

Non-uniform parameters

Koustubh Sharma & Justine Shanti Alexander



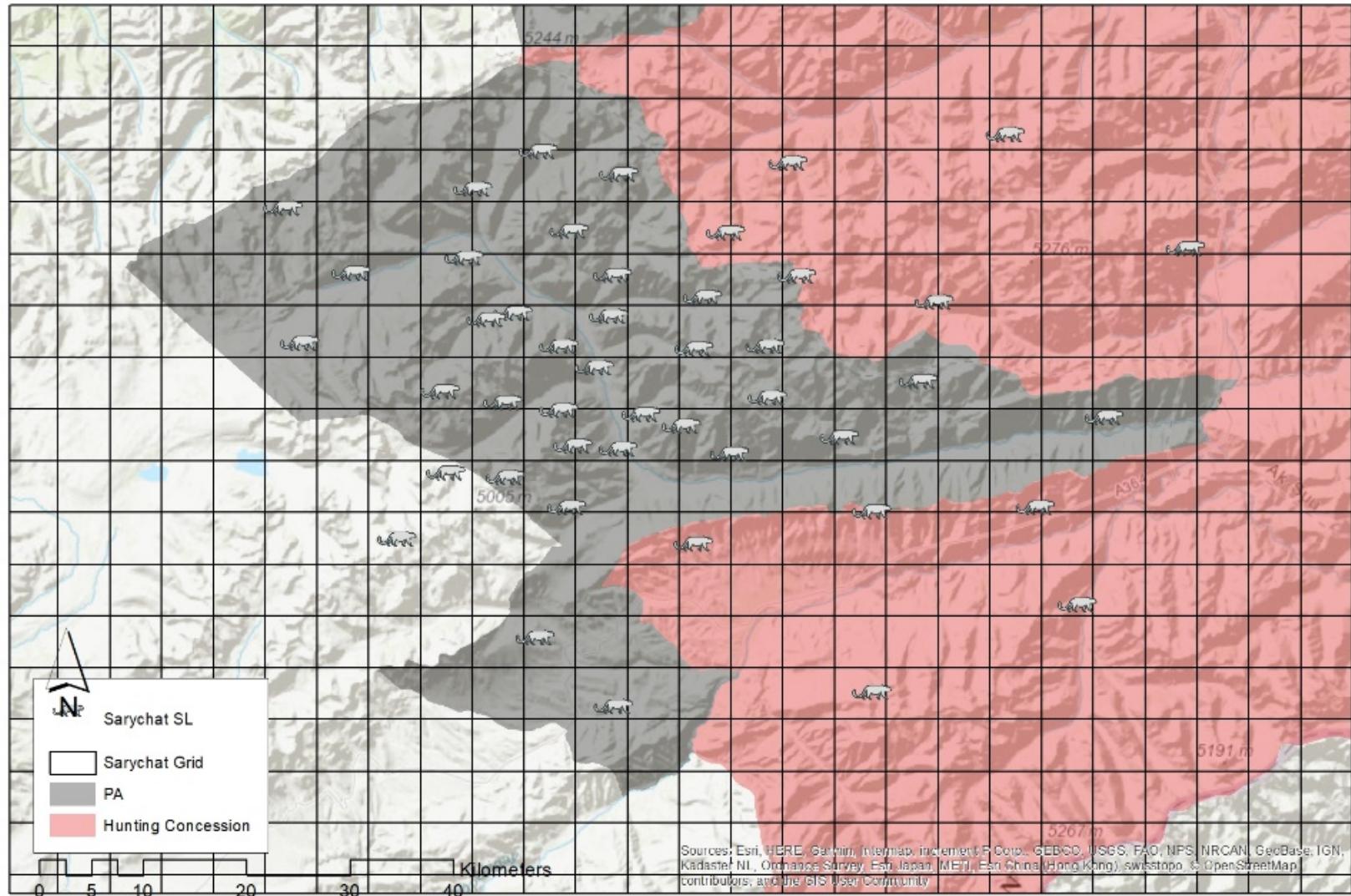
Snow
Leopard
Trust



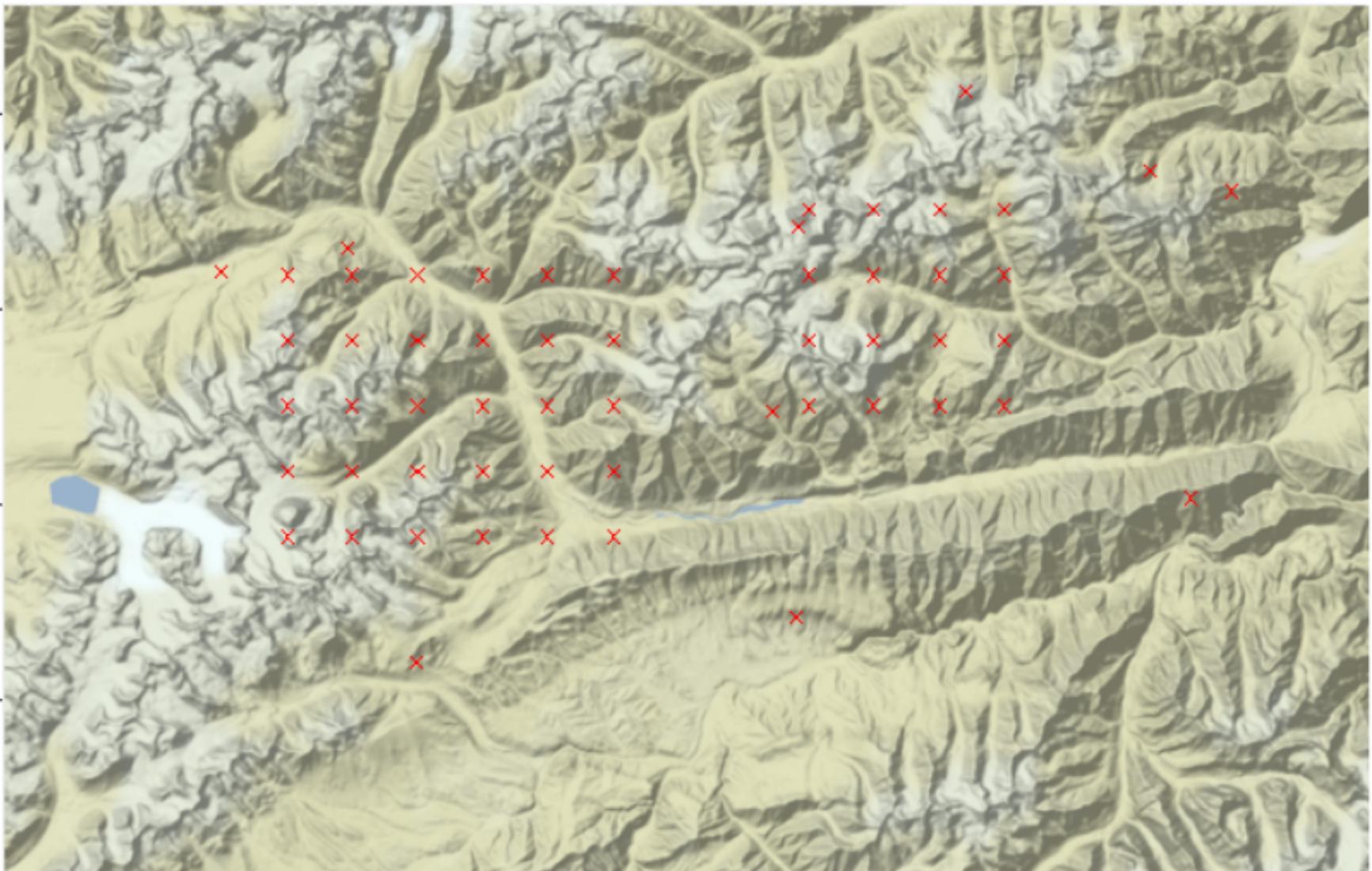
SNOW
LEOPARD
NETWORK



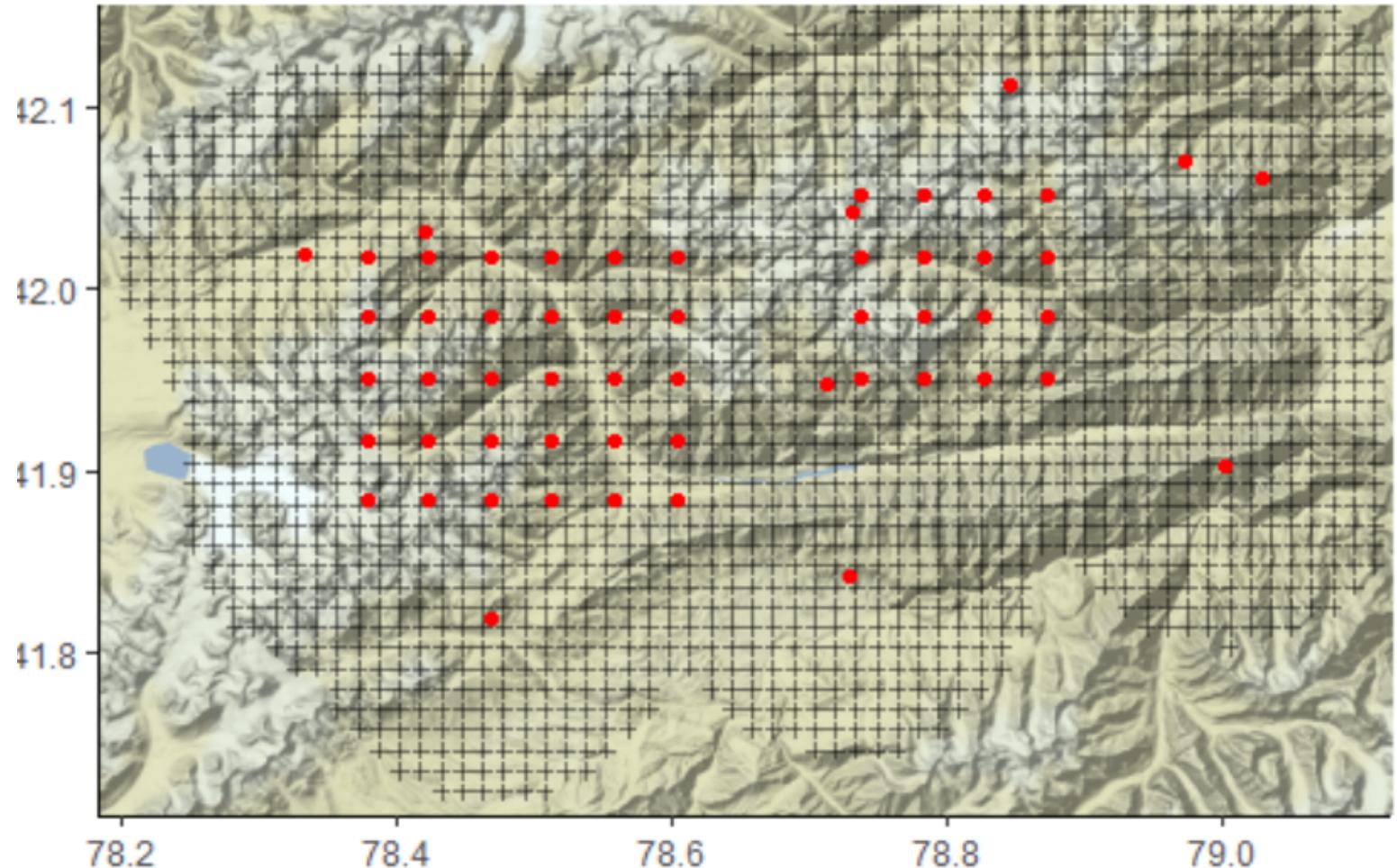
Non-uniform parameters



Setting 56 traps



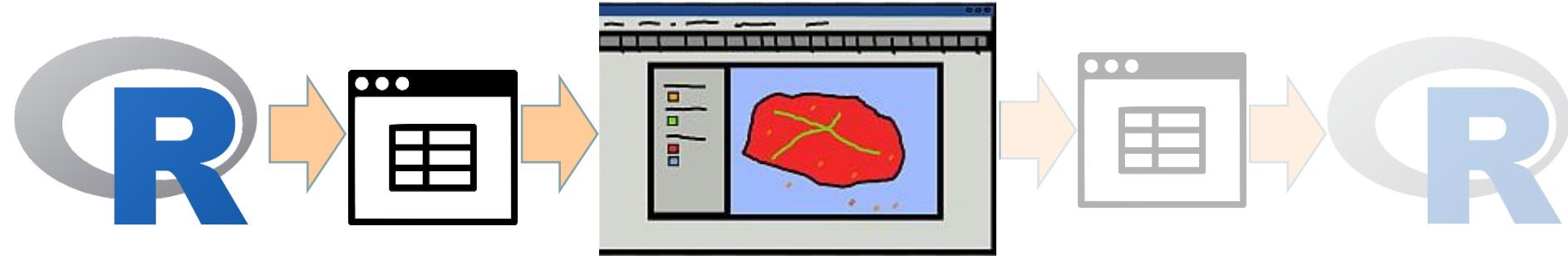
We create a mask!



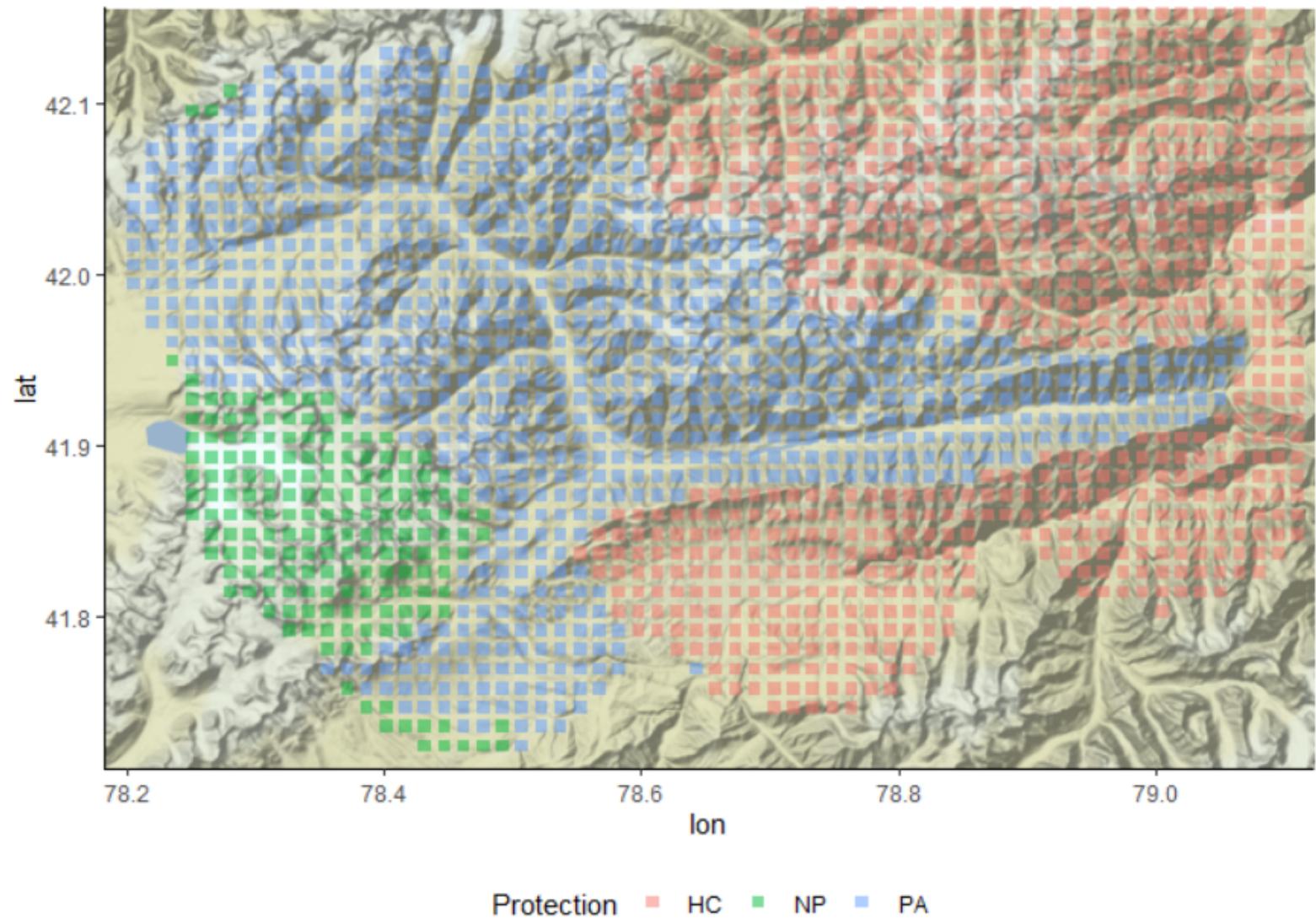
```
Sarychat.mask<-make.mask(traps(all.data.Sarychat),  
buffer = 15000, type = "trapbuffer")
```

We export it

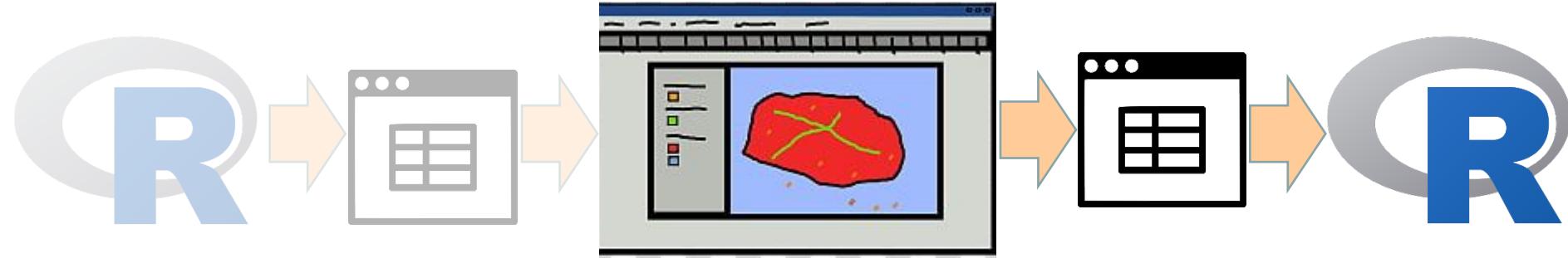
```
write.csv(Sarychat.mask, file =  
"SarychatMask.csv")
```



GIS data



We export it



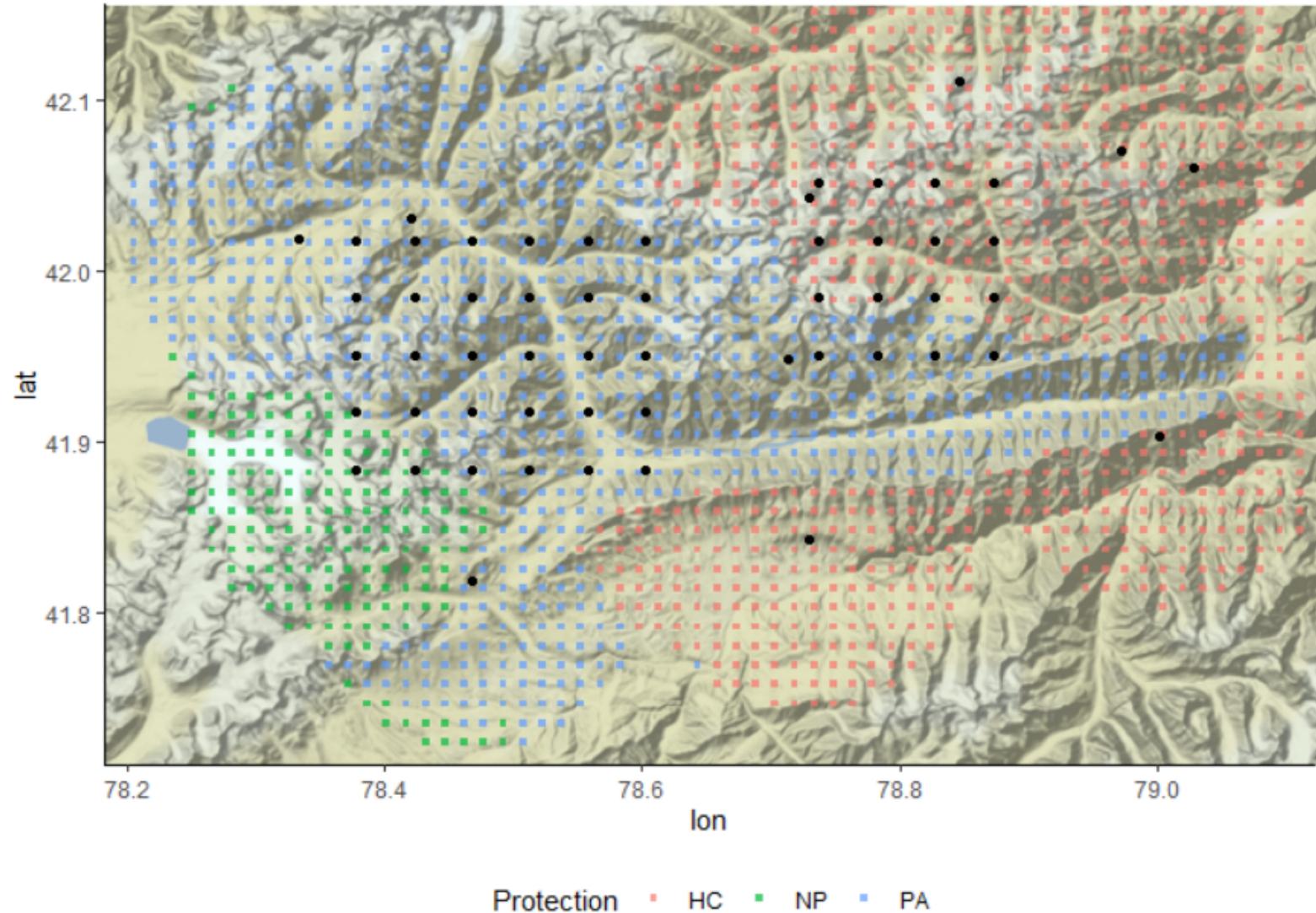
```
SarychatMask.cov<-read.mask(file =  
"SarychatMask_cov.csv", header = TRUE)
```

Check covariates

```
> head(covariates(SarychatMask.cov))  
  
> plot(SarychatMask.cov, covariate =  
"Protection", contour = FALSE, legend  
= FALSE)
```

	x	y	coords.x1	coords.x2	optional	FID	Protection	Dist_stl	std.Diststl
1	8731007	5091231	78.43197	41.72367	TRUE	0	NP	363.400	-0.66978008
2	8732685	5091231	78.44704	41.72367	TRUE	1	NP	310.838	-0.85446955
3	8734363	5091231	78.46211	41.72367	TRUE	2	NP	259.044	-1.03646054
4	8736040	5091231	78.47719	41.72367	TRUE	3	NP	225.457	-1.15447676
5	8737718	5091231	78.49226	41.72367	TRUE	4	NP	203.955	-1.23002932
6	8739396	5091231	78.50733	41.72367	TRUE	5	PA	181.172	-1.31008302
7	8727651	5092909	78.40182	41.73496	TRUE	6	NP	490.251	-0.22405781
8	8729329	5092909	78.41690	41.73496	TRUE	7	NP	435.532	-0.41632647
9	8731007	5092909	78.43197	41.73496	TRUE	8	NP	381.144	-0.60743210
10	8732685	5092909	78.44704	41.73496	TRUE	9	NP	327.702	-0.79521379
11	8734363	5092909	78.46211	41.73496	TRUE	10	PA	287.991	-0.93474816
12	8736040	5092909	78.47719	41.73496	TRUE	11	PA	254.444	-1.01740629

and add covariates to mask



Modelling D ~ Protection

```
Spiti.large.hhn<-secr.fit(all.data.Sarychat,  
model= list(D~1, lambda0~1, sigma~1),  
detectfn="HHN",  
mask=SarychatMask.cov)
```

```
Sarychat.hhn.D_PA<-secr.fit(all.data.Sarychat,  
model = list(D~Protection, lambda0~1, sigma~1),  
detectfn="HHN", mask=SarychatMask.cov)
```

Outputs

```
> coefficients(Sarychat.hhn.D_PA)
```

	beta	SE.beta	lc1	uc1
D	-9.9604176	0.50463840	-10.9494906	-8.971344
D.ProtectionNP	0.8322921	0.93374714	-0.9978186	2.662403
D.ProtectionPA	1.3477827	0.54870709	0.2723365	2.423229
lambda0	1.0191855	0.11790522	0.7880956	1.250276
sigma	8.1253627	0.04752274	8.0322199	8.218506

Outputs

```
> coefficients(Sarychat.hhn)
```

	beta	SE.beta	lcl	ucl
D	-8.958818	0.17500611	-9.301824	-8.615813
lambda0	1.013881	0.11813664	0.782337	1.245424
sigma	8.120981	0.04751196	8.027859	8.214102

$$D = \exp(\beta_D)$$

```
> coefficients(Sarychat.hhn.D_PA)
```

	beta	SE.beta	lcl	ucl
D	-9.9604176	0.50463840	-10.9494906	-8.971344
D.ProtectionNP	0.8322921	0.93374714	-0.9978186	2.662403
D.ProtectionPA	1.3477827	0.54870709	0.2723365	2.423229
lambda0	1.0191855	0.11790522	0.7880956	1.250276
sigma	8.1253627	0.04752274	8.0322199	8.218506

$$D = \exp(\beta + \beta_{PA} + \beta_{NP})$$

Predict doesn't work now!

```
> predict(Sarychat.hhn.D_PA)
```

	link	estimate	SE.estimate	1cI	ucI
D	log	4.73016251e-01	1.756005e-05	1.269973e-04	
lambda0	log	2.770937e+00	3.278467e-01	2.199204e+00	3.491305e+00
sigma	log	3.379093e+03	1.606745e+02	3.078568e+03	3.708956e+03

$$D = \exp(\beta + \beta_{PA} \cdot PA + \beta_{NP} \cdot NP)$$

$$D_{PA} = \exp(-9.96 + (1.34 \times 1))$$

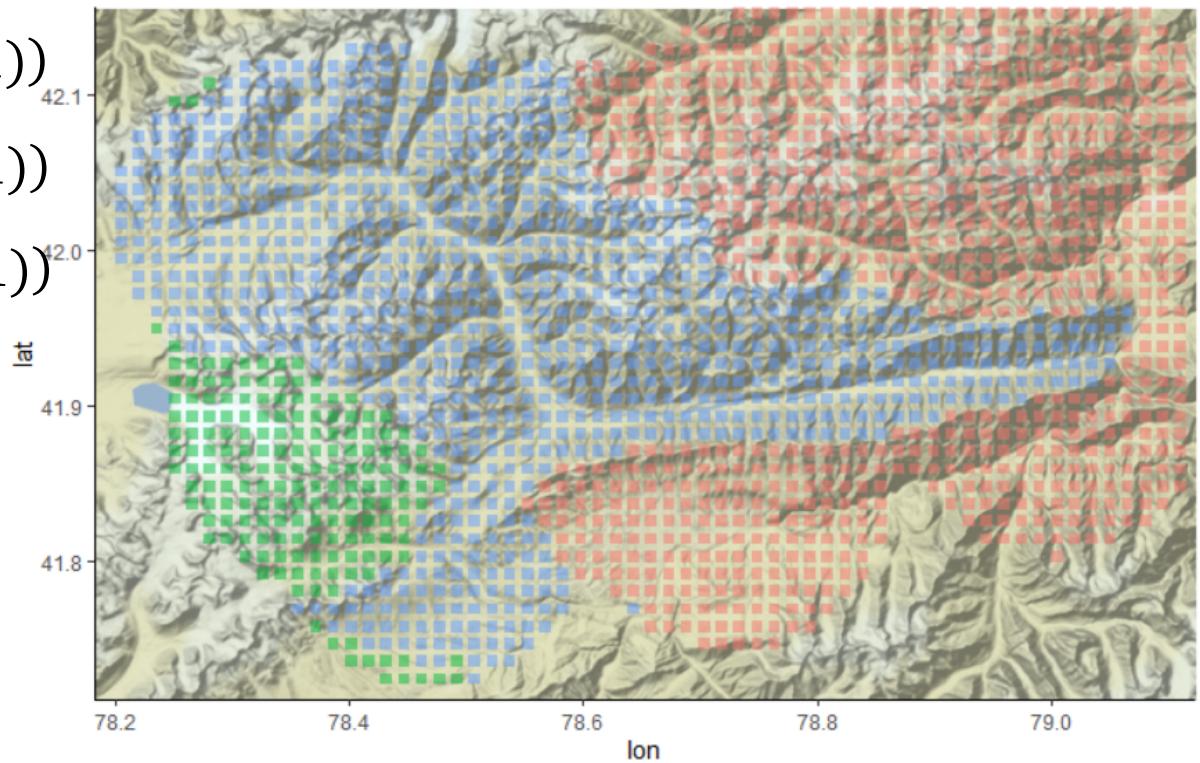
$$D_{HC} = \exp(-9.96 + (0.83 \times 1))$$

$$D_{NP} = \exp(-9.96 + (0.83 \times 1))$$

$$D_{PA} = 1.83 \text{ SL km}^{-2}$$

$$D_{HC} = 0.47 \text{ SL km}^{-2}$$

$$D_{NP} = 1.09 \text{ SL km}^{-2}$$



But overall density?

D = Abundance/Area

= region.N(Sarychat.hhn.D_PA) / Mask Area

= 62.34579 / 5591

= 1.25 (95%CI: 1.02-1.59)

per 100 sq km

Which one to use?

AIC(Sarychat.hhn, Sarychat.hhn.D_PA)

```
> AIC(Sarychat.hhn, Sarychat.hhn.D_PA)
      model      detectfn npar   logLik     AIC    AICc dAICc AICCw
Sarychat.hhn.D_PA D~Protection Lambda0~1 sigma~1 hazard halfnormal  5 -271.5130 553.026 555.169 0.000 0.8229
Sarychat.hhn          D~1 Lambda0~1 sigma~1 hazard halfnormal  3 -275.7207 557.441 558.241 3.072 0.1771
```

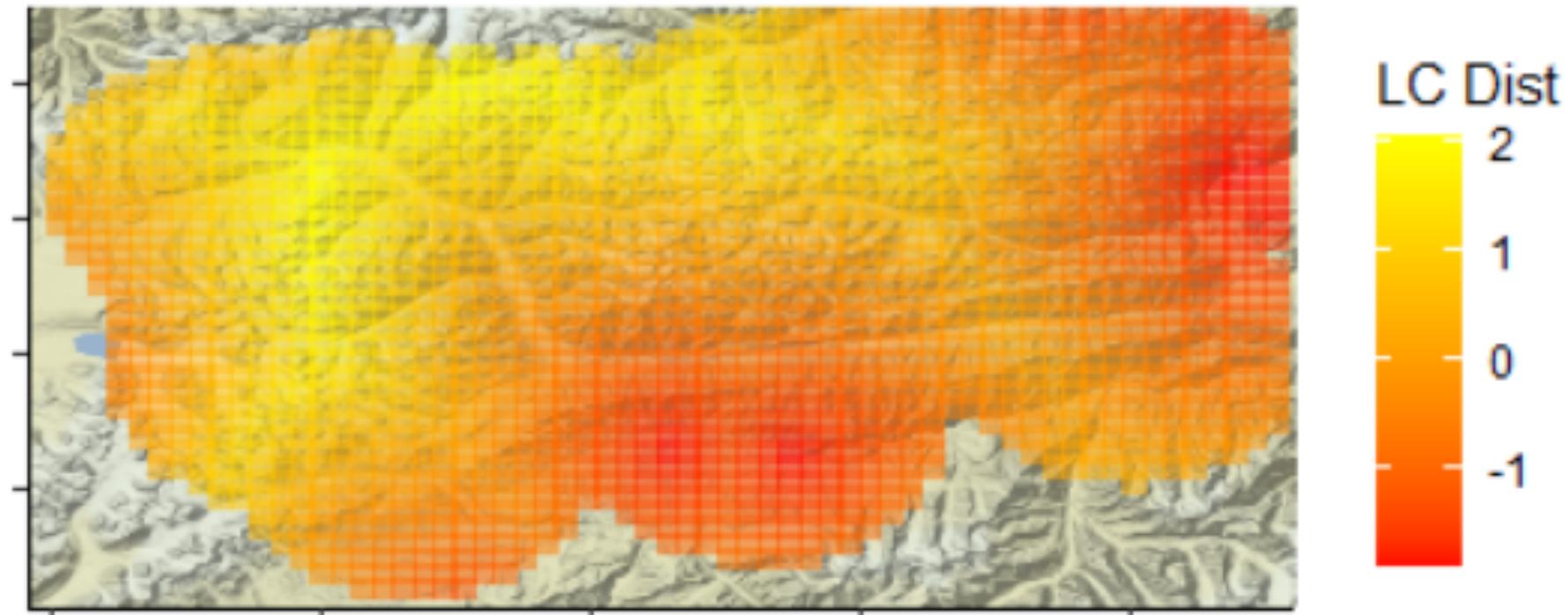
```
> region.N(Sarychat.hhn)
```

	estimate	SE.estimate	lcl	ucl	n
E.N	71.89936	12.679790	51.02224	101.3189	34
R.N	75.28396	9.427497	60.53800	98.2236	34

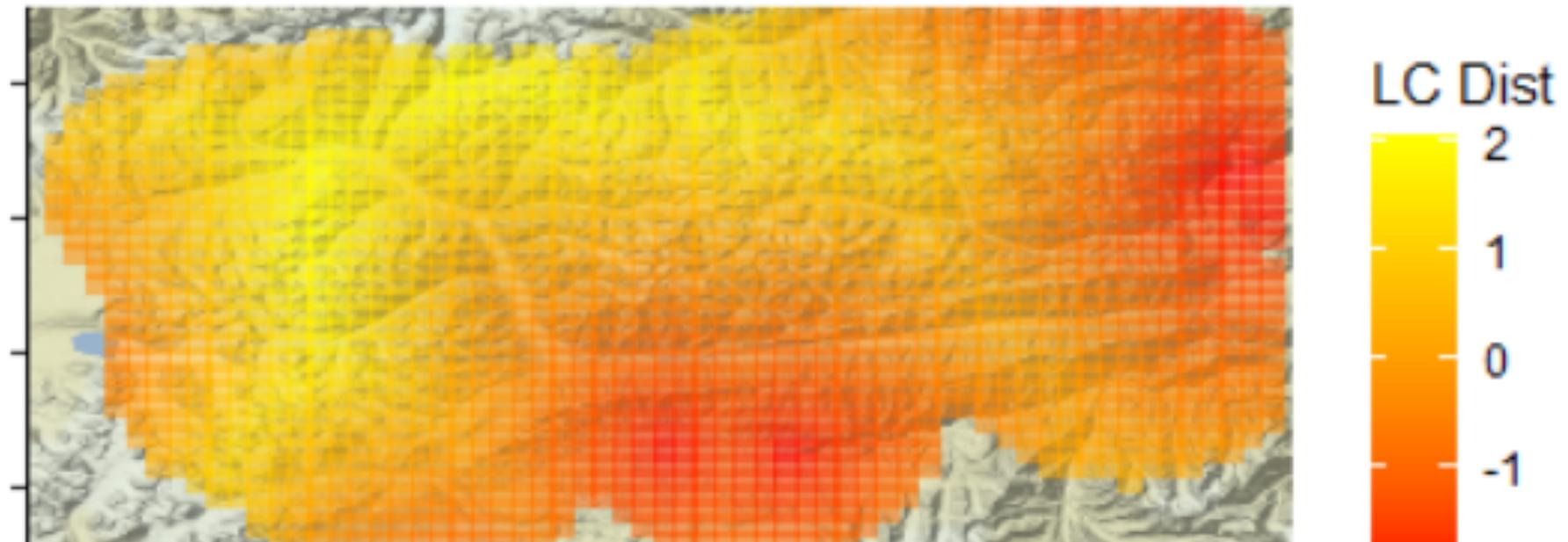
```
> region.N(Sarychat.hhn.D_PA)
```

	estimate	SE.estimate	lcl	ucl	n
E.N	62.34579	11.230434	43.92395	88.49381	34
R.N	69.63769	7.986042	57.09381	88.99503	34

What if we have more covariates?



We can test several relationships



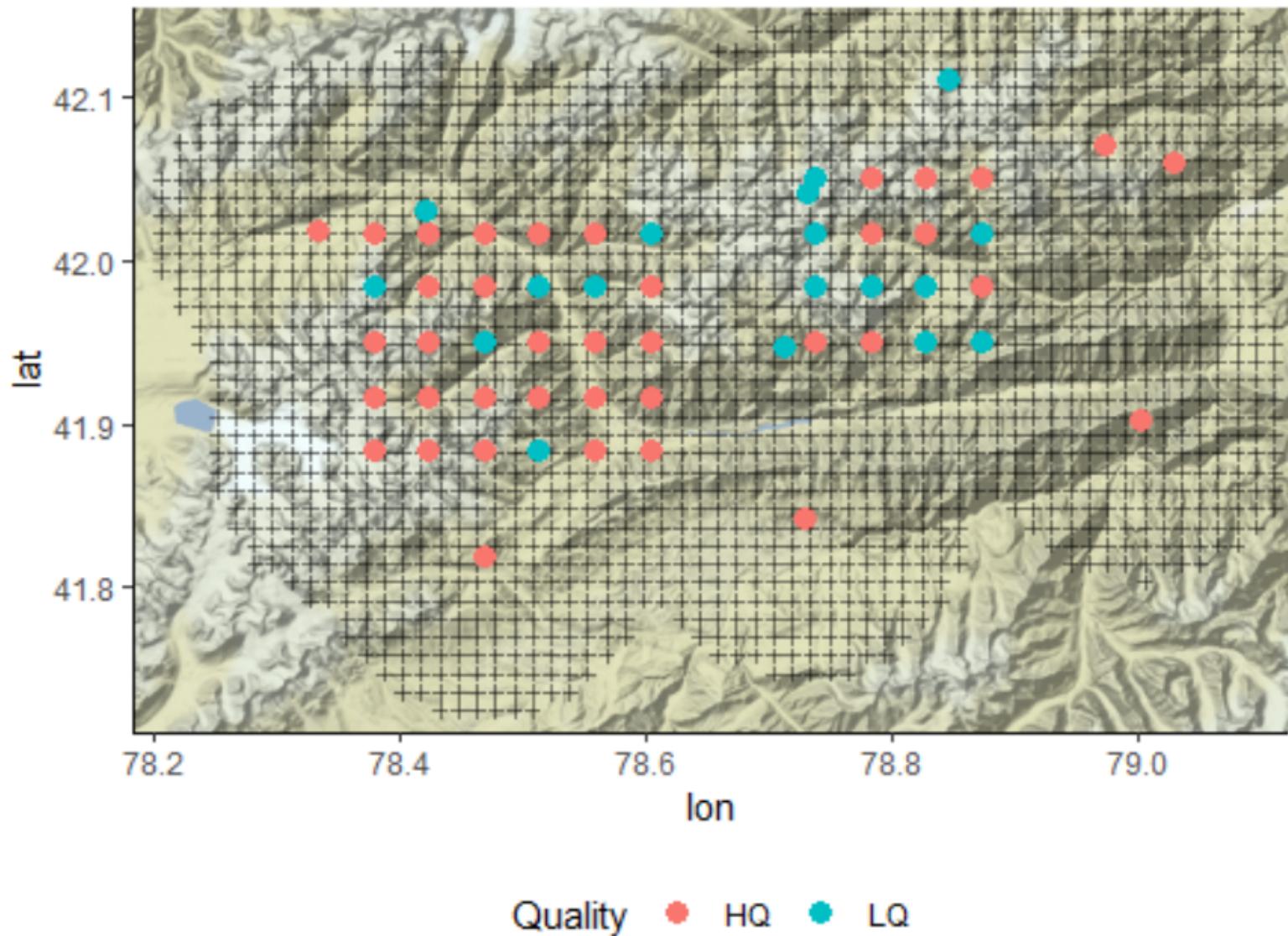
D ~ 1

D ~ Protection

D ~ LC Dist

D ~ LC Dist + Protection

Variable camera quality



Capture data

	Session	ID	Occasion	trapID
1	1	2	1	C41
2	1	2	1	C40
3	1	2	1	C40
4	1	2	1	C44
5	1	2	1	C41
6	1	2	1	C41
7	1	4	1	C49
8	1	5	1	C16
9	1	5	1	C16
10	1	5	1	C17
11	1	5	1	C16
12	1	5	1	C28
13	1	7	1	C16

Trap data

	Session	ID	Occasion	trapID
1	1	2	1	C41
2	1	2	1	C40
3	1	2	1	C40
4	1	2	1	C44
5	1	2	1	C41
6	1	2	1	C41
7	1	4	1	C49
8	1	5	1	C16
9	1	5	1	C16
10	1	5	1	C17
11	1	5	1	C16
12	1	5	1	C28
13	1	7	1	C16

```
> table(Sarychat.captured.Q$Quality)
```

HQ LQ
132 7
38 18

	x	y	CamID	Quality
1	8725000	5115000	C1	HQ
2	8730000	5115000	C2	HQ
3	8735000	5115000	C3	HQ
4	8740000	5115000	C4	LQ
5	8745000	5115000	C5	HQ
6	8750000	5115000	C6	HQ
7	8725000	5120000	C7	HQ
8	8730000	5120000	C8	HQ
9	8735000	5120000	C9	HQ
10	8740000	5120000	C10	HQ
11	8745000	5120000	C11	HQ
12	8750000	5120000	C12	HQ
13	8725000	5125000	C13	HQ

Modelling $\lambda_0 \sim$ Quality

```
Spiti.large.hhn<-secr.fit(all.data.Sarychat,
model= list(D~1, lambda0~1, sigma~1),
detectfn="HHN",
mask=SarychatMask.cov)
```

```
Sarychat.hhn.D_PA<-secr.fit(all.data.Sarychat,
model = list(D~Protection, lambda0~1, sigma~1),
detectfn="HHN", mask=SarychatMask.cov)
```

```
Sarychat.hhn.10_Q<-secr.fit(all.data.Sarychat.q,
model = list(D~1, lambda0~Quality, sigma~1),
detectfn="HHN", mask=SarychatMask.cov)
```

Output

```
> coefficients(Sarychat.hhn.10_Q)
```

	beta	SE.beta	lcl	ucl
D	-8.891911	0.17960240	-9.243925	-8.539897
lambda0	1.085988	0.13821278	0.815096	1.356880
lambda0.QualityLQ	-2.059148	0.40730102	-2.857443	-1.260852
sigma	8.116807	0.05389828	8.011168	8.222445

Modelling D ~ Protection λ_0 ~ Quality

```
Spiti.large.hhn<-secr.fit(all.data.Sarychat,  
model= list(D~1, lambda0~1, sigma~1),  
detectfn="HHN",  
mask=SarychatMask.cov)
```

```
Sarychat.hhn.D_PA<-secr.fit(all.data.Sarychat,  
model = list(D~Protection, lambda0~1, sigma~1),  
detectfn="HHN", mask=SarychatMask.cov)
```

```
Sarychat.hhn.10_Q<-secr.fit(all.data.Sarychat.q, model =  
list(D~1, lambda0~Quality, sigma~1),  
detectfn="HHN", mask=SarychatMask.cov)
```

```
Sarychat.hhn.D_PA.10_Q<-secr.fit(all.data.Sarychat.q,  
model = list(D~Protection, lambda0~Quality, sigma~1),  
detectfn="HHN", mask=SarychatMask.cov)
```

Output

```
> coefficients(Sarychat.hhn.10_Q)
```

	beta	SE.beta	lcl	ucl
D	-8.891911	0.17960240	-9.243925	-8.539897
lambda0	1.085988	0.13821278	0.815096	1.356880
lambda0.QualityLQ	-2.059148	0.40730102	-2.857443	-1.260852
sigma	8.116807	0.05389828	8.011168	8.222445

```
> coefficients(Sarychat.hhn.D_PA.10_Q)
```

	beta	SE.beta	lcl	ucl
D	-9.8045511	0.50085706	-10.7862129	-8.822889
D.ProtectionNP	0.6190232	0.94323543	-1.2296843	2.467731
D.ProtectionPA	1.2364823	0.54513686	0.1680337	2.304931
lambda0	1.0866412	0.13760222	0.8169458	1.356337
lambda0.QualityLQ	-2.0296356	0.40573677	-2.8248650	-1.234406
sigma	8.1235905	0.05393249	8.0178848	8.229296

Testing hypotheses!

```
> AIC(Sarychat.hhn.D_PA.I0_Q, Sarychat.hhn.I0_Q, Sarychat.hhn)
```

Model Name	model	npar	logLik	AIC	AICc	dAICc	AICcwt
Sarychat.hhn.D_PA.I0_Q	D~Protection lambda0~Quality sigma~1	6	-220.356	452.712	455.943	0	0.6455
Sarychat.hhn.I0_Q	D~1 lambda0~Quality sigma~1	4	-223.857	455.713	457.142	1.199	0.3545
Sarychat.hhn	D~1 lambda0~1 sigma~1	3	-275.721	557.441	558.241	102.298	0

Questions...

