Lab3-Mayker Elizondo

Informe laborario 3

En siguiente es un informe del capítulo de interpolación que se ubica en el siguiente link:

https://rspatial.org/raster/analysis/4-interpolation.html

En el presente informe, en una primera parte se hace un resumen general, posteriormente se hace una descripción mas general y ser termina con una conclusión.

En informe trata de dos sets de datos, el primero sobre la temperatura del estado de California y el segundo sobre la contaminación en el aire también en el estado de California. La correlación espacial es un problema para muchas pruebas estadísticas, sin embargo, es útil cuando se quiere hacer interpolación espacial y precisamente es el tema principal del presente informe.

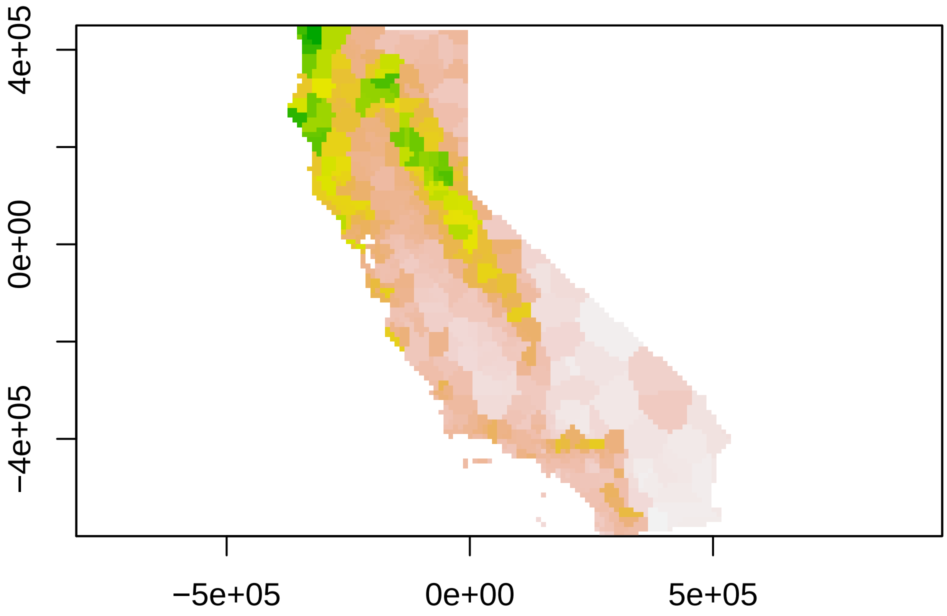
Temperatura en el estado de California

Inicialmente se construyen interpolaciones con cuatro modelos, el primero es el modelo nulo, el cual cosiste en el promedio, el siguiente fue el de "polígonos de proximidad" con cinco vecinos, es importante destacar que este modelo le da el mismo peso a todos los las observaciones, precisamente esa es la diferencia con el modelo de distancia inversa ponderada (IDW) por sus siglas en ingles el cual da un mayor peso a las observaciones más cercanas.

El modelo "nulo" se comparó con el modelo de polígonos de proximidad y el IDW; en ambos casos el modelo "nulo" da el mayor RMSE, es decir tanto el modelo de polígonos de proximidad y el de IDW dan un mejor ajuste que el modelo nulo, es decir el promedio.

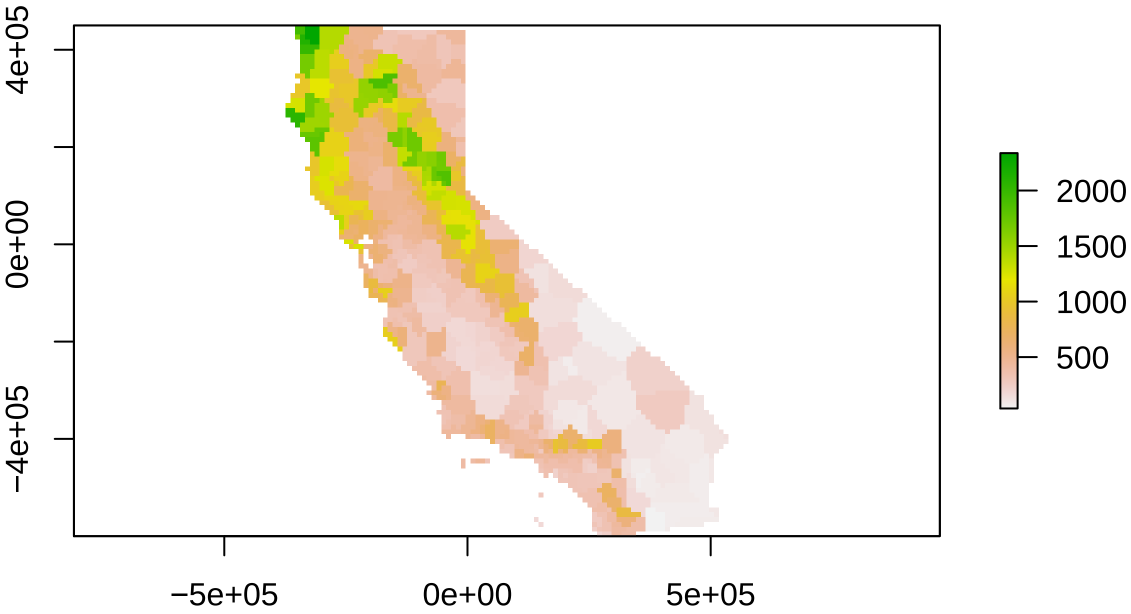
En importante destacar que un modelo IDW, pero con 1 vecino es igual con un modelo de proximidad con un vecino, esto porque la diferencia entre ambos modelos está en la ponderación que le da a las observaciones más cercanas, pero con un vecino el pesó será el mismo.

Tal como se aprecia en los siguientes gráficos, ambos modelos dan el mismo resultado



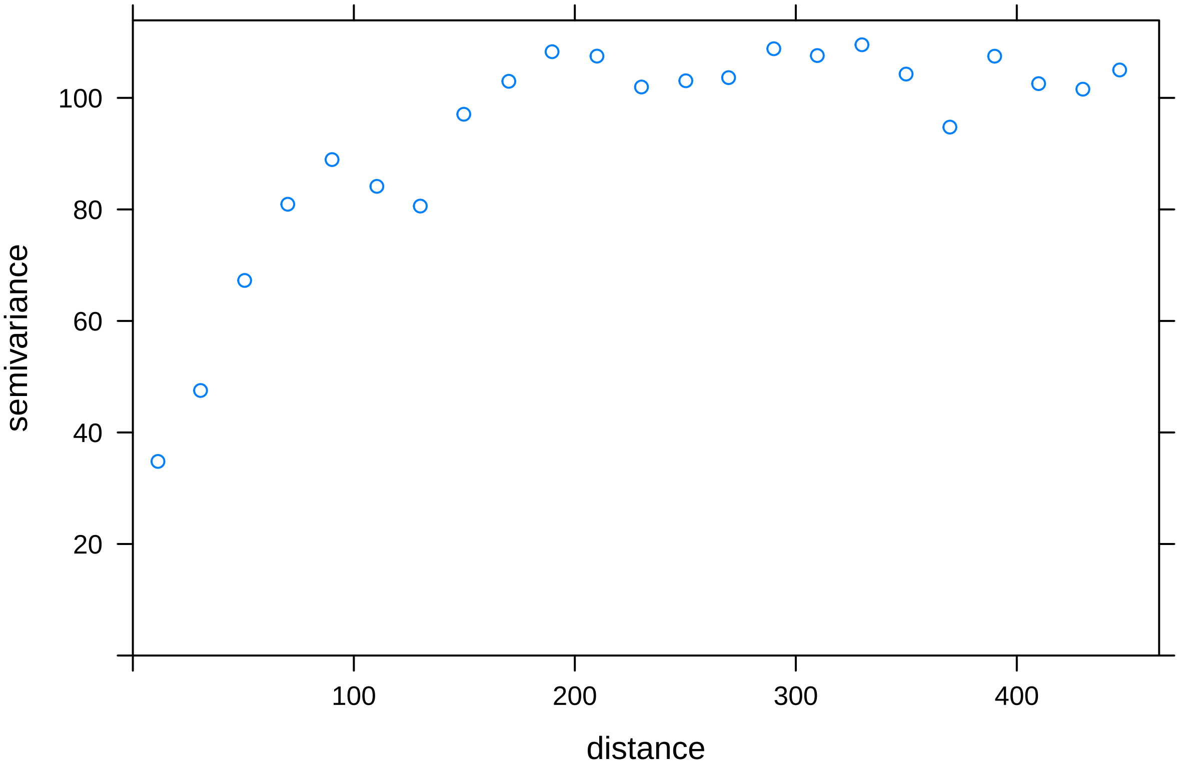
Modelo de IDW con un vecino

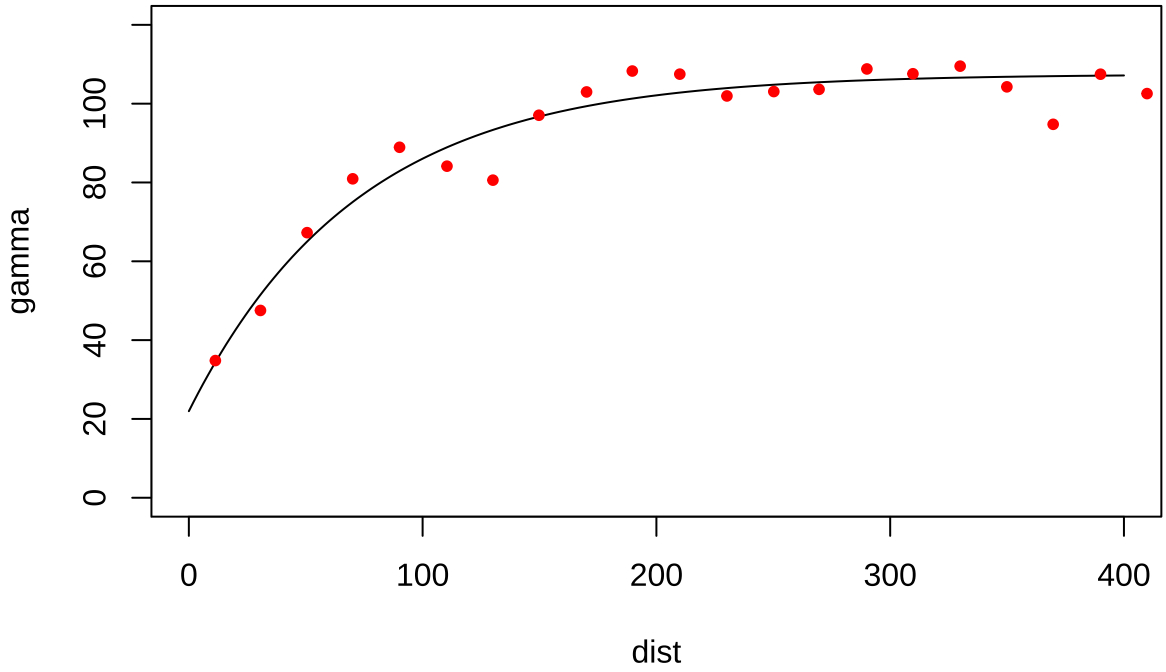
Modelo de poligonos de proximaidad con un vecino

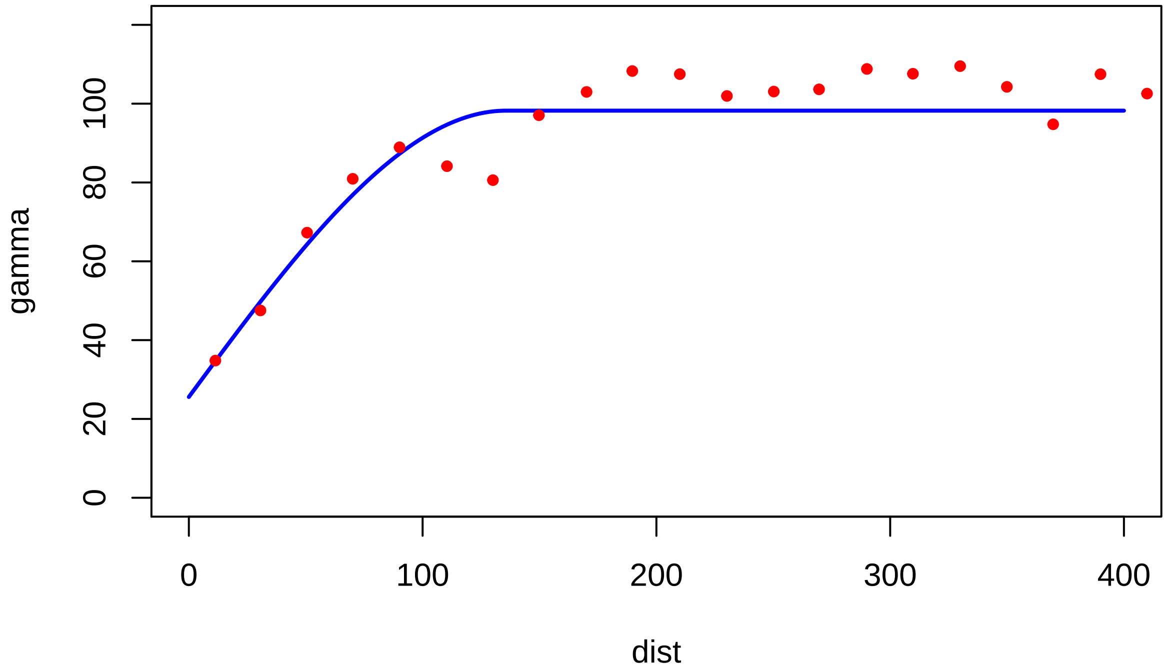


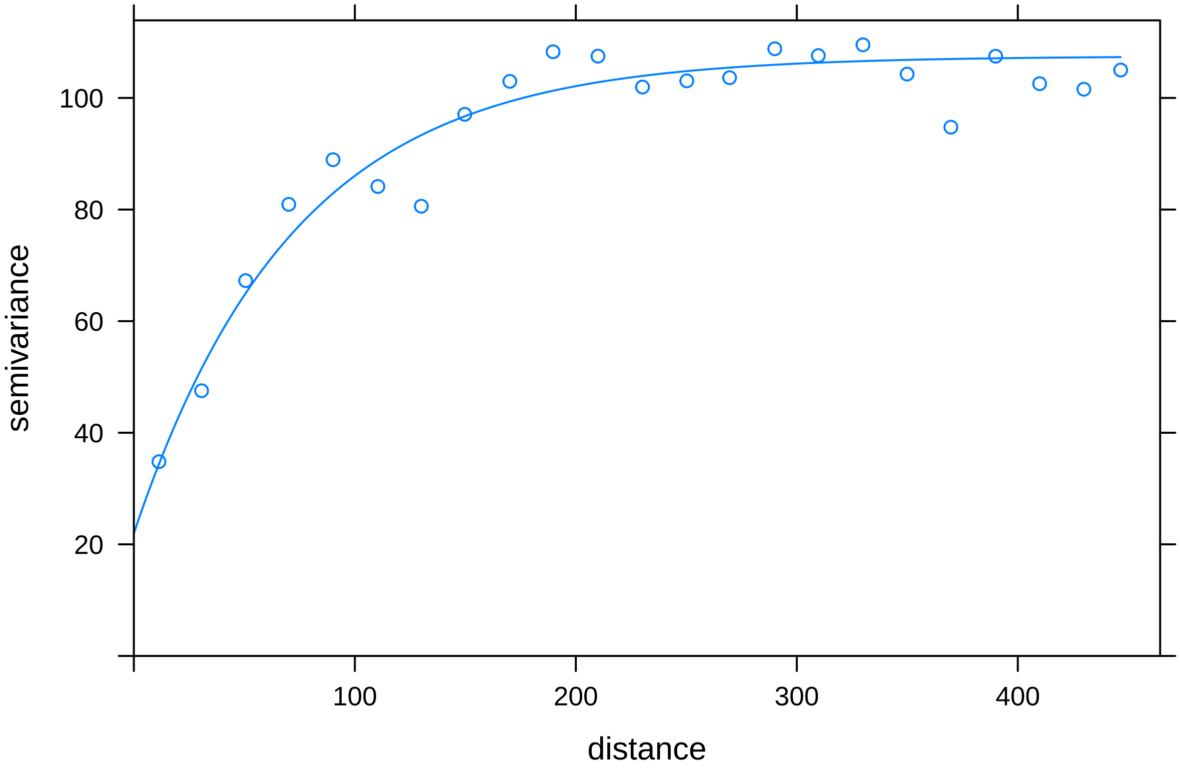
Contaminación en el Aire en el estado de California

En este segmento del laboratorio se utilizan los datos de contaminación del aire del California, inicialmente se hace la lectura de los datos, se utilizan datos del paquete de sp, posterior al tratamiento de los datos se hace un análisis de variograma, específicamente se observan los siguientes variogramas:


semivariance
20
40
60
80
100
distance
100
200
300
400


gamma
0
20
40
60
80
100
0
100
200
300
400
dist


gamma
0
20
40
60
80
100
0
100
200
300
400
dist

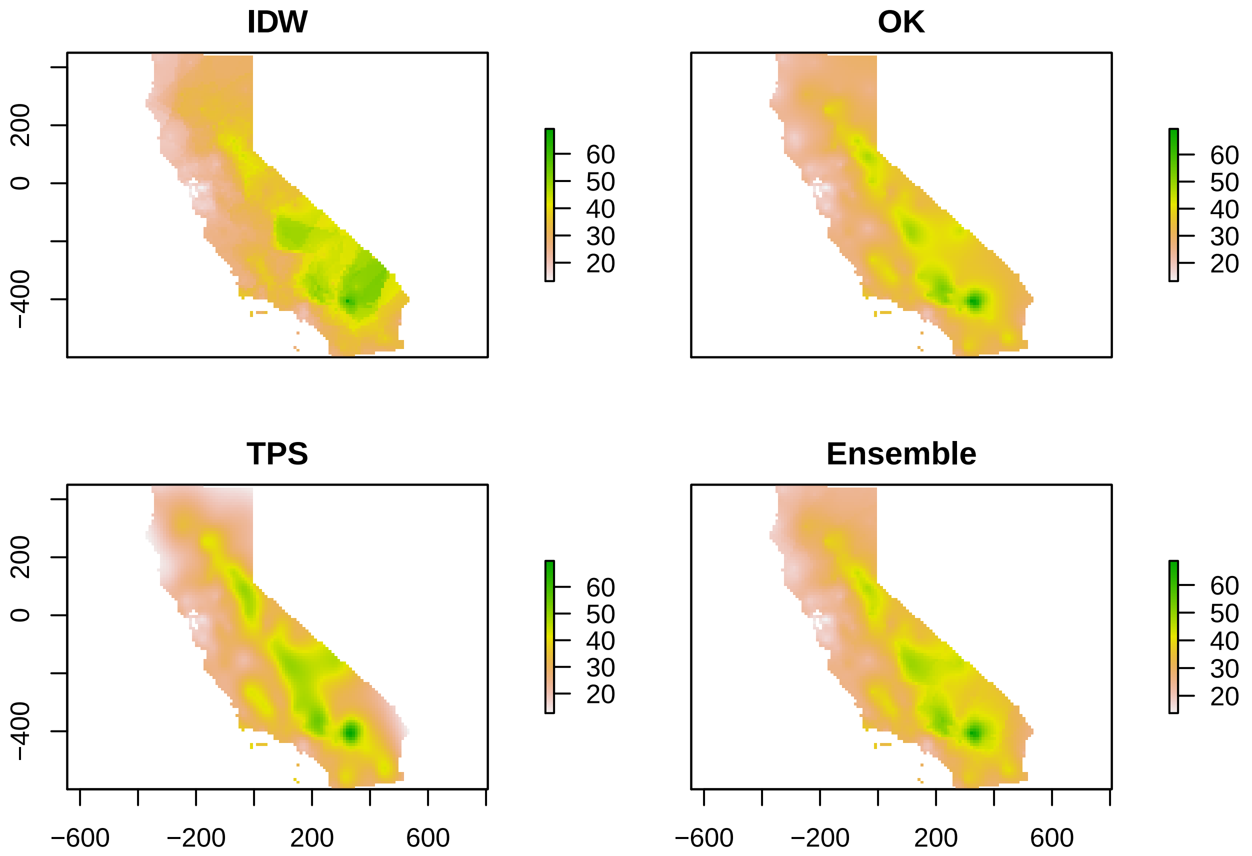


Posteriormente, se realizan otros modelos y mediante validación cruzada se compara cual da un mejor resultado, los modelos que se estiman con el IDW normal, posteriormente el IDW optimo y por último el spline de placa delgada.

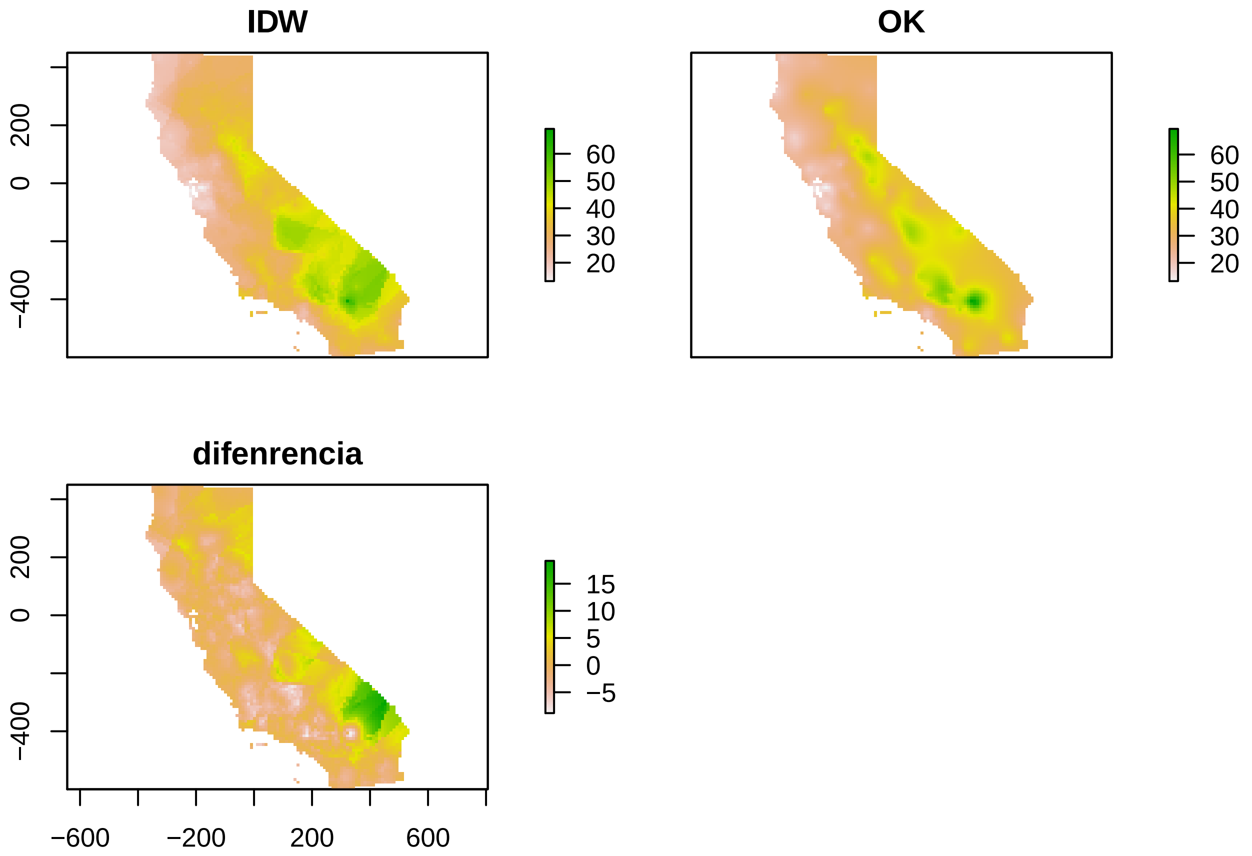
En general se obtiene como resultado que el modelo de spline de placa delgada es que el arroja un valor de RMSE menor, los siguiente son los valores comparativos:

IDW normal: 8.041305 IDW óptimo: 8.307235 Modelo de spline de placa delgada: 7.930799

En general el comparativo de los tres modelos más uno poderados se se puede observar en el siguiente grafico:



Posteriormente, se hace el ejercicio de comparar el modelo de IDW con el optimo y con ello observar donde hay mayor cantidad de diferencias. El resultado se puede apreciar en el siguiente gráfico:



Destaca que la mayor derencia está en la parte inferior de california

En general, la mayor diferencia en la proyección con ambos métodos se da en la parte sur, la cual está coloriada de verde. Eso se pude deber a que con el modelo optimo toma en cuenta una mayor cantidad de vecinos y con ellos la proyección en esa zona es menor porque toma vecinos con menor contaminación, mientras que con el IDW normal toma en cuenta los vecinos más cercanos y es el área con mayor contaminación

A modo de conclusión, el informe trató sobre la interpolación de datos espaciales, se utilizaron dos set de datos, en el primer caso se utilizaron datos de la temperatura de california, en este caso calcularon tres modelos, el nulo que es mas que el promedio, el de vecinos mas cercano y el distancia inversa ponderada, estos últimos dos modelos se calcularon con el nulos mediante el rmse y como resultado se obtiene que estos dan un mejor resultado que el nulo.

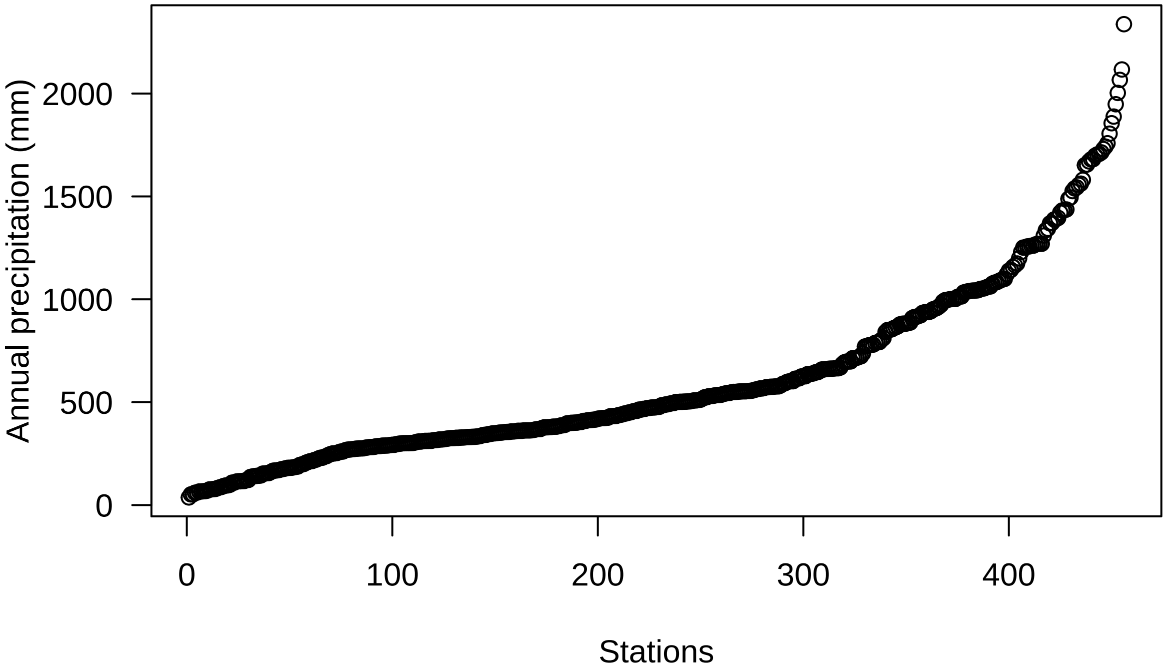
En el segundo set de datos se utilizaron datos de la contaminación del aire en california, en este caso primeramente de analizan los variograma, posteriormente se analizan los modelos de distancia inversa ponderada normal y el optimizado y otro de capa delgada, todos de comparan mediante validación cruzada, también con el indicador de rmse y en general todos dan resultados muy similares

Anexos : Código completo

```
library(rspatial)
d <- sp_data('precipitation')
head(d)
```

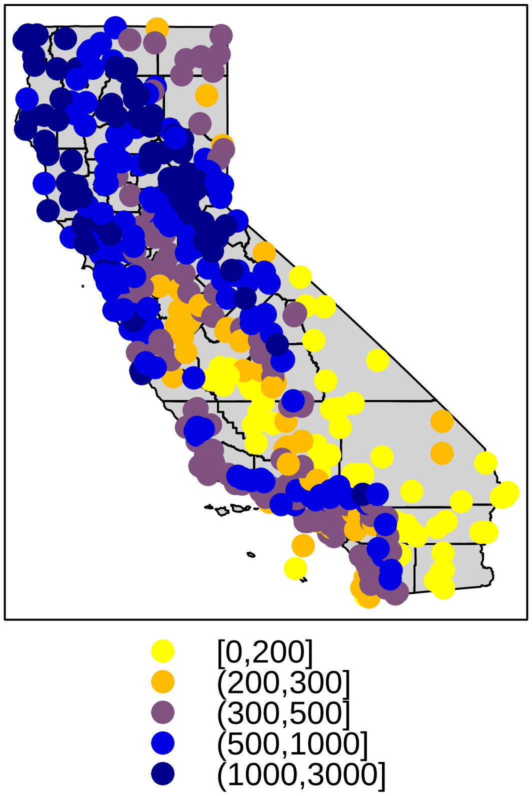
```
##      ID                 NAME   LAT    LONG ALT  JAN FEB MAR APR MAY JUN JUL
## 1 ID741         DEATH VALLEY 36.47 -116.87 -59  7.4 9.5 7.5 3.4 1.7 1.0 3.7
## 2 ID743  THERMAL/FAA AIRPORT 33.63 -116.17 -34  9.2 6.9 7.9 1.8 1.6 0.4 1.9
## 3 ID744          BRAWLEY 2SW 32.96 -115.55 -31 11.3 8.3 7.6 2.0 0.8 0.1 1.9
## 4 ID753 IMPERIAL/FAA AIRPORT 32.83 -115.57 -18 10.6 7.0 6.1 2.5 0.2 0.0 2.4
## 5 ID754               NILAND 33.28 -115.51 -18  9.0 8.0 9.0 3.0 0.0 1.0 8.0
## 6 ID758        EL CENTRO/NAF 32.82 -115.67 -13  9.8 1.6 3.7 3.0 0.4 0.0 3.0
##    AUG SEP OCT NOV DEC
## 1  2.8 4.3 2.2 4.7 3.9
## 2  3.4 5.3 2.0 6.3 5.5
## 3  9.2 6.5 5.0 4.8 9.7
## 4  2.6 8.3 5.4 7.7 7.3
## 5  9.0 7.0 8.0 7.0 9.0
## 6 10.8 0.2 0.0 3.3 1.4
```

```
d$prec <- rowSums(d[, c(6:17)])
plot(sort(d$prec), ylab='Annual precipitation (mm)', las=1, xlab='Stations')
```



```
library(sp)
dsp <- SpatialPoints(d[,4:3], proj4string=CRS("+proj=longlat +datum=NAD83"))
dsp <- SpatialPointsDataFrame(dsp, d)
CA <- sp_data("counties")
# define groups for mapping
cuts <- c(0,200,300,500,1000,3000)
# set up a palette of interpolated colors
blues <- colorRampPalette(c('yellow', 'orange', 'blue', 'dark blue'))
pols <- list("sp.polygons", CA, fill = "lightgray")
spplot(dsp, 'prec', cuts=cuts, col.regions=blues(5), sp.layout=pols, pch=20, cex=2)
```

```
## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum North_American_Da
##  but +towgs84= values preserved
```



```
TA <- CRS("+proj=aea +lat_1=34 +lat_2=40.5 +lat_0=0 +lon_0=-120 +x_0=0 +y_0=-4000000 +datum=NAD83 +units
```

```
## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum North_American_Da
##  but +towgs84= values preserved
```

```
library(rgdal)
dta <- spTransform(dsp, TA)
cata <- spTransform(CA, TA)
```

```
## Warning in spTransform(xSP, CRSobj, ...): NULL source CRS comment, falling back
## to PROJ string
```

NULL model

```
RMSE <- function(observed, predicted) {
  sqrt(mean((predicted - observed)^2, na.rm=TRUE))
}

null <- RMSE(mean(dsp$prec), dsp$prec)
null
```

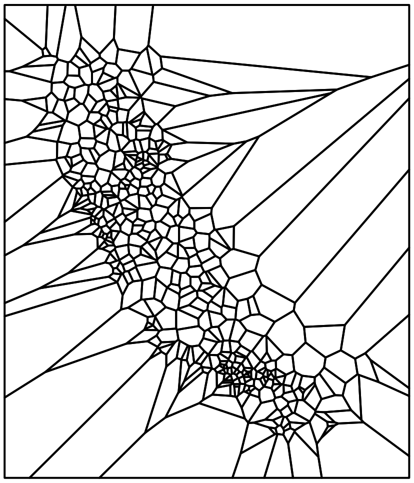
```
## [1] 435.3217
```

proximity polygons

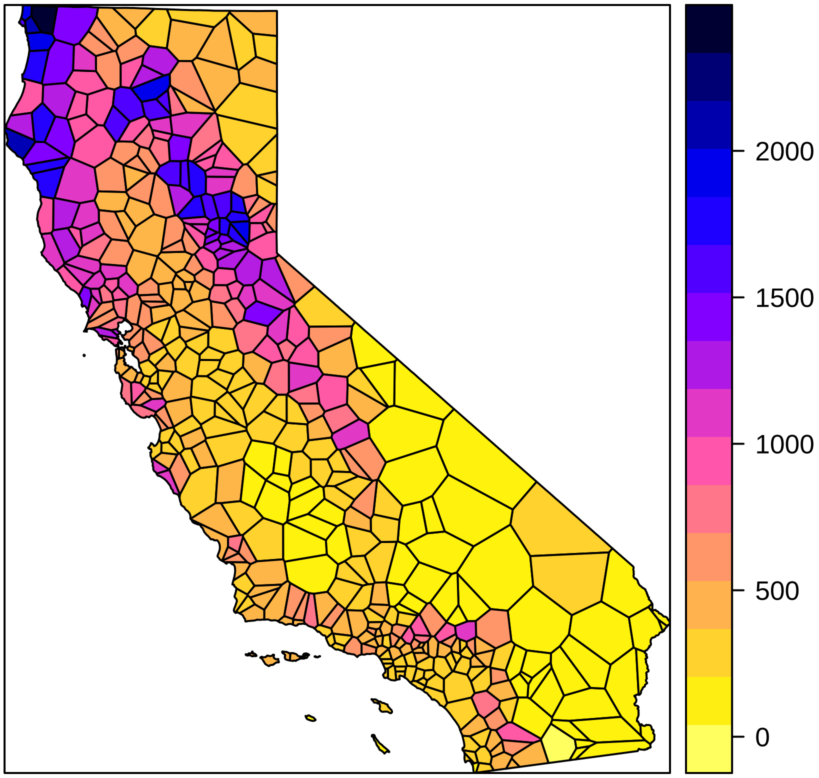
```
library(dismo)
v <- voronoi(dta)
```

```
## Warning in proj4string(xy): CRS object has comment, which is lost in output
## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on
##  but +towgs84= values preserved
```

```
plot(v)
```

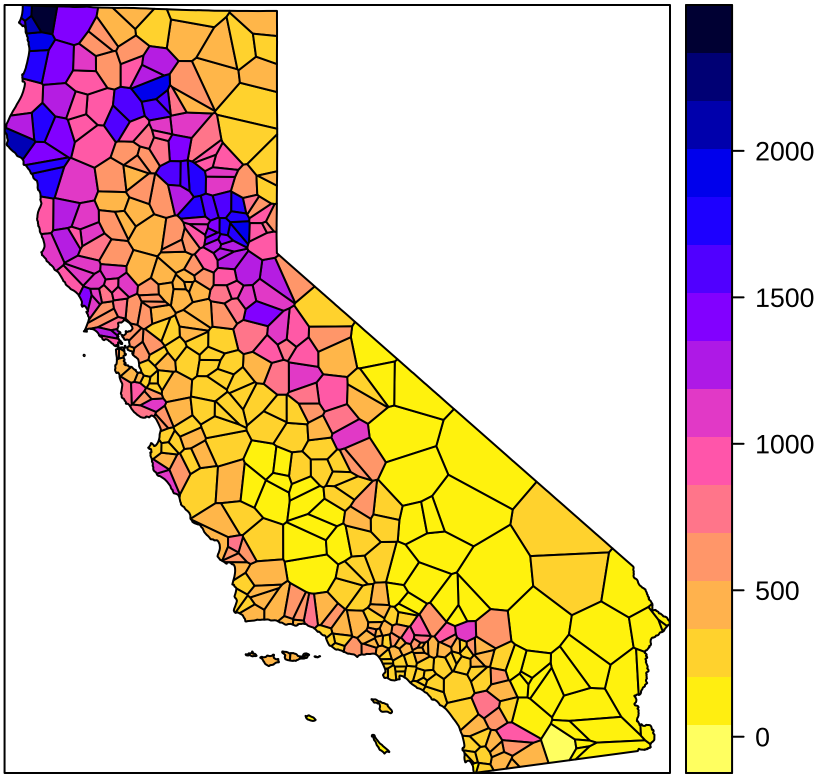


```
ca <- aggregate(cata)
vca <- intersect(v, ca)
spplot(vca, 'prec', col.regions=rev(get_col_regions()))
```

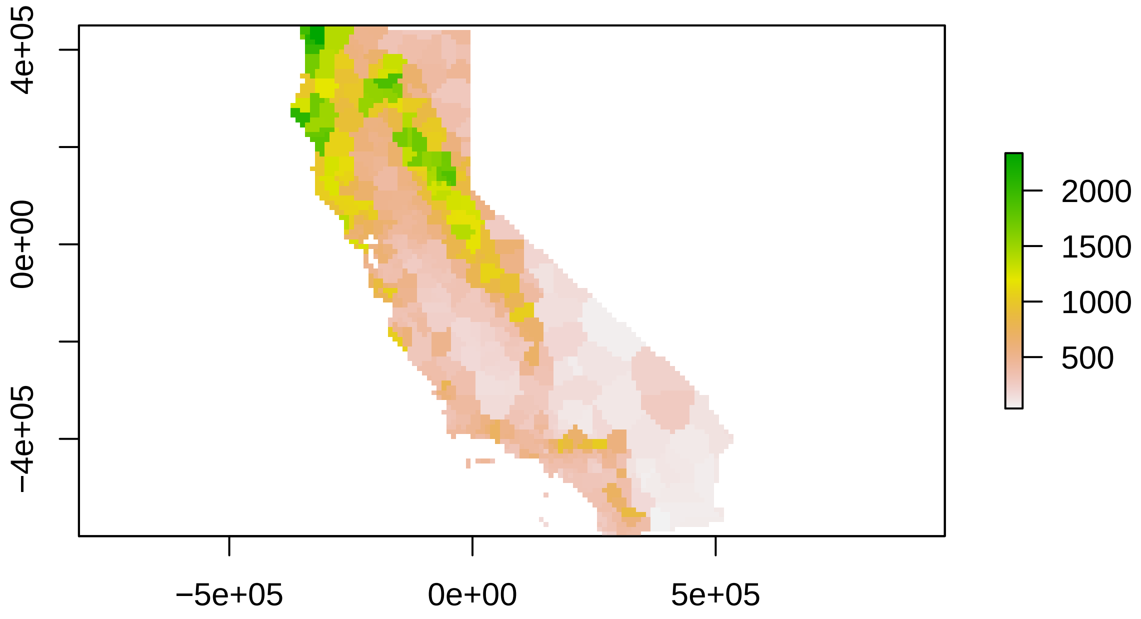


Looks weird. Let's confine this to California

```
ca <- aggregate(cata)
vca <- intersect(v, ca)
spplot(vca, 'prec', col.regions=rev(get_col_regions()))
```



```
r <- raster(cata, res=10000)
vr <- rasterize(vca, r, 'prec')
plot(vr)
```



Now evaluate with 5-fold cross validation

```
set.seed(5132015)
kf <- kfold(nrow(dta))
rmse <- rep(NA, 5)
for (k in 1:5) {
  test <- dta[kf == k, ]
  train <- dta[kf != k, ]
  v <- voronoi(train)
  p <- extract(v, test)
  rmse[k] <- RMSE(test$prec, p$prec)
}
```

```
## Warning in proj4string(xy): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on (
##  but +towgs84= values preserved

## Warning in proj4string(x): CRS object has comment, which is lost in output

## Warning in proj4string(y): CRS object has comment, which is lost in output

## Warning in RGEOSBinPredFunc(spgeom1, spgeom2, byid, func): spgeom1 and spgeom2
## have different proj4 strings

## Warning in proj4string(xy): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on (
##  but +towgs84= values preserved
## Warning in proj4string(x): CRS object has comment, which is lost in output

## Warning in proj4string(y): CRS object has comment, which is lost in output

## Warning in RGEOSBinPredFunc(spgeom1, spgeom2, byid, func): spgeom1 and spgeom2
## have different proj4 strings

## Warning in proj4string(xy): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on
##  but +towgs84= values preserved

## Warning in proj4string(x): CRS object has comment, which is lost in output

## Warning in proj4string(y): CRS object has comment, which is lost in output

## Warning in RGEOSBinPredFunc(spgeom1, spgeom2, byid, func): spgeom1 and spgeom2
## have different proj4 strings

## Warning in proj4string(xy): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on
##  but +towgs84= values preserved

## Warning in proj4string(x): CRS object has comment, which is lost in output

## Warning in proj4string(y): CRS object has comment, which is lost in output

## Warning in RGEOSBinPredFunc(spgeom1, spgeom2, byid, func): spgeom1 and spgeom2
## have different proj4 strings

## Warning in proj4string(xy): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on
##  but +towgs84= values preserved

## Warning in proj4string(x): CRS object has comment, which is lost in output

## Warning in proj4string(y): CRS object has comment, which is lost in output

## Warning in RGEOSBinPredFunc(spgeom1, spgeom2, byid, func): spgeom1 and spgeom2
## have different proj4 strings
```

```r
rmse
```

```
## [1] 199.0686 187.8069 166.9153 191.0938 238.9696
```

```r
mean(rmse)
```

```
## [1] 196.7708
```

```r
1 - (mean(rmse) / null)
```

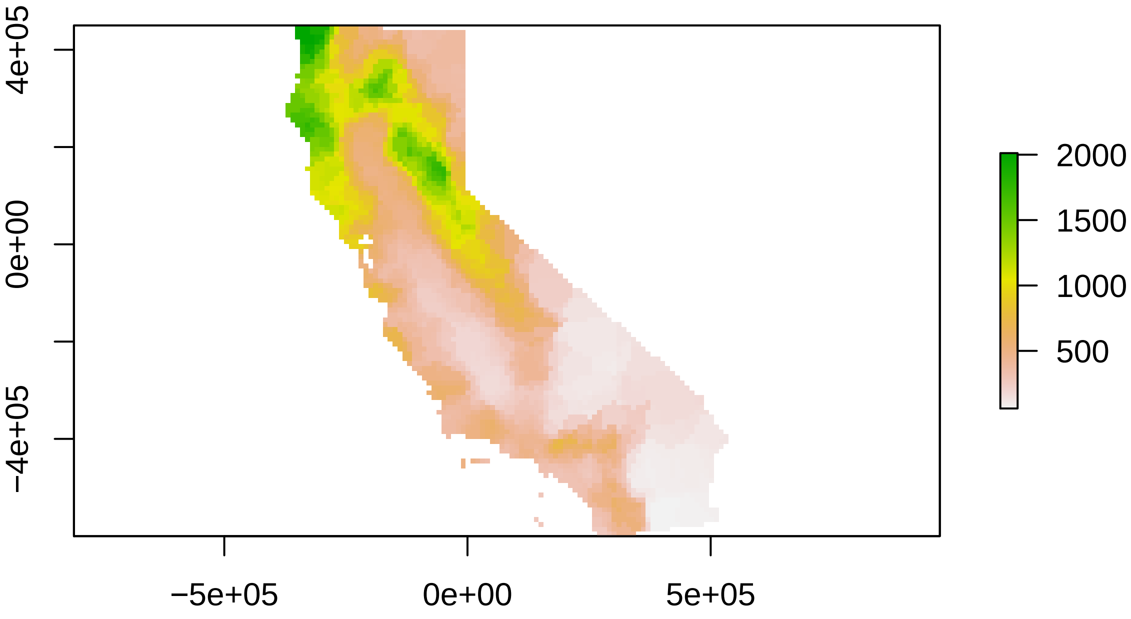
```
## [1] 0.5479875
```

Nearest neighbour interpolation

```r
library(gstat)
gs <- gstat(formula=prec~1, locations=dta, nmax=5, set=list(idp = 0))
nn <- interpolate(r, gs)
```

```
## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## [inverse distance weighted interpolation]

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on C
##  but +towgs84= values preserved

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on C
##  but +towgs84= values preserved

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on C
##  but +towgs84= values preserved

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on C
##  but +towgs84= values preserved
```

```
## [inverse distance weighted interpolation]
nnmsk <- mask(nn, vr)
plot(nnmsk)
```



```r
rmsenn <- rep(NA, 5)
for (k in 1:5) {
  test <- dta[kf == k, ]
  train <- dta[kf != k, ]
  gscv <- gstat(formula=prec~1, locations=train, nmax=5, set=list(idp = 0))
  p <- predict(gscv, test)$var1.pred
  rmsenn[k] <- RMSE(test$prec, p)
}
```

```
## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## [inverse distance weighted interpolation]

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on (
##  but +towgs84= values preserved

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## [inverse distance weighted interpolation]

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on (
##  but +towgs84= values preserved
## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## [inverse distance weighted interpolation]

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on
##  but +towgs84= values preserved

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## [inverse distance weighted interpolation]

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on
##  but +towgs84= values preserved

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## [inverse distance weighted interpolation]

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on
##  but +towgs84= values preserved
```

```
rmsenn
```

```
## [1] 200.6222 190.8336 180.3833 169.9658 237.9067
```

```
mean(rmsenn)
```

```
## [1] 195.9423
```

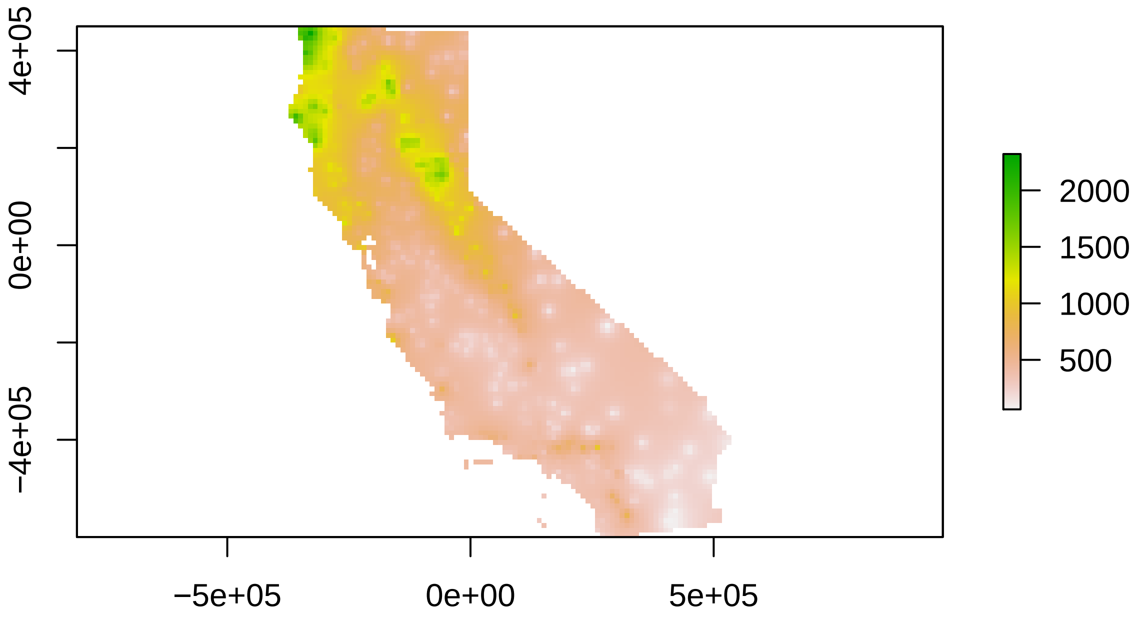
```
1 - (mean(rmsenn) / null)
```

```
## [1] 0.5498908
```

```
library(gstat)
gs <- gstat(formula=prec~1, locations=dta)
idw <- interpolate(r, gs)
```

```
## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## [inverse distance weighted interpolation]

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on
##  but +towgs84= values preserved

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on
##  but +towgs84= values preserved
## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output


## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on G
##  but +towgs84= values preserved

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output


## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on G
##  but +towgs84= values preserved
```

```r
## [inverse distance weighted interpolation]
idwr <- mask(idw, vr)
plot(idwr)
```



```r
rmse <- rep(NA, 5)
for (k in 1:5) {
  test <- dta[kf == k, ]
  train <- dta[kf != k, ]
  gs <- gstat(formula=prec~1, locations=train)
  p <- predict(gs, test)
  rmse[k] <- RMSE(test$prec, p$var1.pred)
}
```

```
## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## [inverse distance weighted interpolation]

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on
##  but +towgs84= values preserved

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## [inverse distance weighted interpolation]

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on
##  but +towgs84= values preserved

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## [inverse distance weighted interpolation]

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on
##  but +towgs84= values preserved

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## [inverse distance weighted interpolation]

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on
##  but +towgs84= values preserved

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## [inverse distance weighted interpolation]

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on
##  but +towgs84= values preserved
```

```r
rmse
```

```
## [1] 215.3319 211.9383 190.0231 211.8308 230.1893
```

```r
mean(rmse)
```

```
## [1] 211.8627
```

```r
1 - (mean(rmse) / null)
```

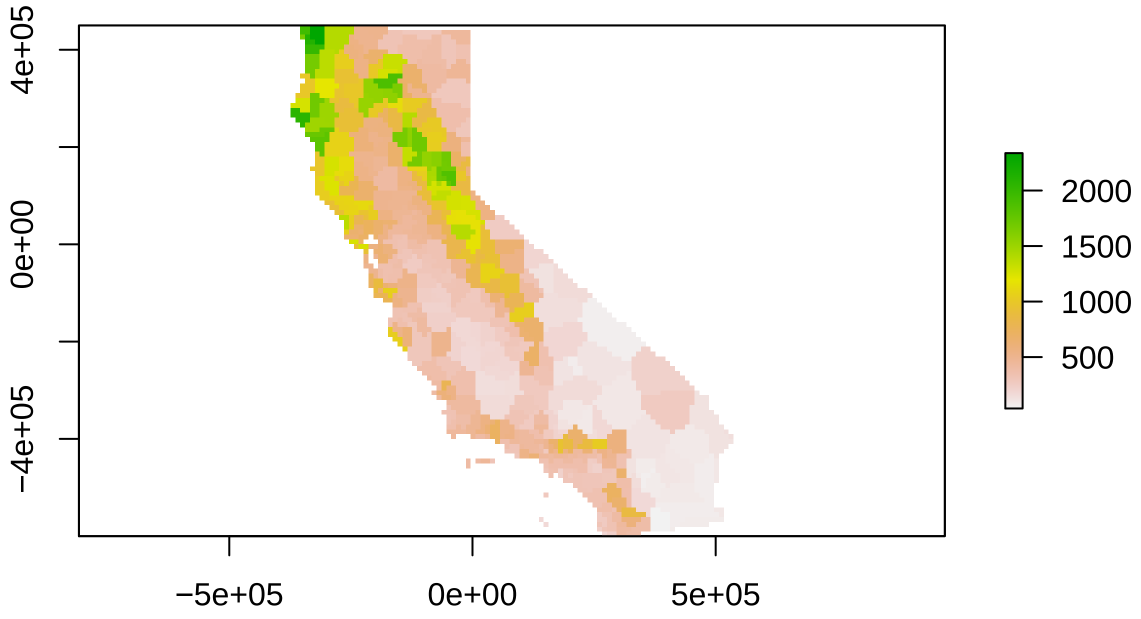
```
## [1] 0.5133192
```

modelo de IDW con un vecino

```
gs2 <- gstat(formula=prec~1, locations=dta, nmax=1, set=list(idp=1))
vmc <- interpolate(r, gs2)
```

```
## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## [inverse distance weighted interpolation]

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on
##  but +towgs84= values preserved

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on
##  but +towgs84= values preserved

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on
##  but +towgs84= values preserved

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on
##  but +towgs84= values preserved
```

```
## [inverse distance weighted interpolation]
vmcr <- mask(vmc, vr)
plot(vmcr)
```

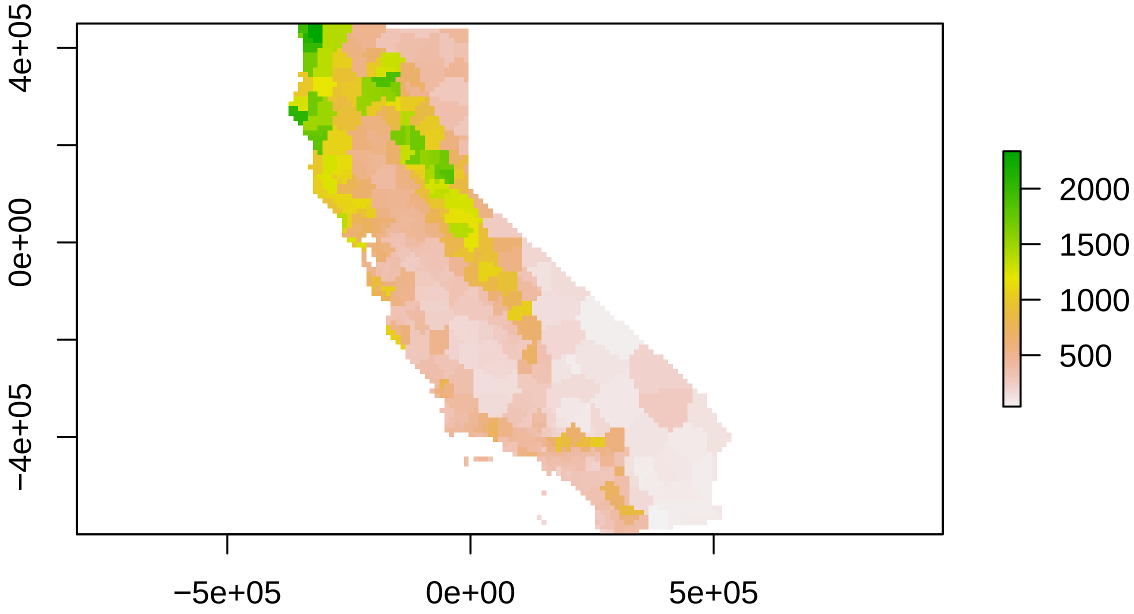


Modelo de poligonos de proximaidad con un vecino

```
gs3 <- gstat(formula=prec~1, locations=dta, nmax=1, set=list(idp=0))
vmc3 <- interpolate(r, gs3)
```

```
## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## [inverse distance weighted interpolation]

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on
##  but +towgs84= values preserved

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output


## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on
##  but +towgs84= values preserved

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output


## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on
##  but +towgs84= values preserved

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output


## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in showSRID(uprojargs, format = "PROJ", multiline = "NO"): Discarded datum Unknown based on (
##  but +towgs84= values preserved
```

```
## [inverse distance weighted interpolation]
vmcr3 <- mask(vmc3, vr)
plot(vmcr3)
```



Calfornia Air Pollution data

```
library(rspatial)
x <- sp_data("airqual")
x$OZDLYAV <- x$OZDLYAV * 1000


library(sp)
coordinates(x) <- ~LONGITUDE + LATITUDE
proj4string(x) <- CRS('+proj=longlat +datum=NAD83')
TA <- CRS("+proj=aea +lat_1=34 +lat_2=40.5 +lat_0=0 +lon_0=-120 +x_0=0 +y_0=-4000000 +datum=NAD83 +units
library(rgdal)
aq <- spTransform(x, TA)
```

```
cageo <- sp_data('counties.rds')
ca <- spTransform(cageo, TA)
```

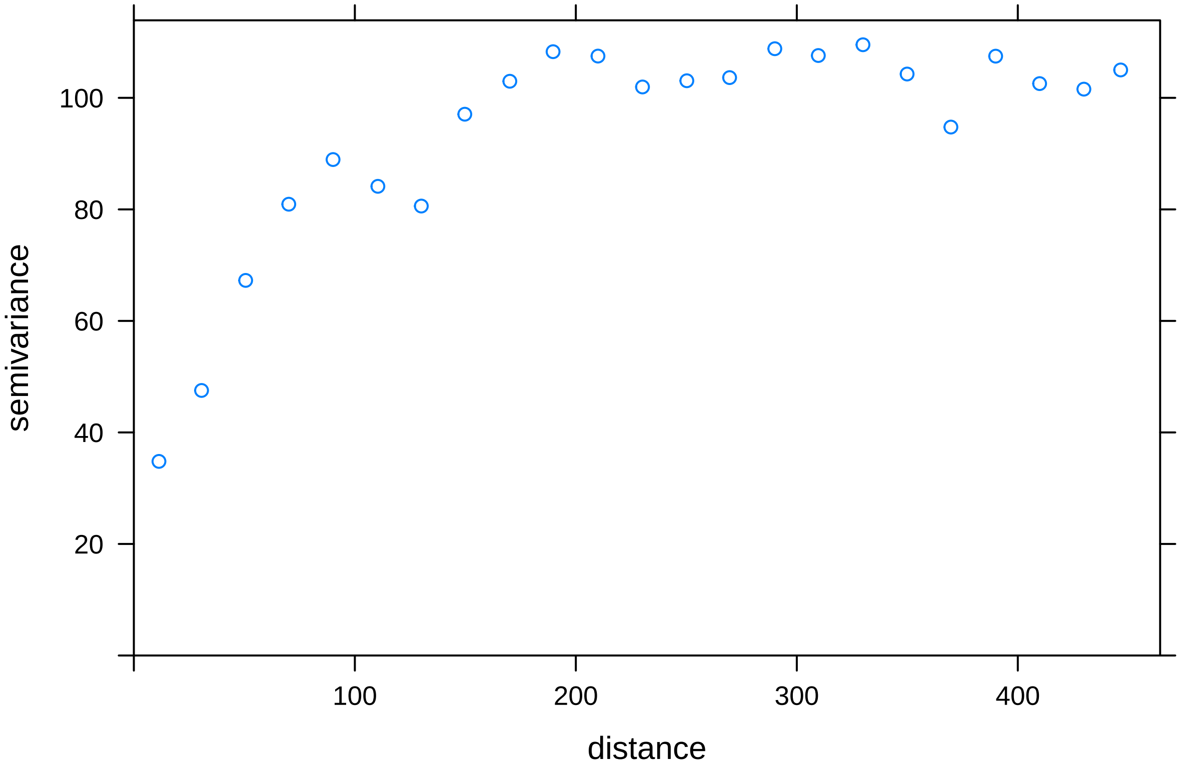
```
## Warning in spTransform(xSP, CRSobj, ...): NULL source CRS comment, falling back
## to PROJ string
```

```
r <- raster(ca)
res(r) <- 10  # 10 km if your CRS's units are in km
g <- as(r, 'SpatialGrid')
```

```
library(gstat)
gs <- gstat(formula=OZDLYAV~1, locations=aq)
v <- variogram(gs, width=20)
head(v)
```

```
##     np      dist    gamma dir.hor dir.ver   id
## 1 1010  11.35040 34.80579       0       0 var1
## 2 1806  30.63737 47.52591       0       0 var1
## 3 2355  50.58656 67.26548       0       0 var1
## 4 2619  70.10411 80.92707       0       0 var1
## 5 2967  90.13917 88.93653       0       0 var1
## 6 3437 110.42302 84.13589       0       0 var1
```

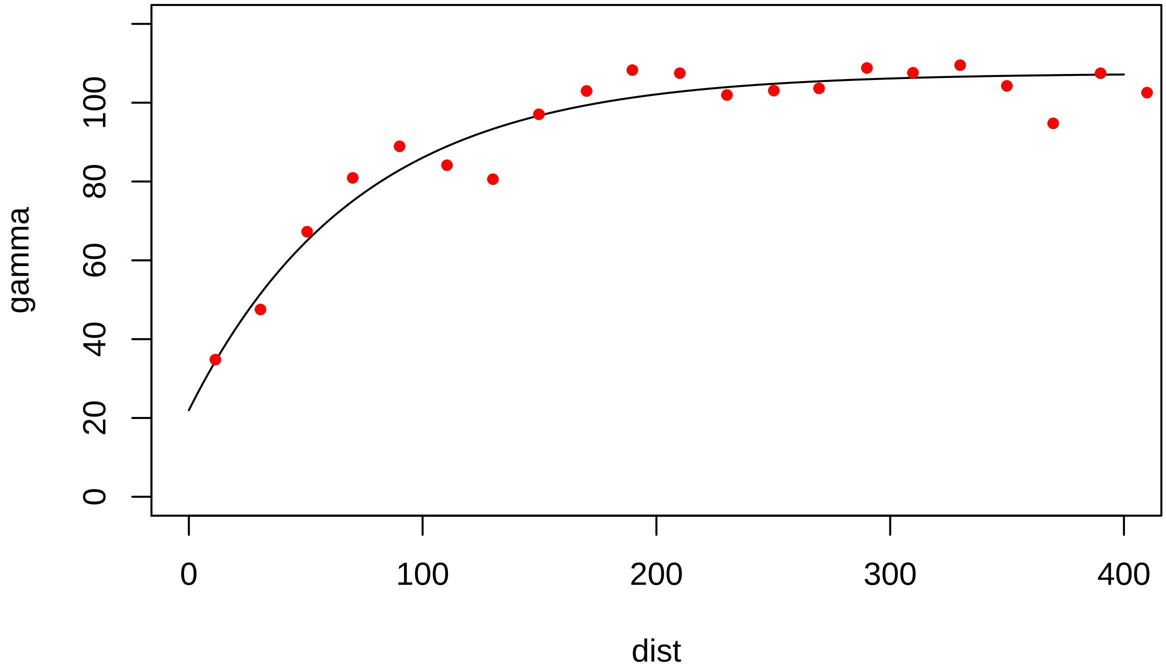
```
plot(v)
```



```
fve <- fit.variogram(v, vgm(85, "Exp", 75, 20))
fve
```

```
##   model    psill    range
## 1   Nug 21.96600  0.00000
## 2   Exp 85.52957 72.31404
```

```
plot(variogramLine(fve, 400), type='l', ylim=c(0,120))
points(v[,2:3], pch=20, col='red')
```

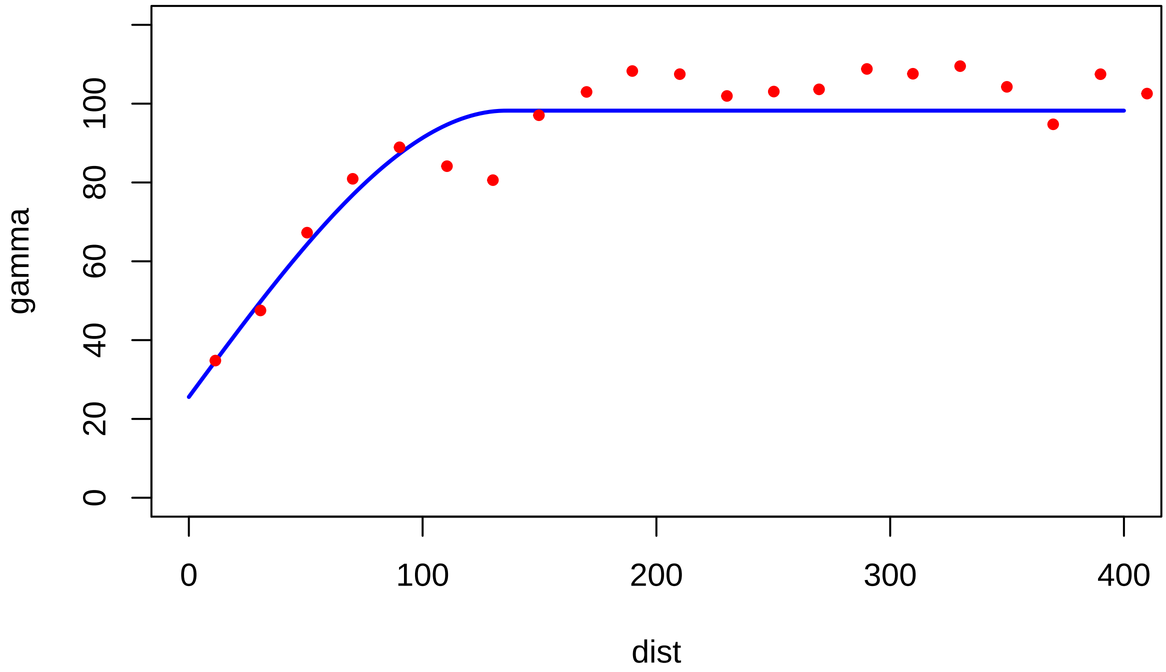


Try a different type (spherical in stead of exponential)

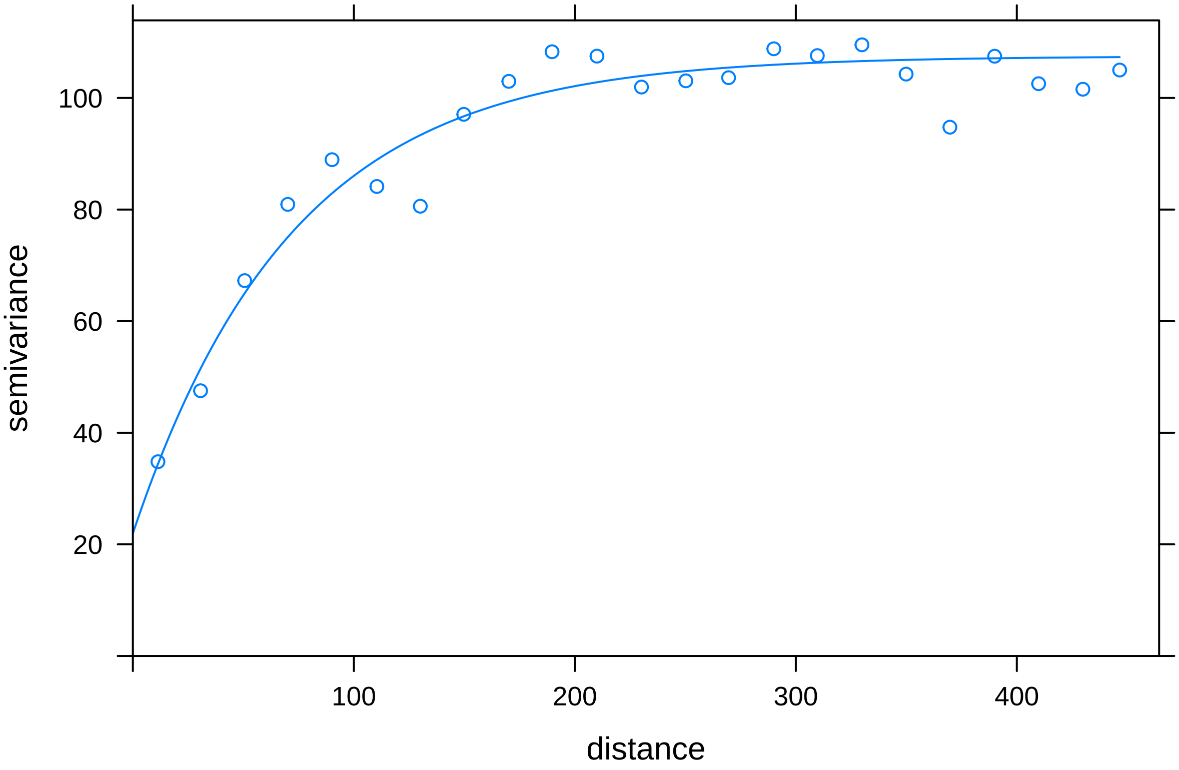
```
fvs <- fit.variogram(v, vgm(85, "Sph", 75, 20))
fvs
```

```
##   model    psill    range
## 1   Nug 25.57019   0.0000
## 2   Sph 72.65881 135.7744
```

```
plot(variogramLine(fvs, 400), type='l', ylim=c(0,120) ,col='blue', lwd=2)
points(v[,2:3], pch=20, col='red')
```



```r
plot(v, fve)
```

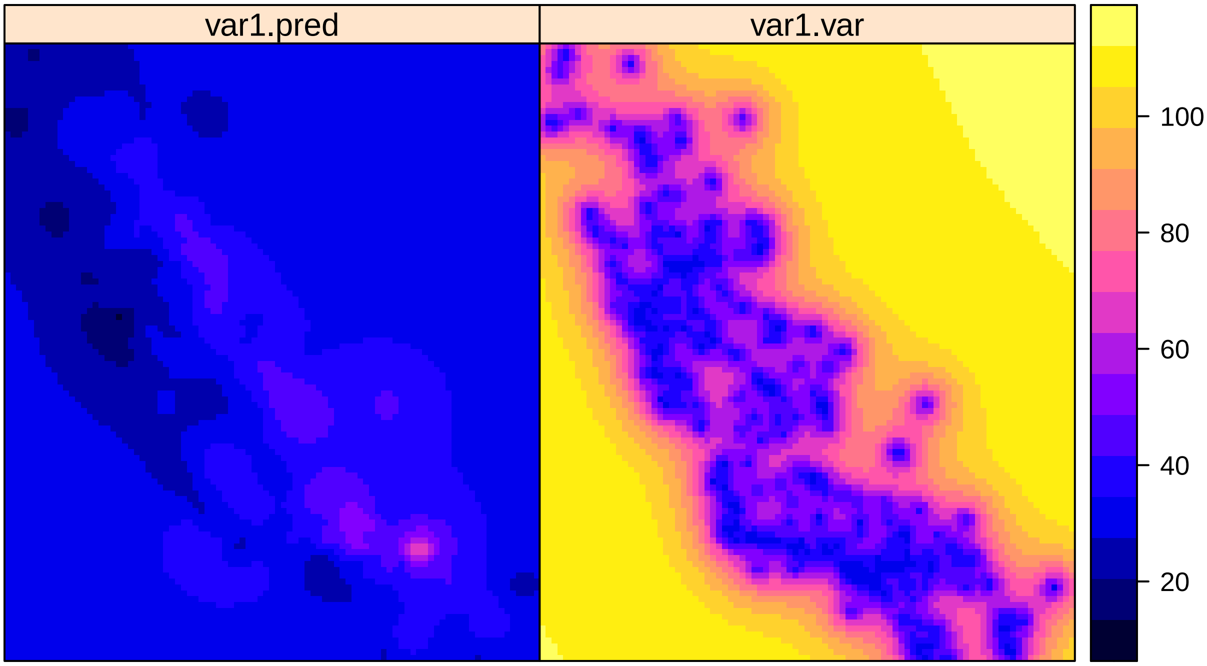


Ordinary kriging

```
k <- gstat(formula=OZDLYAV~1, locations=aq, model=fve)
kp <- predict(k, g)
```

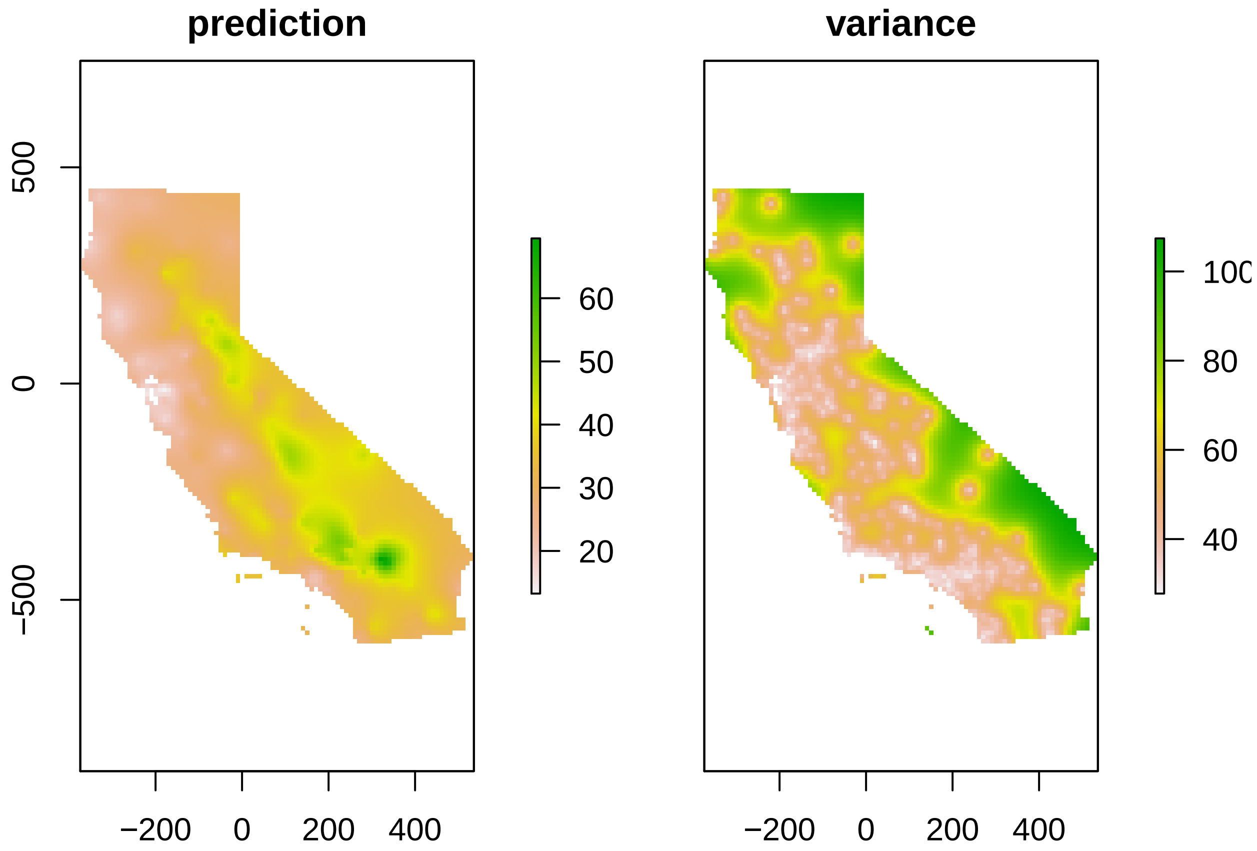
```
## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## [using ordinary kriging]

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
spplot(kp)
```



variance

```
ok <- brick(kp)
ok <- mask(ok, ca)
names(ok) <- c('prediction', 'variance')
plot(ok)
```

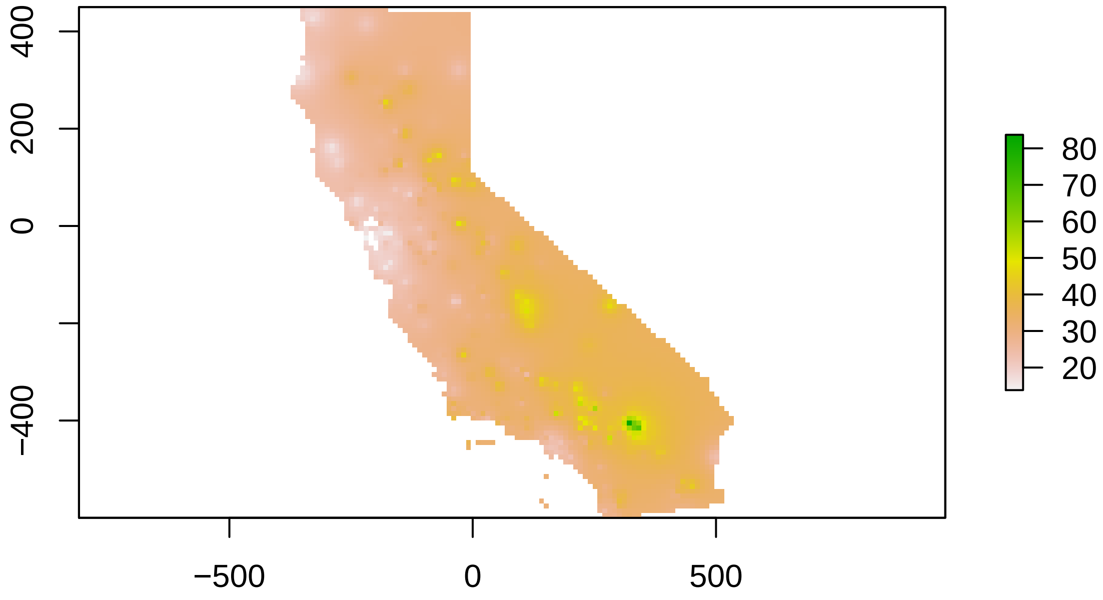


Compare with other methods

```
library(gstat)
idm <- gstat(formula=OZDLYAV~1, locations=aq)

idp <- interpolate(r, idm)
```

```
## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## [inverse distance weighted interpolation]

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## [inverse distance weighted interpolation]
idp <- mask(idp, ca)
plot(idp)
```



```
RMSE <- function(observed, predicted) {
  sqrt(mean((predicted - observed)^2, na.rm=TRUE))
}
f1 <- function(x, test, train) {
  nmx <- x[1]
  idp <- x[2]
  if (nmx < 1) return(Inf)
  if (idp < .001) return(Inf)
  m <- gstat(formula=OZDLYAV~1, locations=train, nmax=nmx, set=list(idp=idp))
  p <- predict(m, newdata=test, debug.level=0)$var1.pred
  RMSE(test$OZDLYAV, p)
}
set.seed(20150518)
i <- sample(nrow(aq), 0.2 * nrow(aq))
tst <- aq[i,]
trn <- aq[-i,]
opt <- optim(c(8, .5), f1, test=tst, train=trn)
```

```
## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
opt
```

```
## $par
## [1] 9.2594442 0.6817524
##
## $value
## [1] 7.861426
##
## $counts
## function gradient
##       35       NA
##
## $convergence
## [1] 0
##
## $message
## NULL
```

Our optimal IDW model

```
m <- gstat(formula=OZDLYAV~1, locations=aq, nmax=opt$par[1], set=list(idp=opt$par[2]))
idw <- interpolate(r, m)
```

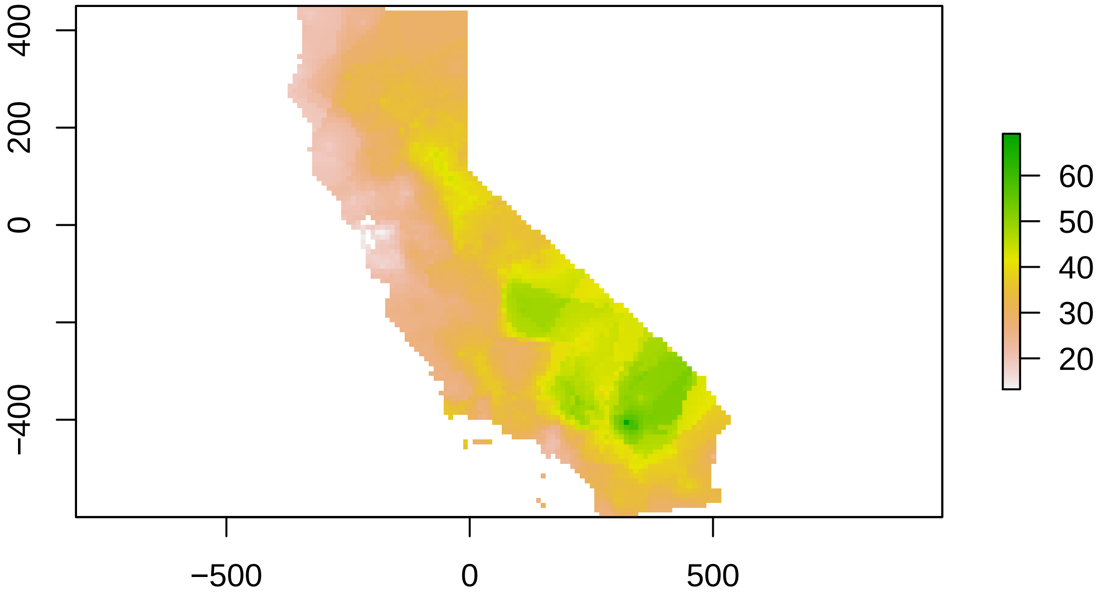
```
## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## [inverse distance weighted interpolation]

## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

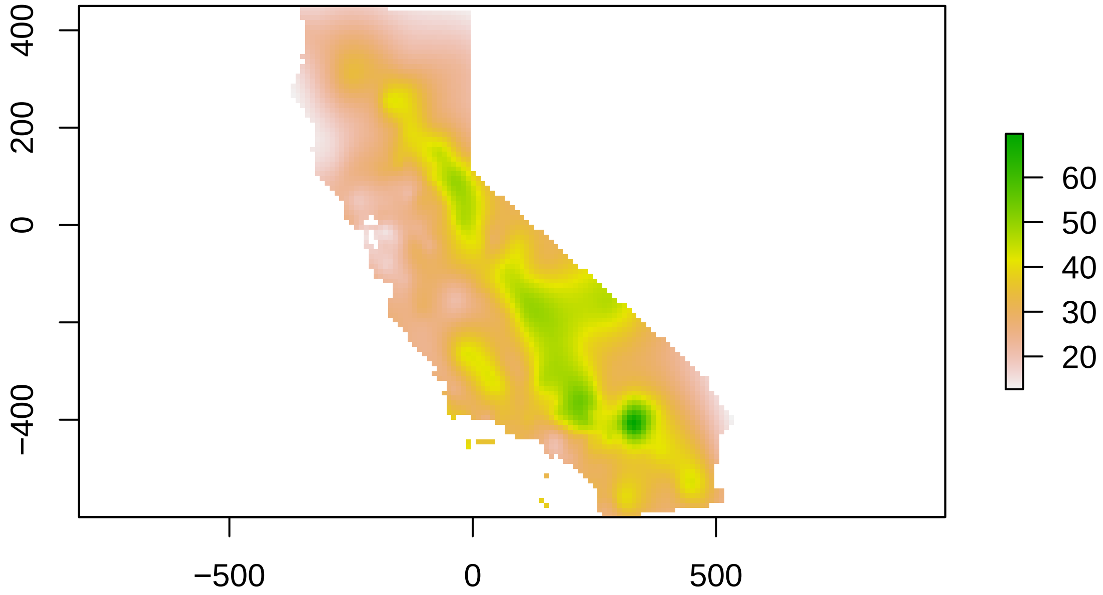
```
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

```
## [inverse distance weighted interpolation]
idw <- mask(idw, ca)
plot(idw)
```



A thin plate spline model

```
library(fields)
m <- Tps(coordinates(aq), aq$OZDLYAV)
tps <- interpolate(r, m)
tps <- mask(tps, idw)
plot(tps)
```



Cross-validate

```
library(dismo)
nfolds <- 5
k <- kfold(aq, nfolds)
ensrmse <- tpsrmse <- krigrmse <- idwrmse <- rep(NA, 5)
for (i in 1:nfolds) {
  test <- aq[k!=i,]
  train <- aq[k==i,]
  m <- gstat(formula=OZDLYAV~1, locations=train, nmax=opt$par[1], set=list(idp=opt$par[2]))
  p1 <- predict(m, newdata=test, debug.level=0)$var1.pred
  idwrmse[i] <-  RMSE(test$OZDLYAV, p1)
  m <- gstat(formula=OZDLYAV~1, locations=train, model=fve)
  p2 <- predict(m, newdata=test, debug.level=0)$var1.pred
  krigrmse[i] <-  RMSE(test$OZDLYAV, p2)
  m <- Tps(coordinates(train), train$OZDLYAV)
  p3 <- predict(m, coordinates(test))
  tpsrmse[i] <-  RMSE(test$OZDLYAV, p3)
  w <- c(idwrmse[i], krigrmse[i], tpsrmse[i])
  weights <- w / sum(w)
  ensemble <- p1 * weights[1] + p2 * weights[2] + p3 * weights[3]
  ensrmse[i] <-  RMSE(test$OZDLYAV, ensemble)
}
```

```
## Warning in proj4string(d$data): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
## Warning in proj4string(d$data): CRS object has comment, which is lost in output
## Warning in proj4string(newdata): CRS object has comment, which is lost in output

## Warning in proj4string(newdata): CRS object has comment, which is lost in output
```

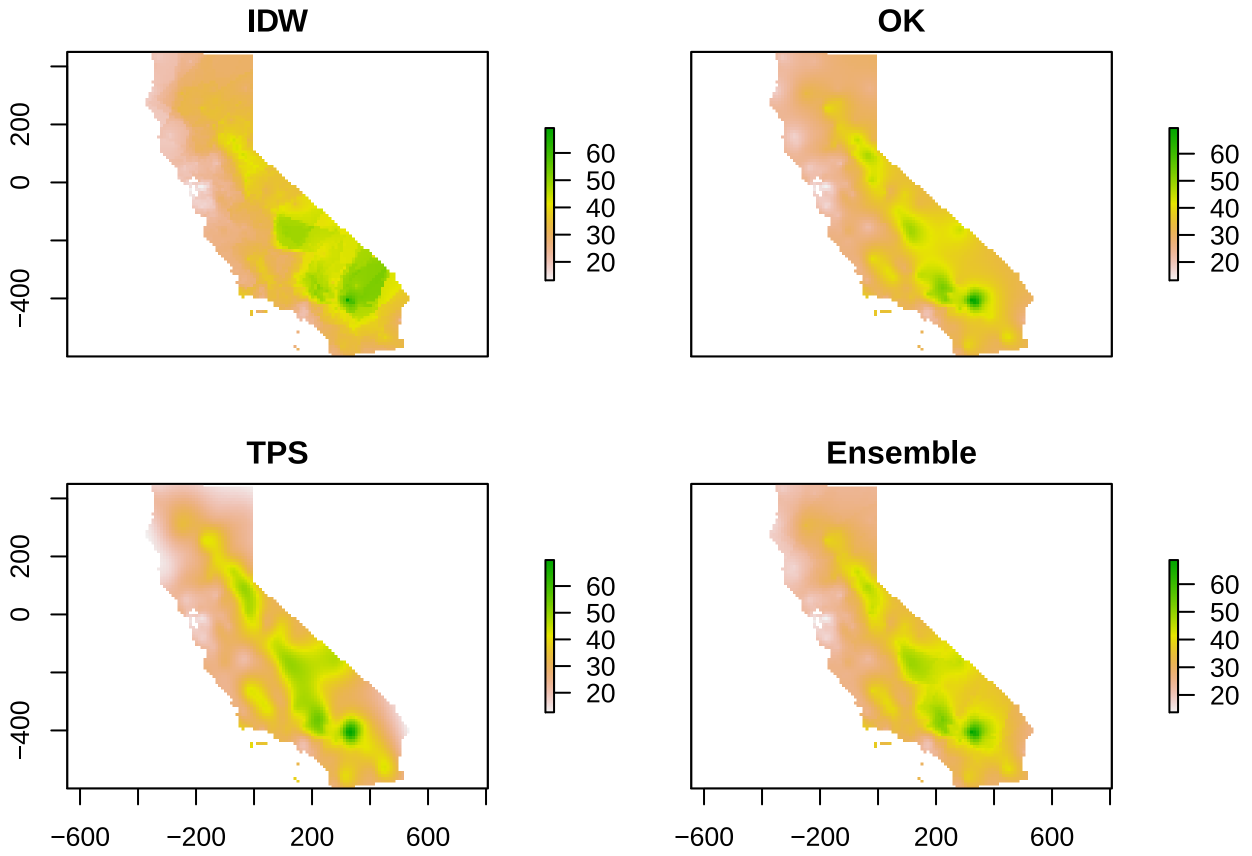
```r
rmi <- mean(idwrmse)
rmk <- mean(krigrmse)
rmt <- mean(tpsrmse)
rms <- c(rmi, rmt, rmk)
rms
```

```
## [1] 8.041305 8.307235 7.930799
```

```
rme <- mean(ensrmse)
rme
```

```
## [1] 7.858051
```

```
weights <- ( rms / sum(rms) )
s <- stack(idw, ok[[1]], tps)
ensemble <- sum(s * weights)

s <- stack(idw, ok[[1]], tps, ensemble)
names(s) <- c('IDW', 'OK', 'TPS', 'Ensemble')
plot(s)
```

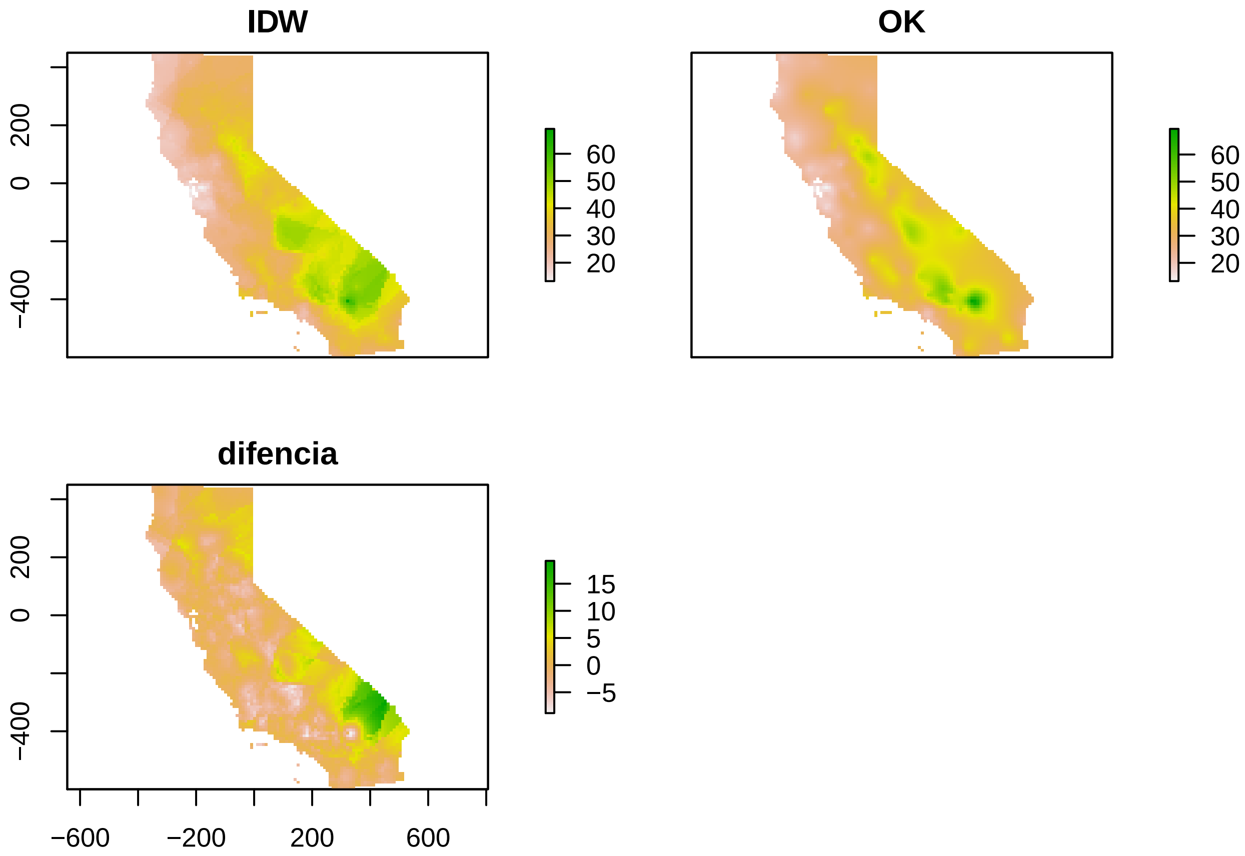


```
dd <- sum(idw- ok[[1]])
```

```
## Warning in sum(new("RasterLayer", file = new(".RasterFile", name = "",
## datanotation = "FLT4S", : Nothing to summarize if you provide a single
## RasterLayer; see cellStats
```

```
ss <- stack(idw, ok[[1]], dd)
names(ss) <- c('IDW', 'OK', 'difencia' )
```

```
plot(ss)
```



IDW

OK

difencia