

- Convex Omhulsel

↳ Het omhulsel vaarin alle punten vallen



- Triangulatie: => Point cloud



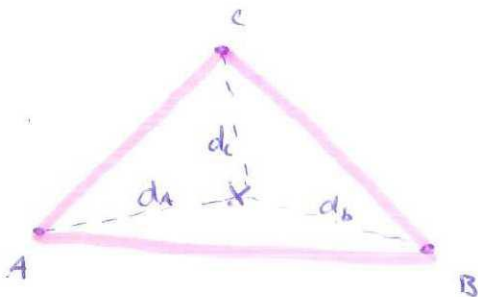
=> Delauney -> snel te maken

-> maximaliseer de minimum hoek van alle triangulaties

-> ieder punt is hoekpunt

-> omgeschreven cirkel om hoekpunten valt er geen andere punt x een driehoek te maken is.

- Triangular interpolatie



form.

$$h_x = \frac{A_{Ax}}{A_{tot}} \cdot h_c + \frac{A_{Bx}}{A_{tot}} \cdot h_b + \frac{A_{Cx}}{A_{tot}} \cdot h_a$$

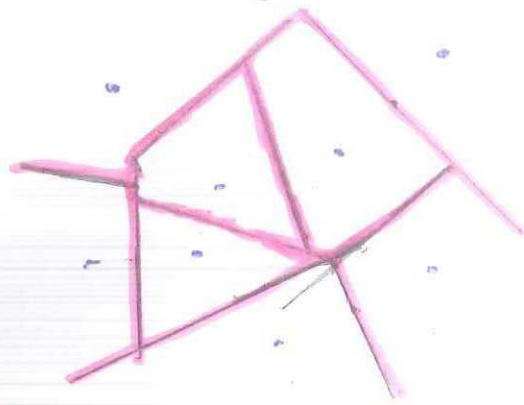
- Inverse distance Interpolatie

form.

$$h_x = \frac{1}{\sum \frac{1}{d_i^p}} \cdot \left( \frac{h_1}{d_1^p} + \frac{h_2}{d_2^p} + \dots + \frac{h_n}{d_n^p} \right)$$

h = hoogte  
 d = afstand tot x  
 d<sub>i</sub> = som afst. tot x  
 p = power

- Voronoi diagram



- middenloodlij - om alle punten  
Deze lopen dood op andere midden

- ~~---~~

→ geschikte covariatie, omth van eenheden, -1 tot 1

- Correlatie = mate waarin punten beïnvloed wordt door omringende punten.

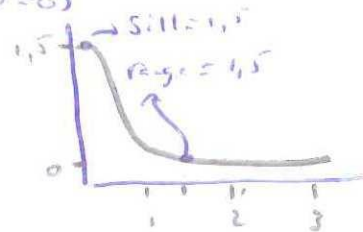
→ afhankelijk van eenheden, niet geschikt

- covariatie = mate waarin deze beïnvloed zijn, kracht.

↳ berekenen: functie (f(x)), ΔxΔy invullen → cov

- Sill = max cov, range = einde van cov. (cov=0)

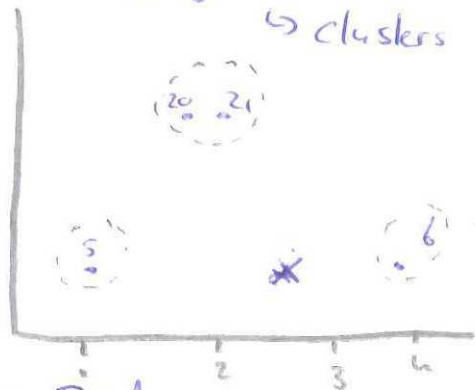
↳ ΔxΔy > 1,5 punt is niet geïsoleerd.



- kriging mean.

↳ clusters maken!

3-clusters



$$k_m = \left(\frac{1}{3} \cdot 5\right) + \left(\frac{1}{3} \cdot 6\right) + \left(\frac{1}{6} \cdot 20\right) + \left(\frac{1}{6} \cdot 21\right) = 10\frac{1}{2}$$

- Redundancy matrix is inverteerbaar

Simple

omdat opgesteld wordt adhv alle maar unieke cov-functie die OK waarde geeft.

$C_{ii}$  = covariatie tussen

$P_i$  en  $P_i$  (=0)

$f(0) = cov$

$$\begin{pmatrix} C_{11} & C_{12} & C_{13} & C_{14} \\ C_{21} & C_{22} & & \\ C_{31} & & C_{33} & \\ C_{41} & & & C_{44} \end{pmatrix}$$

$$\begin{pmatrix} C_{11} & C_{12} & C_{13} & C_{14} & 1 \\ C_{21} & C_{22} & & & 1 \\ C_{31} & & C_{33} & & 1 \\ C_{41} & & & C_{44} & 1 \\ 1 & 1 & 1 & 1 & 0 \end{pmatrix}$$

$C_{12}$  = covariatie tussen  $P_1$  en  $P_2$  (=3)

$f(3) = cov$

Simple = -geen trend

- random veld/punt

- random hoogte

OK - respecterend hoogte

## Proximity vector

$$\begin{pmatrix} C_{x_1} \\ C_{x_2} \\ C_{x_3} \\ \vdots \\ C_{x_n} \end{pmatrix} \rightarrow C_{x_i} = \text{covariatie tussen } x \text{ en } p_i \\ \text{enzv.}$$

- Totaal:

→ Is inverteerbaar omdat:

$$\begin{pmatrix} \text{cov.} \\ \text{redundancy} \\ \text{matrix} \end{pmatrix} \cdot \begin{pmatrix} \text{ge} \\ \text{inf} \\ \text{ent} \end{pmatrix} = \begin{pmatrix} \text{proxi} \\ \text{mity} \end{pmatrix}$$

- Landsat

↳ Banden onderscheidt door kleur/golflengte

↳ resolutie pixel =  $30 \times 30 \text{ m}$

↳ false color image

- feature space zg - aantal dimensies

↳ 11 banden

↳ 4 geometrisch  $\Rightarrow$  15 dimensies

- Sensoren:

- GPS

- Radar

- Laser (LiDAR)

- GIS raster = Alle data in matrix, rijen + kolommen (coördinaten op x, y en hoogte in matrix)

- GIS vector = Alle data in punten of polygoon, (veel moeilijker te versuiken)

## - LIDAR

$R$  = range laser to object

$c$  = lichtsnelheid

$t$  = two way travel time

$$R = \frac{1}{2} \cdot c \cdot t$$

let op!

- belangrijk is - positie van laser

↳ Altitude, orientatie van laser

- stand van uitzichting

↳ head-roll-pitch

## - ellips

- x-y-z  $\rightarrow$  in coördinaten / graden,  $\varphi$ - $\theta$ - $R$   
↳ wgs

## - Cartesisch

- x-y-z waarde, zo af te lezen  
↳ UTM

## - Least Squares : $y = ax + b$

$y$  = vector van data/observaties, afstand

$A$  = model matrix, tijd

$x$  = vector of parameter

$e$  = vector of residuals or mistakes

↳  $\hat{x}$  = velocity = rijnsnelheid = stijgsnelheid =  $\frac{A \cdot y}{A^T A}$

$$A = \begin{pmatrix} 1 \\ 2 \\ 4 \end{pmatrix} \quad A^T = (1 \ 2 \ 4) \quad AA^T = 1 + 4 + 16 = 21 \quad \text{↳ official } \hat{x} = \frac{A^T \cdot y}{A \cdot A^T}$$

## - Slope

↳ monte carlo

= checker

met st. dev.

$$S = \frac{1}{2} \sqrt{(F-D)^2 + (B-H)^2}$$

↳ slope voor E

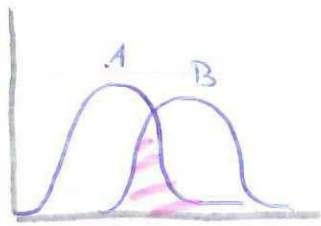
|   |   |   |
|---|---|---|
| A | B | C |
| D | E | F |
| G | H | I |



## nearest centroid classification

- ↳ mean van elke classe bepalen
- ↳ Om deze mean waarde Voronoi diagram
- ↳ Zo wilt je data in vak, nearest neighbour method

## - Linear Separation Problem



A = classe A

B = classe B

De overlap van A en B is de separation problem, je kan niet bepalen of het om classe A of B gaat.

## - (un)supervised

Supervised is zelf hande en classes bepalen  
un supervised is door programma laten doen

- fouten: ommissie =  $\text{bos} \neq \text{bos}$  foutief weggeleken  
Commissie =  $\text{bos} = \text{waaerwicht}$  foutief toegevoegd

↳ pixel foutief bos

## - Ransac

- ↳ lij - tussen 2 random punke
- ↳ bepaal residue tussen alle punke en deze lij
- ↳ kies standaard band maect om lij - hoe
- ↳ tel Inliers/outliers

Da genoeg lij - kies de hoogste score voor lij, meeste

Inliers

## - pixels - data

Correlatie kan gebruikt worden om van 14 data punten, in een pixel, naar 4 classes te gaan. Wanneer punten gecorreleerd zijn zijn ze beïnvloed door een ander punt. Je zoekt dus de niet gecorreleerde punten om de classe in dat pixel te bepalen.

## - Soort data:

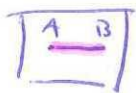
↳ Spectrale: Bos en water goed te herkennen, kleur komt in.

↳ Geometrische: zorgt voor duidelijk verschil in bergen en vlaktes.

## - Coördinaat systeem

↳ UTM, is om vervorming zo veel mogelijk te beperken. Deze zo klein mogelijk te houden.

↳ WGS 84, deze trekt de aarde recht, ziet de aarde als een vlak. Wanneer je afstand/lijntussen twee punten trekt en vervolgens verticaal zal deze niet kloppen doordat de aarde bol



loopt. Lijntje zal niet over surface lopen maar door ondergrond.