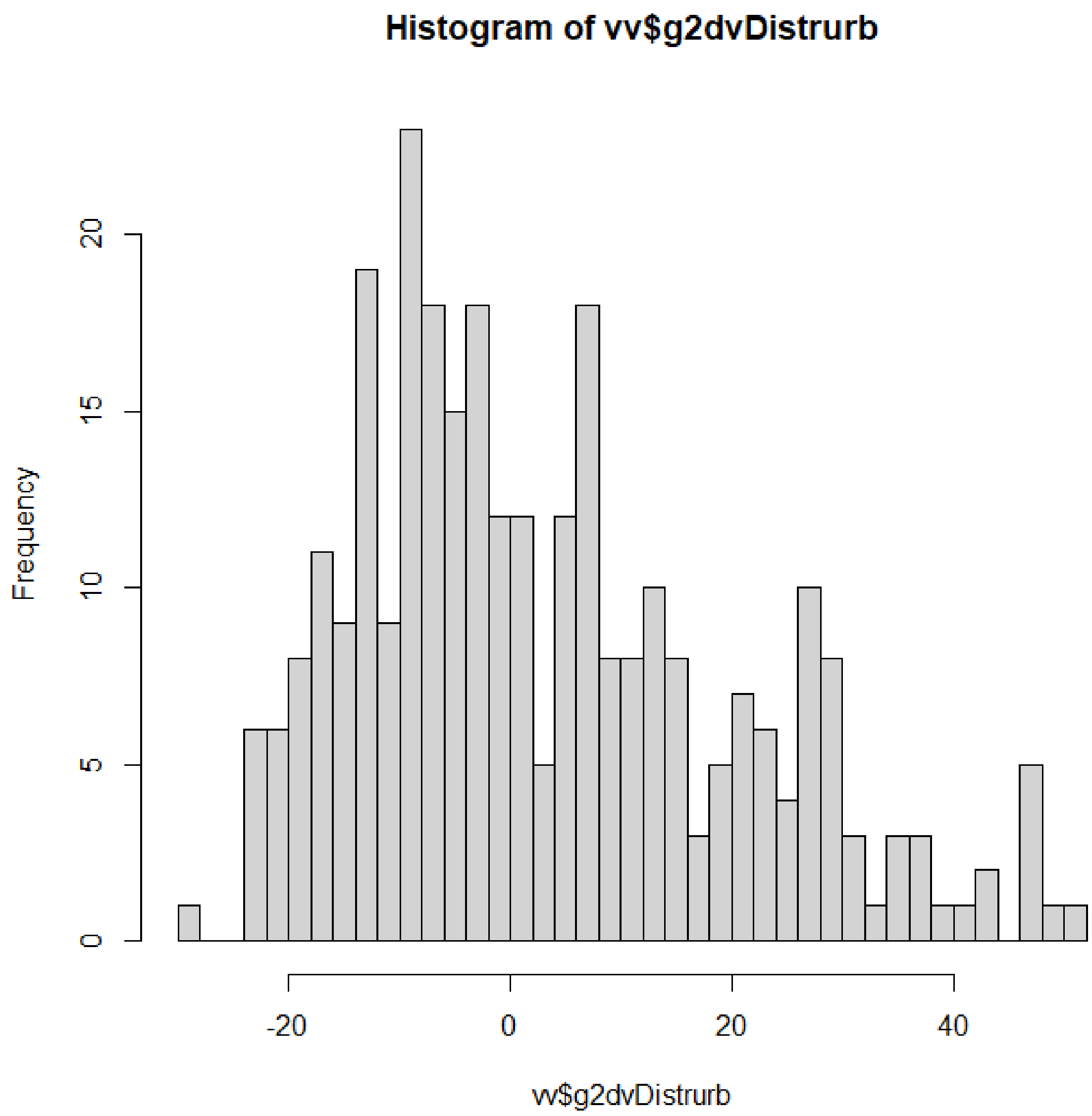
Gravity Disturbance

```
gdist<-targ[[18]]
v <- na.omit(terra::extract(gdist, train,xy=T, method = "simple", ID=F))
vv<-vect(v,geom=c('x','y'))
hist(vv$GDisturb,n=50)# not resolving
```



Second derivative on Gravity Disturbance

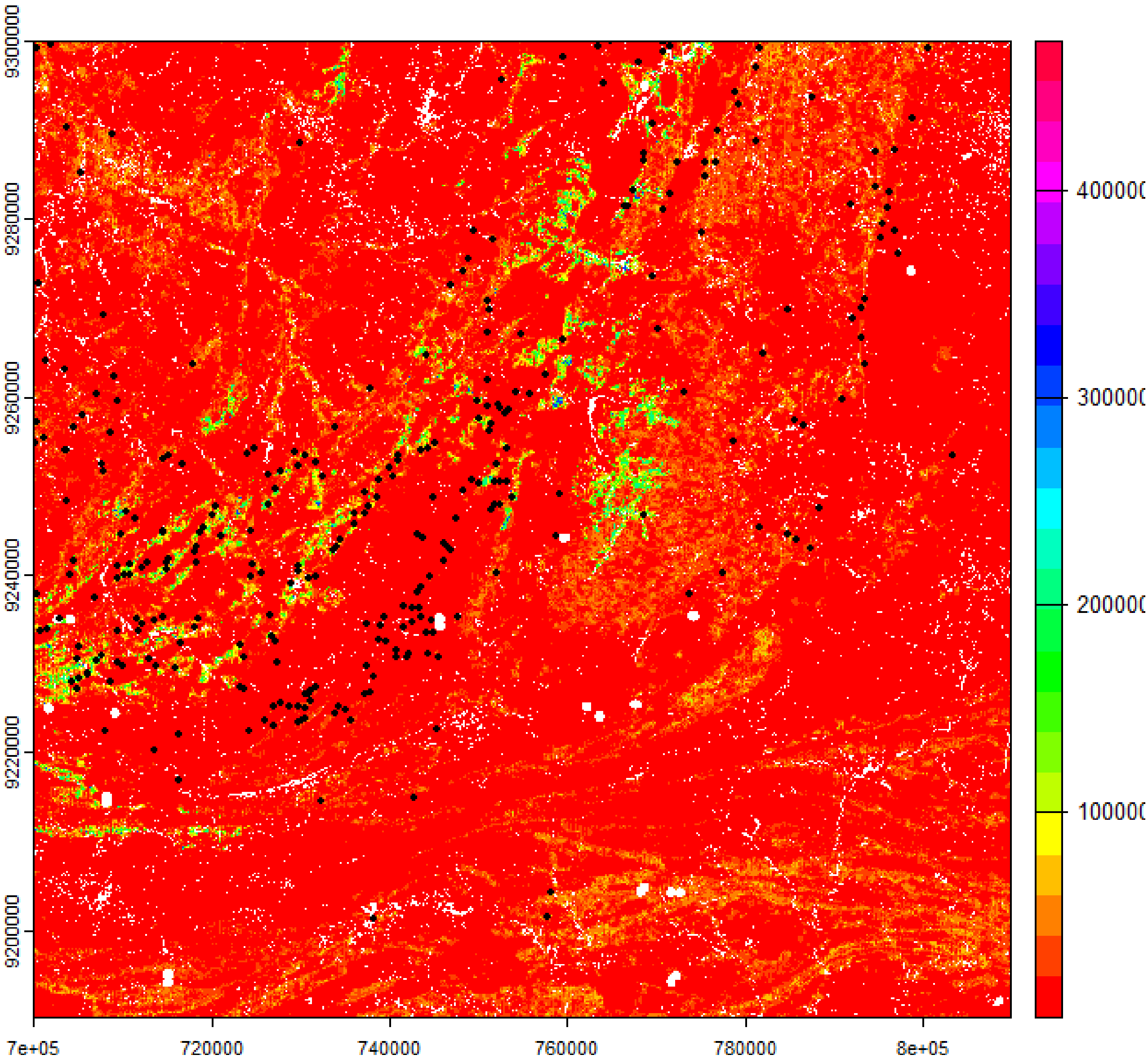
```
gdist2dv<-targ[[19]]
v <- na.omit(terra::extract(gdist2dv, train,xy=T, method = "simple", ID=F))
vv<-vect(v,geom=c('x','y'))
hist(vv$g2dvDistrurb,n=50)# not resolving
```



Preliminary Targeting Result

Generating the preliminary targeting result from the results created so far.

```
library(terra)
wd<-'C:/Users/User/Desktop/R algo/WDIR_P3'
setwd(wd)
train<-vect('tungst_occu.shp')
tgst<-rast(c('geoclass.tif','demclass.tif','crestclass.tif','kclass.tif',
              'uclass.tif','thclass.tif','tcclass.tif','uthclass.tif','ukclass.tif',
              'thkclass.tif','ffclass.tif','tmfclass.tif','asaclass.tif'))
multip<-tgst[[1]]*tgst[[2]]*tgst[[3]]*tgst[[4]]*tgst[[5]]*tgst[[6]]*tgst[[7]]*
        tgst[[8]]*tgst[[9]]*tgst[[10]]*tgst[[11]]*tgst[[2]]*tgst[[13]]
plot(multip,col=rainbow(24))
plot(train,add=T,col='black',cex=0.5)
```



This straightforward preliminary result is quite impressive, and it is important to note that:

- No occurrence data filtering, it was assumed all entries are good entries.
- No occurrences population separation or evaluation.
- No deeper evaluation of layer responses.
- No geological reasoning of results.
- No weighting of individual layers considered.

On Part 4 we will cover technics and procedures to refine and evaluate the result achieved.