
Assimilation de
données

Analyse objective: méthodes de base

- Interpolation
- Régression
- Analyse par noyau
- Notion de covariance d'un champ
- ACP (analyse en composantes principales)

2. Analyse objective

analyse: représentation la plus exacte possible de la réalité dans un modèle à un instant donné.

analyse objective: selon des critères automatisés et généraux:

- contraintes physiques du système modélisé: continuité spatiale des champs, valeurs admissibles (concentration>0, $q<q_{sat}$...)
- propriétés statistiques: variabilité, corrélations spatiales...
- équations d'évolution dans le temps (physiques ou statistiques)

critère d'optimalité: définir "analyse la plus exacte possible"

Illustration simple : analyse 1D

Notations souvent utilisées dans ce cours:

- coordonnée d'espace *i* sur un intervalle
- *p* observations: valeurs y_j d'un paramètre à analyser à un instant donné, à des positions *j* quelconques
- "état de modèle": valeurs x_i sur une grille dont les points sont indicés par *i=1...n*

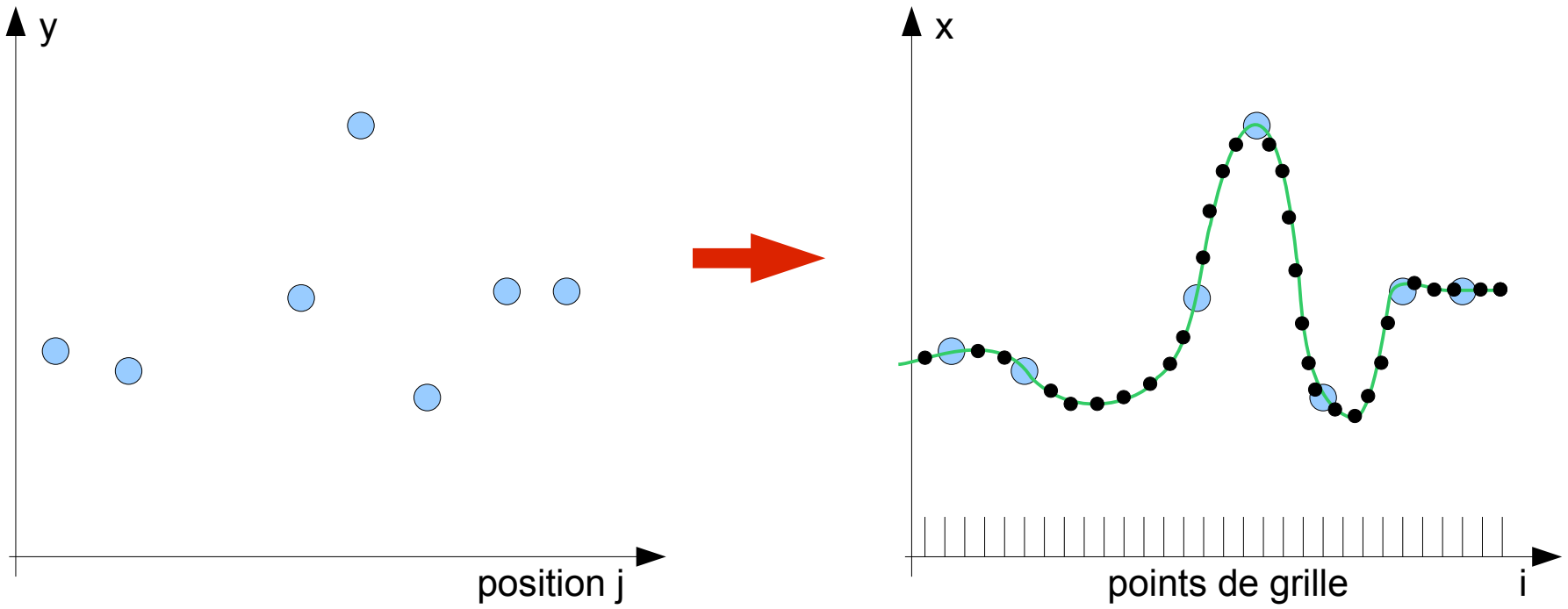
<u>problème:</u> calculer les x_i qui *tiennent compte des observations* et *respectent au mieux des contraintes fixées a priori*

Se généralise facilement à des modèles 2D ou 3D et à plusieurs variables.

Analyse de données par interpolation

Si les variables à analyser sont les variables observées, 'analyser' revient souvent à spatialiser par interpolation:

- les obs y_j sont les points à interpoler
- on veut une valeur interpolée x_i en chaque point de grille i du modèle



Analyse de données par interpolation

Technique très simple: on construit l'analyse *localement* avec des morceaux de fonctions choisis a priori et ajustés aux observations

Du plus simple au plus compliqué:

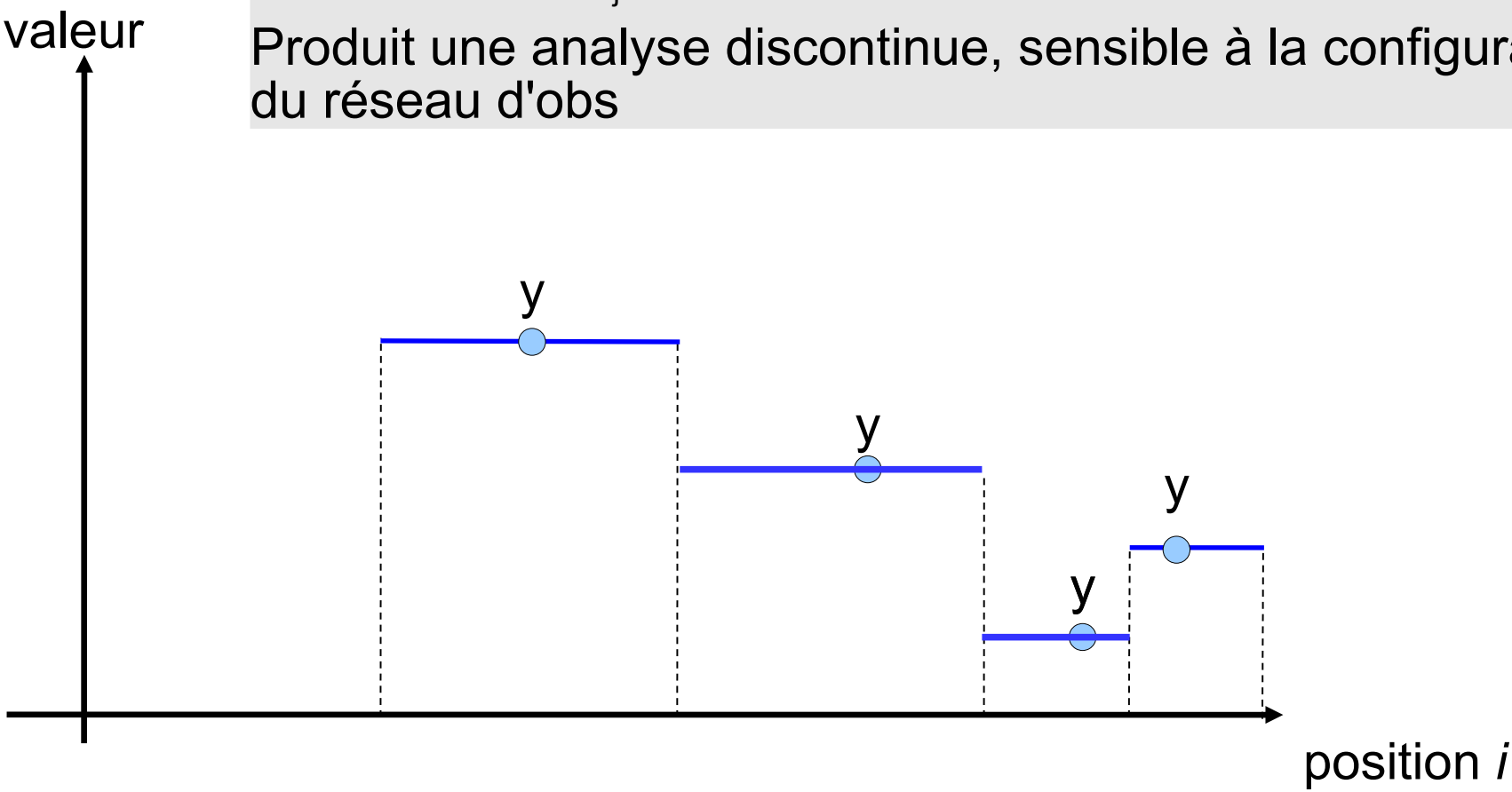
1) **plus proche voisin:** analyse=observation la plus proche.
2) **interpolation linéaire:** l'analyse=fonction linéaire par morceaux qui passe par les obs encadrant le point du modèle considéré.
3) **spline**: polynômes par morceaux
4) **interpolation par une famille de fonctions** plus générale, choisie par l'utilisateur
5) **régression** (linéaire, etc.) = analyse avec tolérance d'écart aux obs
5) idem en **2D, 3D, en multivarié** (ex. analyse de vent, de courant)
6) **méthode des noyaux:** technique simple et pratique
7) définition automatique des fonctions interpolantes: **variogramme**
8) **utilisation d'une ébauche**

Analyse par le plus proche voisin

Pour chaque point i:

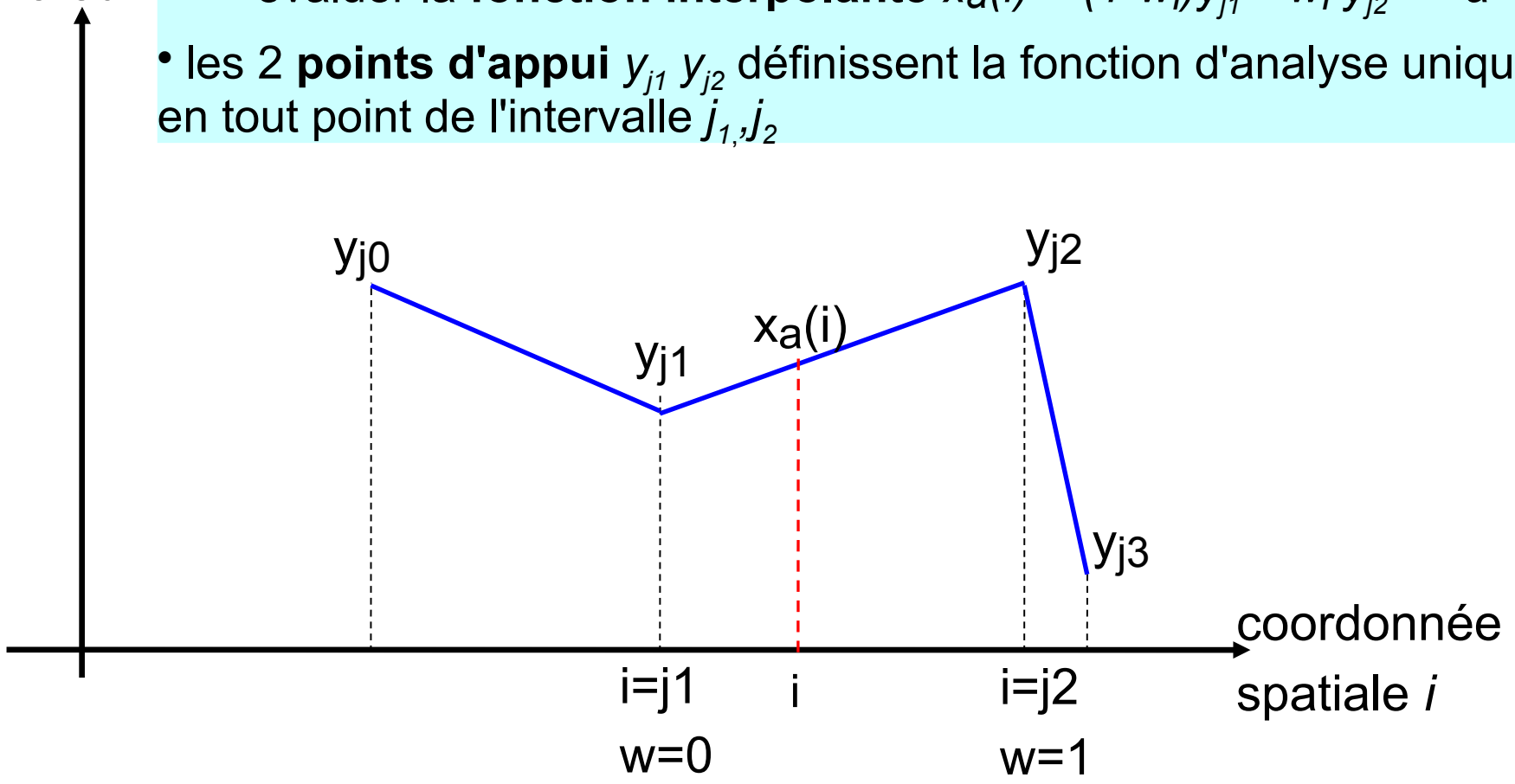
- **trouver l'obs** y_j la plus proche
- définir $x_i=y_j$

Produit une analyse discontinue, sensible à la configuration du réseau d'obs



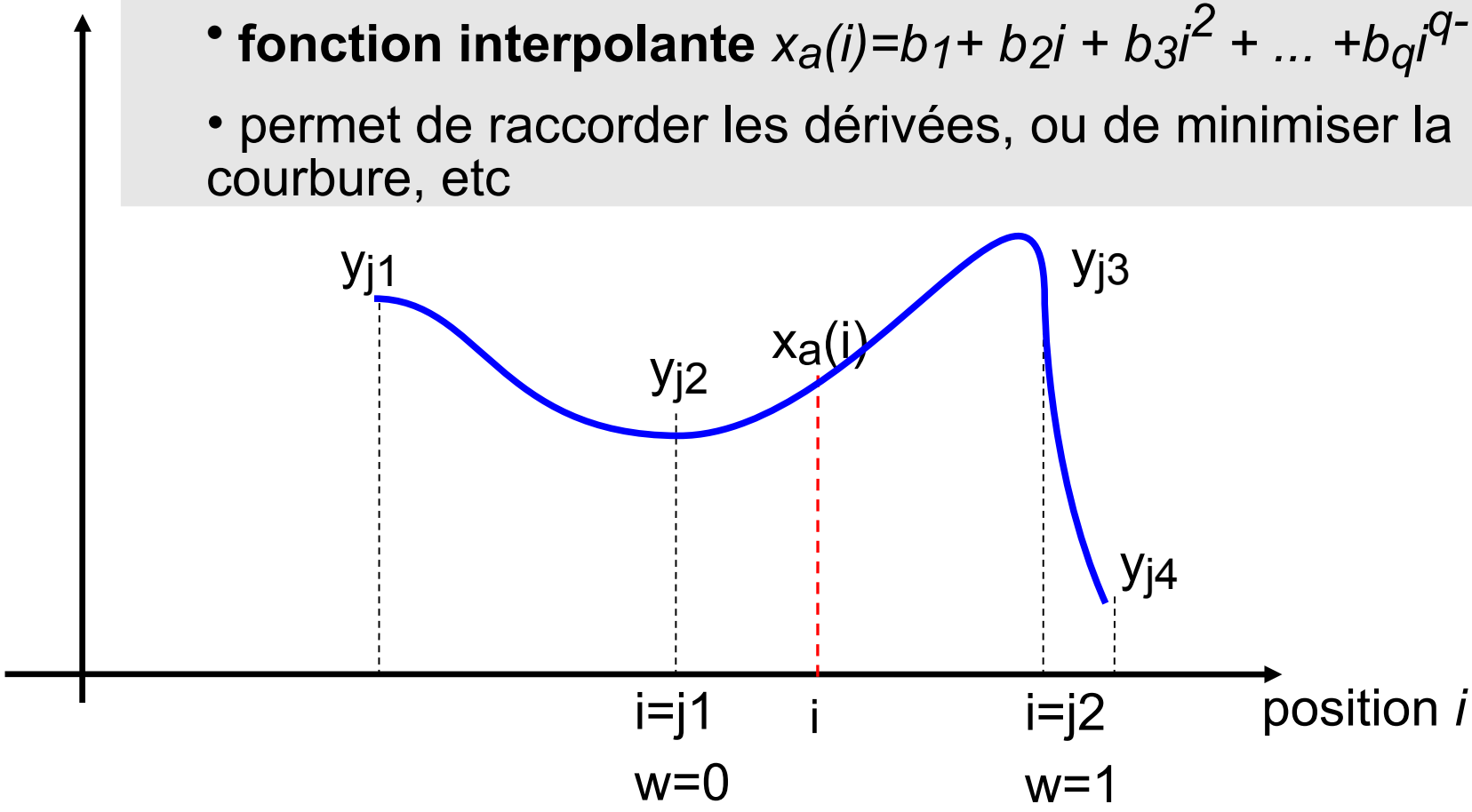
Analyse par interpolation linéaire

- Analyse en chaque point i:
  - **sélectionner** les 2 obs y_{j1} et y_{j2} qui encadrent i
  - calculer le **poids** $w_i=(i-j_1)/(j_2-j_1)$
  - évaluer la **fonction interpolante** $x_a(i) = (1-w_i)y_{j1} + w_i\, y_{j2} \quad = a+bi$
- les 2 **points d'appui** y_{j1} y_{j2} définissent la fonction d'analyse unique en tout point de l'intervalle j_1, j_2



Interpolation cubique (ou spline, ou autre polynôme)

- Analyse en chaque point i:
  - **sélectionner** les *q* obs y_{j1} ... y_{jq} qui encadrent i
  - calculer les **coefficients** $b_{1..q} = f(y_{j1} \ldots y_{jq})$
  - **fonction interpolante** $x_a(i)=b_1+ b_2 i + b_3 i^2 + \ldots + b_q i^{q-1}$
  - permet de raccorder les dérivées, ou de minimiser la courbure, etc



Interpolation par des fonctions génériques

1) Prédéfinir une famille de *fonctions interpolantes* $F_b(i)$ où *b* est une liste (un vecteur) de paramètres ajustables. Exemples:

- *polynômes dont b sont les coefficients*
- *fonctions sinus/cosinus: phase et amplitude*
- *composantes principales du climat d'un modèle*

2) Dans chaque région, choisir les ***points d'appui*** $\{y_i\}$ à utiliser pour définir la fonction interpolante $F_b(i)$

3) Définir un *critère de qualité* pour la fonction interpolante $F_b(i)$ dans cette région. Exemple: erreur quadratique moyenne:

$$\textbf{b } \textit{doit minimiser la fonction } J(b) = \sum_j \left(F_b(j) - y_j \right)^2$$

4) Calculer le b optimal (= résoudre un système d'équations en b, par exemple J'(b)=0)

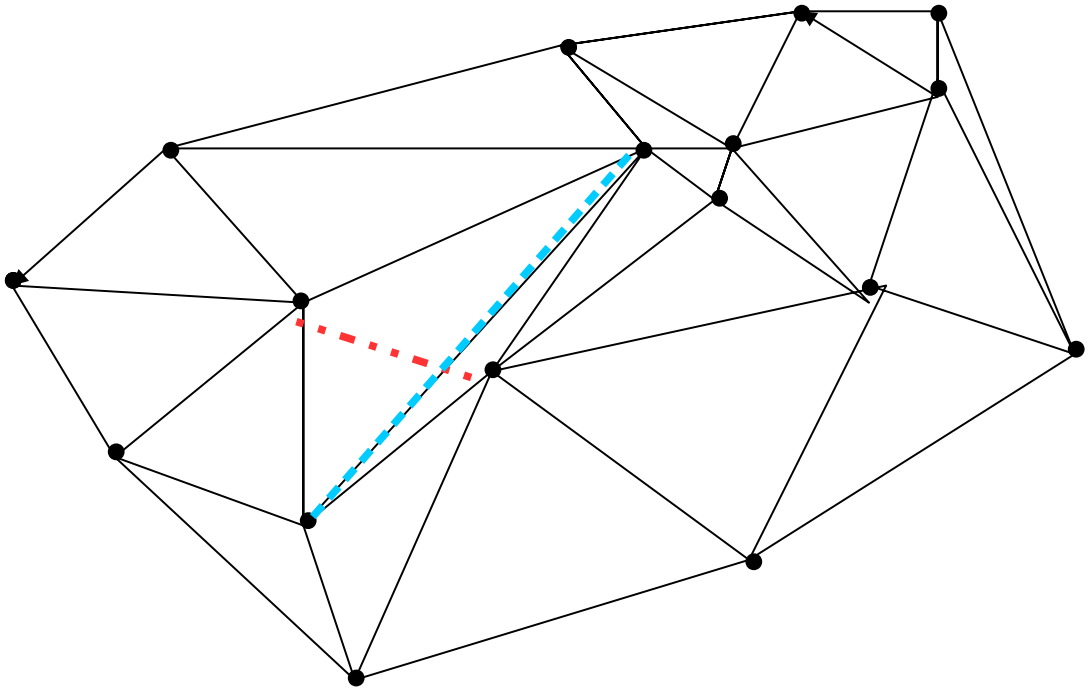
5) L'analyse dans cette région est définie par $x_a(i)=F_b(i)$

Généralisation en 2D et 3D

i devient une coordonnée dans le plan ou l'espace

le choix de bons points d'appui pour définir $F_b(i)$ devient complexe si les obs sont irrégulières. Idéalement on veut 'entourer' chaque région par quelques obs.

- ***algorithme KD-Tree*** *pour trouver efficacement les n points les + proches d'un point i*
- ***triangulation de Delaunay*** *pour définir des triangles aussi "compacts" que possibles*



la régression linéaire, une forme d'analyse

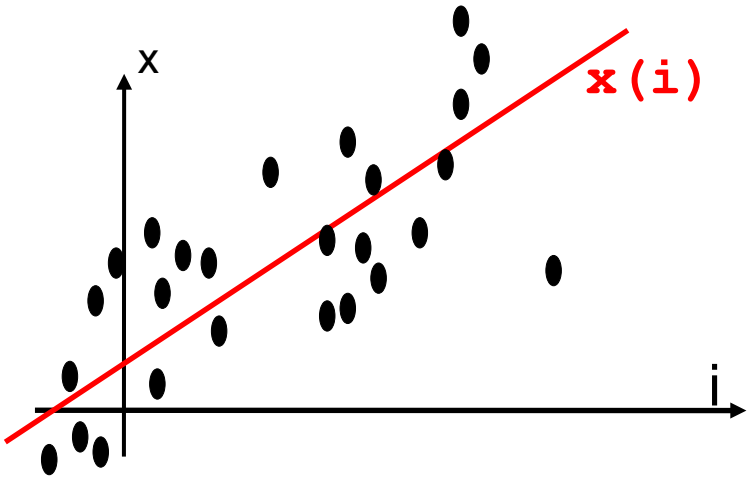
- on utilise partout toutes les observations y_i
- on cherche **une seule fonction linéaire** qui soit la plus proche possible des obs:

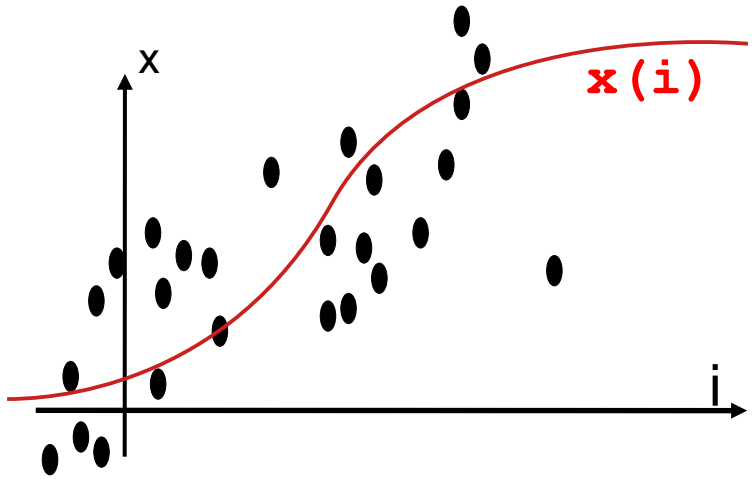
$$x_i = b_1 + b_2\, i$$

- le but est de trouver les coefficients (b_1,b_2) qui minimisent

$$J(b_1,b_2) = \Sigma_i\, (y_i - b_1 - b_2\, i\,)^2$$

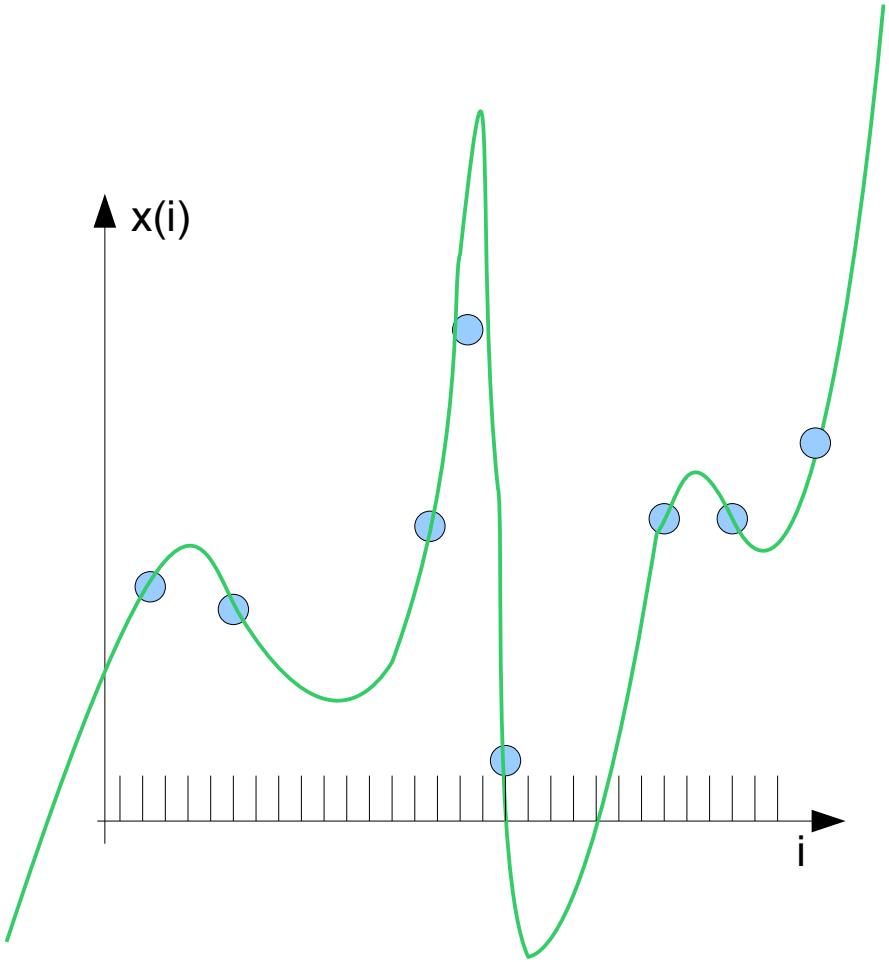
- <u>solution:</u> trouver (b_1,b_2) tels que la dérivée $J'(b_1,b_2)=0$
- on peut utiliser des fonctions plus complexes, ou modifier la coordonnée d'espace ou la variable analysée pour avoir un meilleur résultat (ex: Log(x)=f(Log i))
- en IA, l'**apprentissage supervisé** est une régression (souvent avec fonction th() ou linéaire par morceaux, appelée "neurones") adaptée au grands volumes de données ("big data")

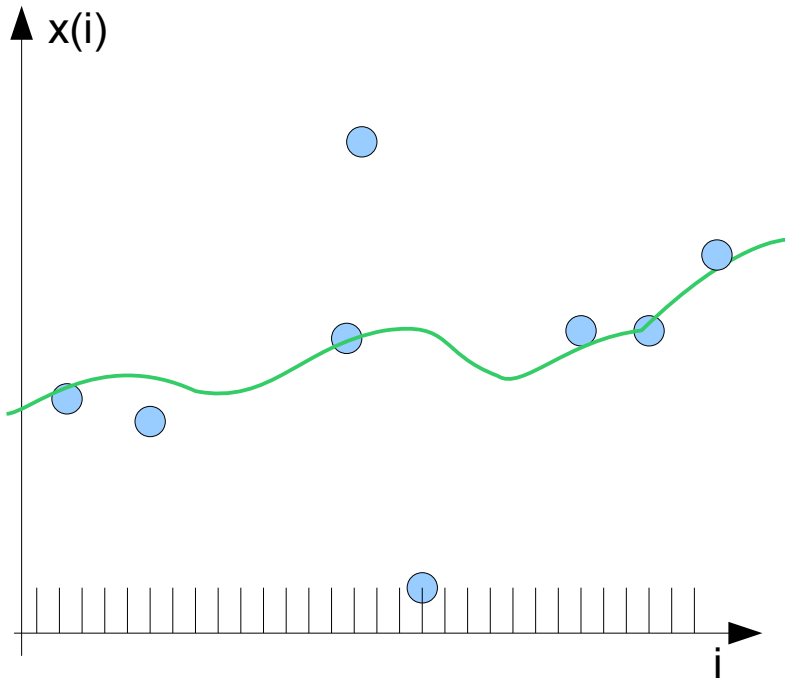




Problèmes d'interpolation

Dépassement (overshoot), surinterprétation des données (overfitting), lissage excessif (underfitting), extrapolation arbitraire à l'extérieur de la zone observée...





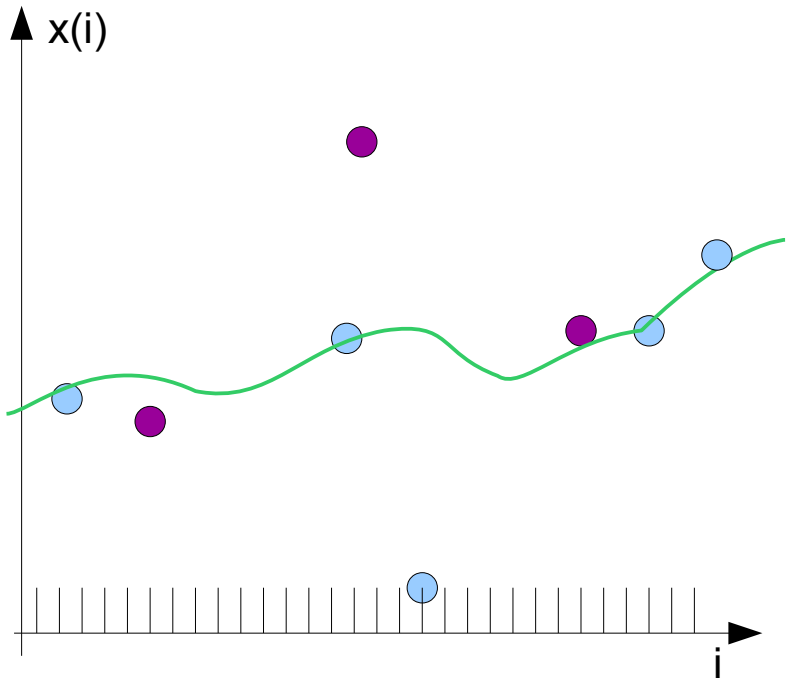
Validation d'une méthode d'analyse

Validation croisée: partitionner l'ensemble des obs en 2:

- *training set y* = échantillon d'apprentissage pour calculer la fonction interpolante F: min(|| F(x) - y ||^2)

- *validation set z* = échantillon de contrôle (non utilisé pour construire F) pour calculer la distance entre F et les obs non utilisées

erreur: || F(x) - z ||^2



Inconvénients des interpolations géométriques

- que faire s'il n'y a pas d'obs à proximité du point analysé ?
- que faire au bord du domaine ? extrapoler les obs ?
- on ne peut pas passer exactement par des obs à la fois proches et en désaccord mutuel: un compromis est nécessaire

- la sélection des points d'appui est difficile en 2D ou 3D
- coût calcul élevé si fonctions interpolantes complexes
- l'analyse dépend beaucoup de la densité locale du réseau d'obs

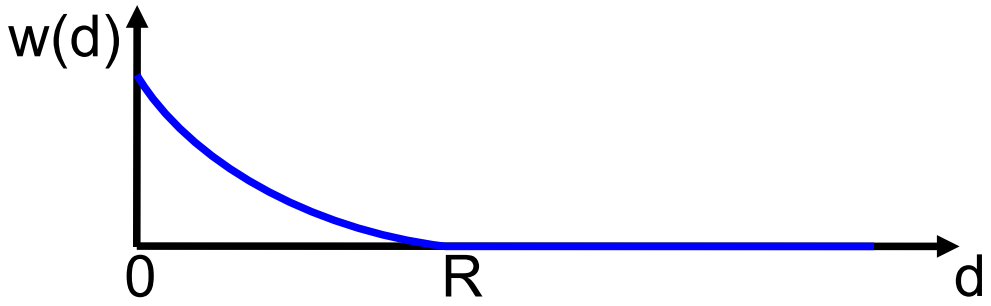
- l'interpolation est utile si on a beaucoup d'observations disposées régulièrement.
- la régression est utile si l'on sait à l'avance que l'analyse x doit appartenir à la famille de fonctions choisie.

Autre méthode de spatialisation: analyse par noyau

= RBFN (Radial Basis Function Neural Network), Nadaraya-Watson kernel regression, analyse de Cressman

•Prédéfinir une fonction poids interpolante *w(d) (noyau)*, fonction de la distance *d* entre obs et points du modèle:

*ex: $w=max(0, 1 - d^2/R^2)$ où R='**rayon d'influence**' fixé*



•En chaque point i, à partir de toutes les observations $\{y_j\}$ on **définit** la valeur analysée:

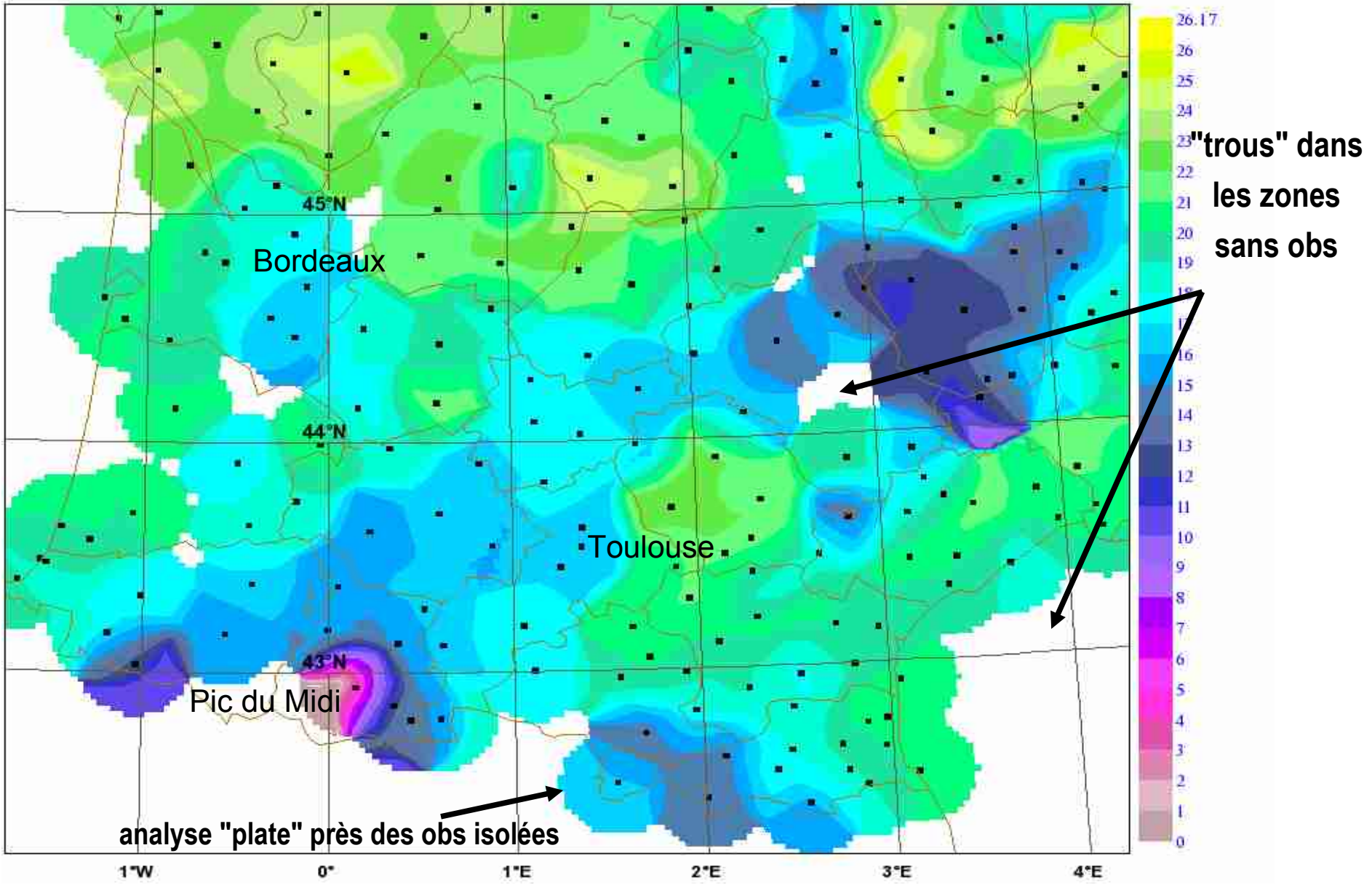
$$x_a(i)= \Sigma_j [\ w(d_{ij})\ y_j\ ] /\ \Sigma_j\ w(d_{ij})$$

– c'est une moyenne pondérée des obs avoisinantes.

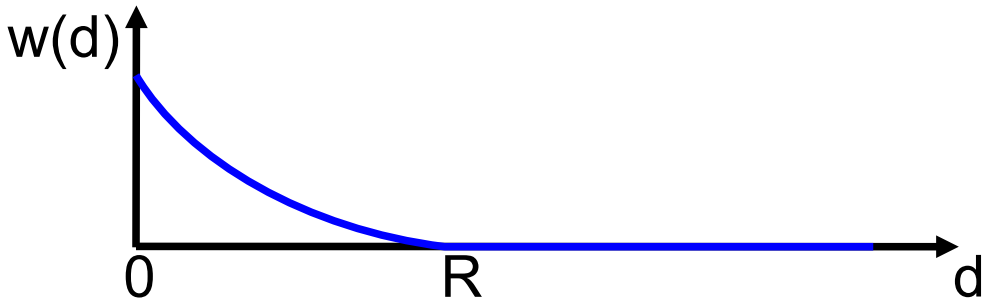
• l'analyse est indéfinie si $\Sigma_j\ w(d_{ij})=0$. (= trop loin des obs)

exemple d'analyse par noyau 2D

(obs de température de l'air, mesurée aux points noirs)



Problème: comment définir la fonction de poids ? avec des statistiques



•*fonction arbitraire* $w=max(0, 1 - d^2/R^2)$

$$x_a(i)= \Sigma_j [w(d_{ij}) y_j] / \Sigma_j w(d_{ij})$$

- Intuitivement, la finesse de l'analyse (donc R) devrait dépendre des **échelles caractéristiques** du champ analysé (ici T2m)
- Comment calculer cette échelle ? Par analyse statistique de **corrélations spatiales dans un historique** (d'obs ou de modélisations) représentatif des situations passées.
- On peut aussi calculer ainsi toute la fonction w: c'est le *variogramme* du champ (la méthode s'appelle *krigeage*)

Notions mathématiques nécessaires dans la suite du cours

Pour modéliser statistiquement les variations des champs et de leurs erreurs

- **statistiques** : représentation des variations moyennes, à partir de données d'apprentissage (historique d'obs ou de modélisations)
- **covariances** : résumé des statistiques des champs, et surtout de leurs **erreurs**
- **calcul matriciel** : représentation algébrique des covariances.

Lorsque l'on représente un champ à analyser par un vecteur d'état x (de dimension n), sa covariance est une matrice carrée (dimension $n \times n$).

Utilité des covariances d'erreur

- pour exprimer notre incertitude sur des observations et des champs de modèle, ce qui permet:

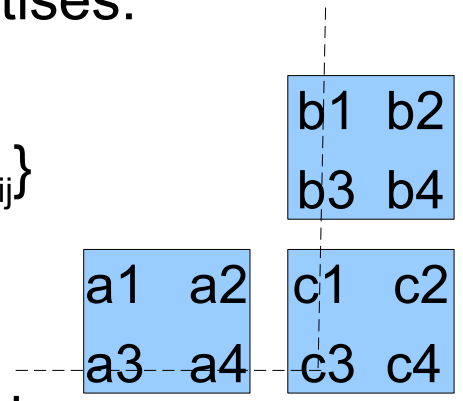
- de trouver l'analyse qui soit statistiquement la meilleure possible, compte tenu de la qualité limitée des données disponibles (observations et prévisions de modèles)

- de combiner facilement des informations multiples lors du calcul de l'analyse (ex: erreur d'obs + erreur de prévision pour combiner une obs avec une prévision récente)

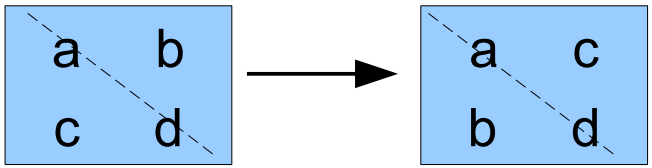
- d'estimer la qualité des analyses produites

rappel math 1: calcul matriciel

Indispensable pour les calculs sur des champs discrétisés:



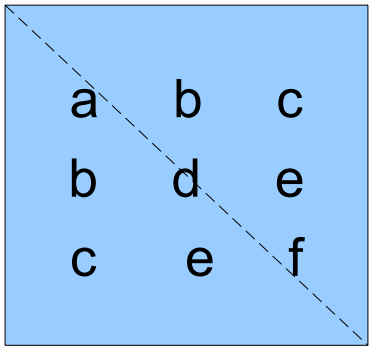
- **matrice:** tableau 2D de coefficients ex: $A = \{a_{ij}\}$
- **vecteur:** = matrice à 1 colonne
- **nombre réel:** = matrice 1x1
- **multiplication matricielle:** C=AB de coeffs $c_{ij}=\Sigma a_{ik}b_{kj}$
  - A doit avoir autant de colonnes que B a de lignes
  - l'action de A est assimilable à une application linéaire
- **addition** A+B et multiplication scalaire kA : idem sur les coeffs
- matrice **identité** I : carrée avec 1 sur diagonale, zéro ailleurs
  - $AI=IA=A$
- matrice **inverse** A^{-1} : $A^{-1}A=AA^{-1}=I$
  - $(AB)^{-1}=B^{-1}A^{-1}$
- matrice **transposée** A^T: avec lignes et colonnes permutées
  - $(AB)^T=B^TA^T$



rappel math 2: calcul matriciel

Matrices symétriques définies positives: importantes pour manipuler des statistiques d'erreur (covariances)

- **matrice symétrique:** $A^T=A$
- **matrice symétrique définie positive:** telle que $x^T Ax > 0$ pour tout x non nul
- une telle matrice est inversible et diagonalisable, c'est à dire qu'il existe M inversible et D diagonale telles que $A = M^T D M$
- **les matrices de covariance d'erreurs sont symétriques définies positives** (si toutes les variances sont non nulles)



rappel math 3: moyenne, variance, covariance, corrélation

Soit des réels $(a_k)_{k=1...K}$ et $(b_k)_{k=1...K}$: K "réalisations" des variables a et b

- **moyenne** de a : $\bar{a} = \Sigma_k a_k / K$
- **moyenne quadratique** de a (= *rms, root mean square*):

$$rms(a)=(\overline{a^2})^{1/2}= (\Sigma_k a_k^2 / K)^{1/2}$$

- **variance** de a: (version simple)

$$v(a)= rms^2(a - \bar{a}) = \Sigma [a_k- \bar{a}]^2 / K$$

- **écart-type** de a: $\sigma(a)=v^{1/2}(a)$
- **covariance** de a et b: *moyenne des produits débiaisés*

$$cov(a,b)= moy[(a- \bar{a}) (b- \bar{b})]$$

- **corrélation** de a et b: $\rho(a,b) = cov(a,b) / [\sigma(a)\sigma(b)]$
- formules utiles:

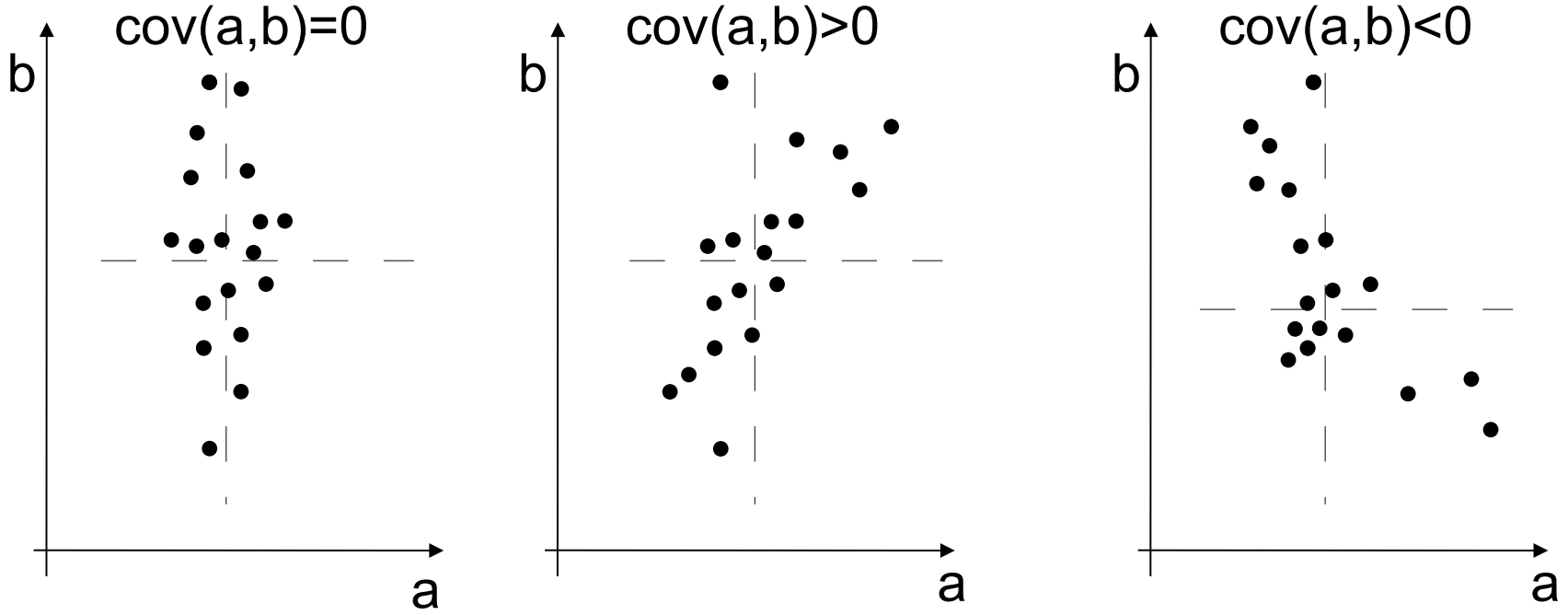
$v(a)=\overline{a^2} - (\bar{a})^2$ $\qquad$ $cov(a,b) = \overline{(ab)} - \bar{a}\,\bar{b}$

interprétation physique de la covariance entre 2 variables

Soit K réalisations des réels a et b: $(a_k)_{k=1...K}$ et $(b_k)_{i=1...K}$

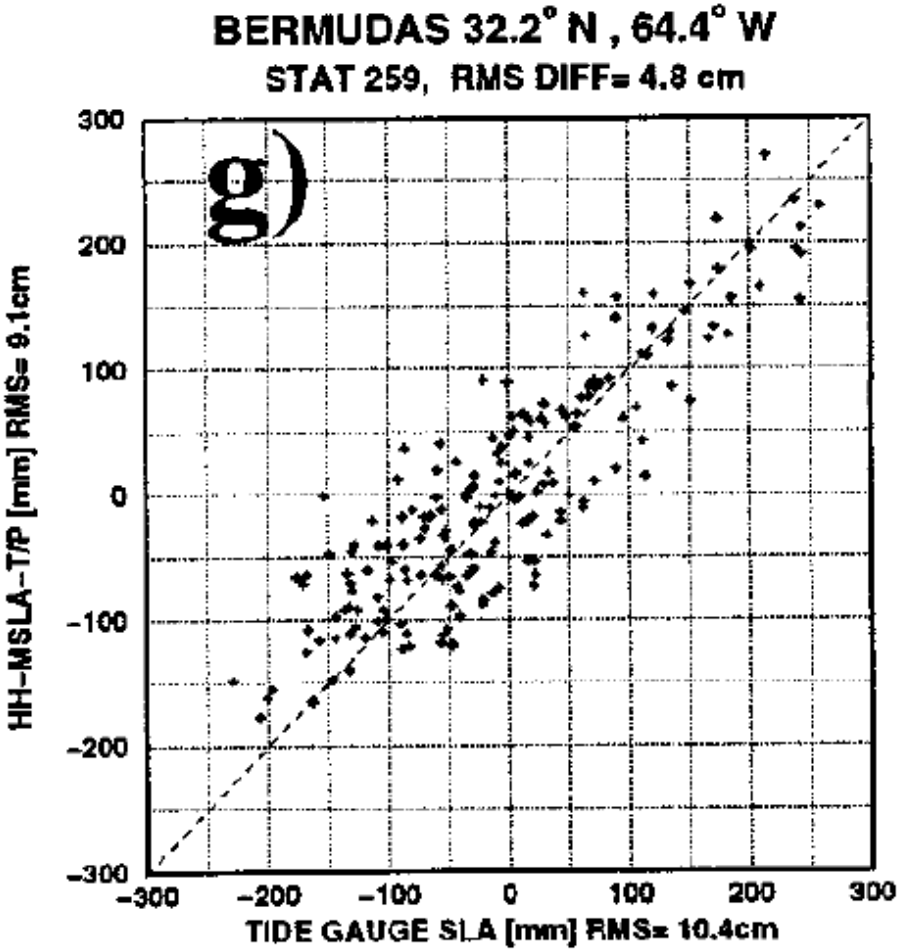
- **écart-type** de a: $\sigma(a)=v^{1/2}(a)$ mesure sa variabilité
- **covariance** et **corrélation** de (a,b) mesurent leur lien mutuel, par rapport au point moyen ($\overline{\mathbf{a}}$, $\overline{\mathbf{b}}$)

Regarder un nuage de points de coordonnées (a_k,b_k):



exemple d'analyse de corrélation entre 2 variables

hauteurs de la mer mesurées par satellite et par marégraphe
en un point, sur plusieurs mois:
elles "varient à peu près ensemble"



rappel math 4: vecteur aléatoire (=statistiques de champs)

Soit K **vecteurs** $(\mathbf{x_k})_{k=1...K}$, chacun de dimension n:

- chaque réalisation k a n composantes: $x_k=(x_k(i=1) \dots x_k(i=n))$
- chaque composante i a K réalisations: $x_{k=1}(i) \dots x_{k=K}(i)$
- ex: ensemble de plusieurs réalisations du champ x

Les opérations statistiques s'étendent aux vecteurs en les appliquant à leurs composantes:

moyenne d'un vecteur = vecteur $\bar{\mathbf{x}}$ dont les composantes sont les moyennes des K réalisations de x (idem pour variance, écarts-types)

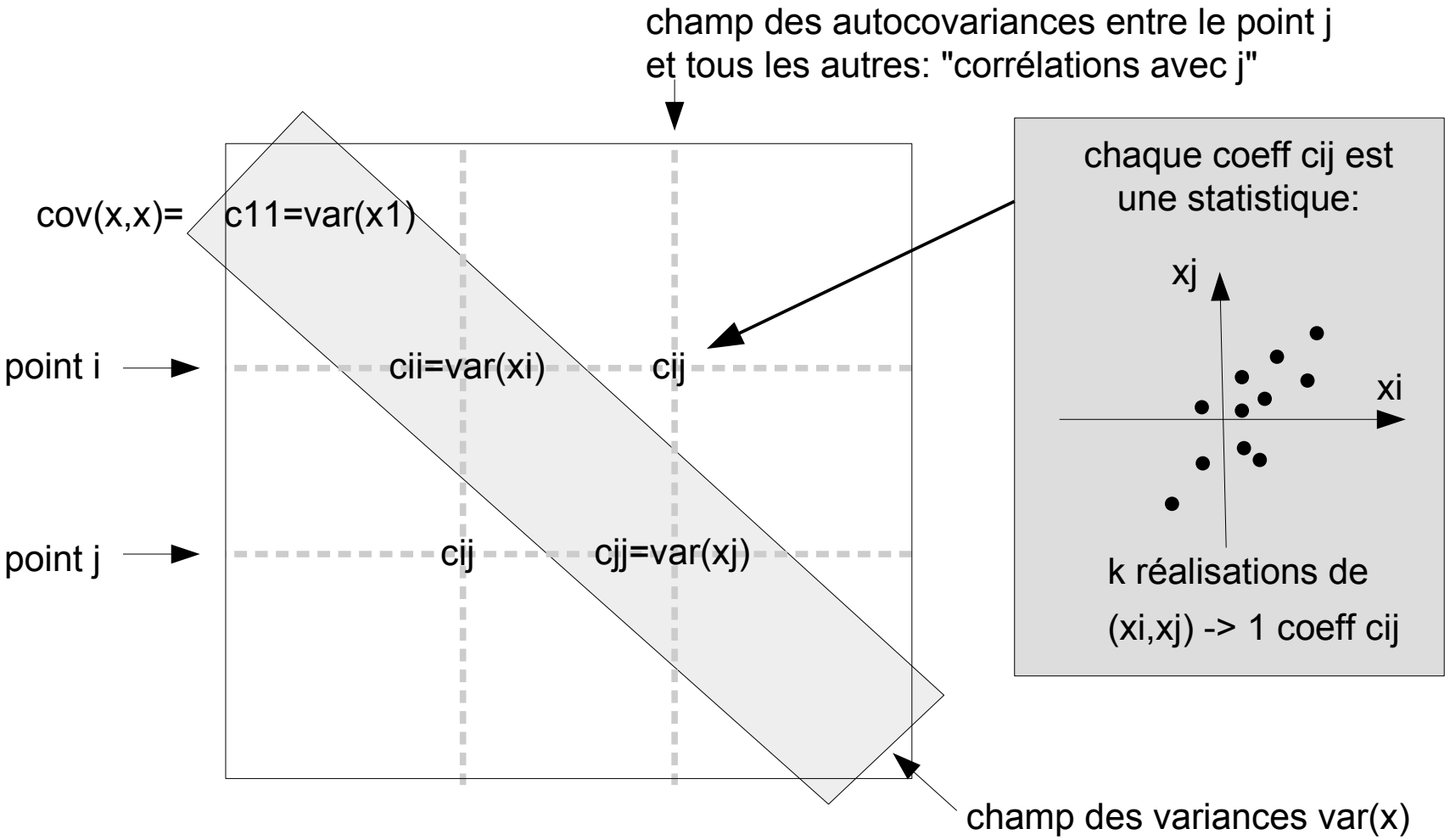
Mais: **la covariance** cov(x,y) entre 2 familles de vecteurs x,y est 1 ***matrice*** *dimx* **x** *dimy* dont les coeffs sont les covariances entre tous les couples de composantes de x et de y:

$$\mathbf{cov(x,y) = m[(x-\bar{x})\ (y-\bar{y})^T]}$$

•NB. ce sont des "moyennes d'ensemble" point par point, ne pas confondre avec des moyennes spatiales.

matrice d' (auto)covariance d'un champ

- dans ce cours on parlera surtout d'autocovariances, ex: cov(x,x) pour un champ x
- en ordonnant les points de x en un vecteur 1D, la covariance cov(xi,xj) est une matrice carrée symétrique: (on note xi=x(i) et *cij=cov(x(i),x(j))*)

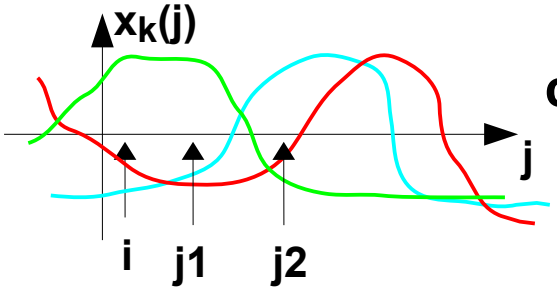


autocovariance d'un champ

- en géostatistique on utilise les autocovariances des champs manipulés, estimées à partir d'un **ensemble de réalisations du champ x** (historique, ensemble de prévisions...)

ex: "autocovariance du champ de température"

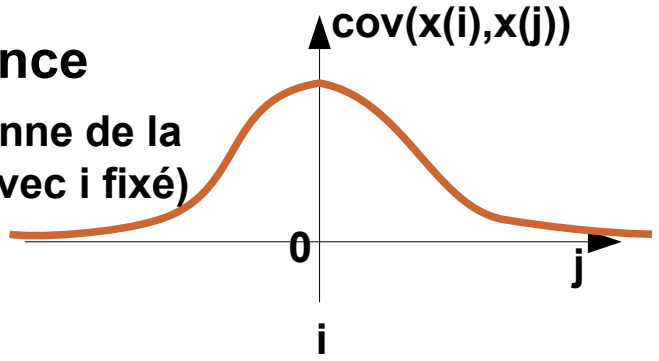
- l'autocovariance de x est définie pour **tous les couples de points** (i,j): cov(x(i), x(j)) = "cov(i,j)"
- elle indique si xi et xj varie plutôt ensemble, ou indépendamment
- souvent les champs des fluides sont plutôt **lisses: cov(i,j) est maximal si (i,j) sont proches**
- en termes de corrélations: cor(i,j) -> 1 si i -> j

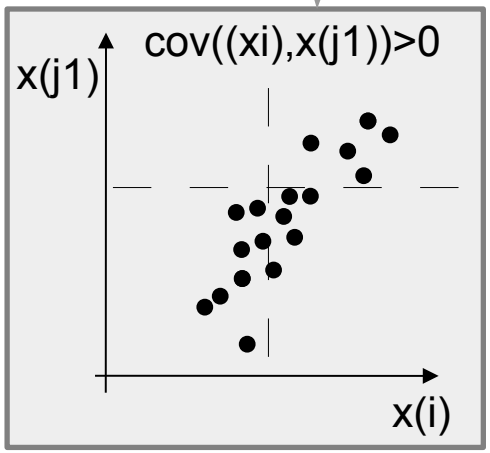


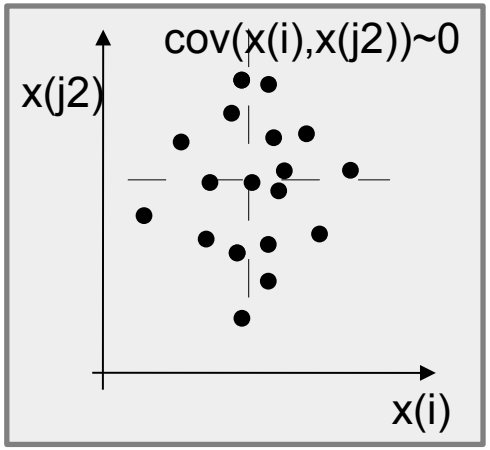
champs

covariance

(une colonne de la matrice avec i fixé)



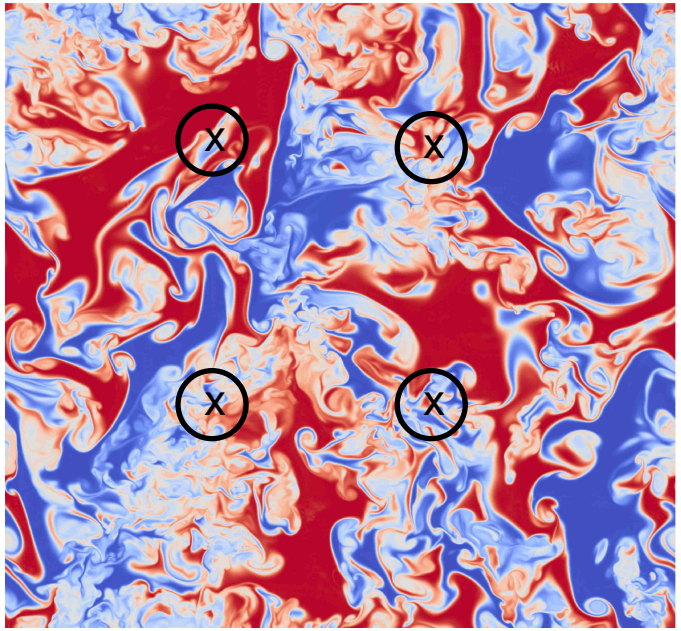




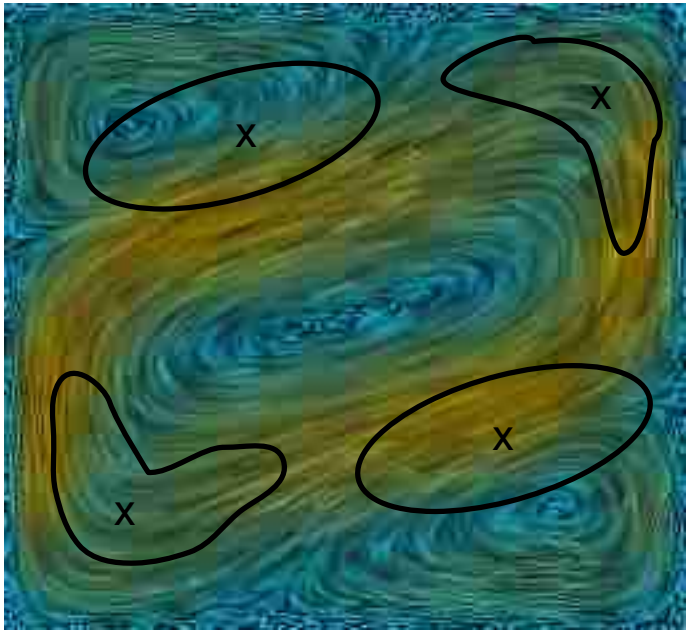
autocovariance d'un champ

- exemples: les autocorrélations caractérisent la texture locale du champ

isotrope, corrélations locales
= bcp de degrés de liberté



anisotrope, fortes corrélations
sur tout le domaine
= peu de degrés de liberté

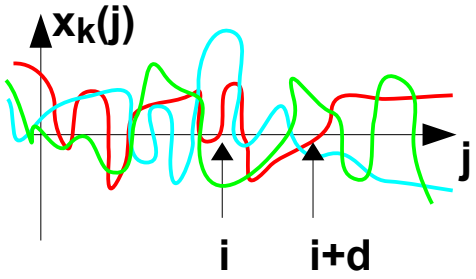


autocovariance d'un champ

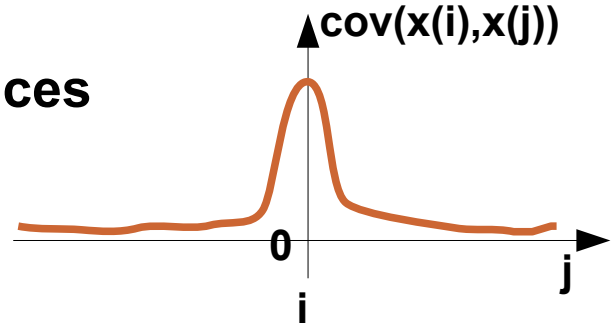
- **autocovariances**: cov(x,x) = covariances des couples de coordonnées (i,j): cov(xi , xj)
- souvent on regarde la moyenne en fonction de la distance: cov(d) = moyenne de tous les couples cov[x(i) , x(i+d)] dont la distance géométrique mutelle est *d*(i,j)=d *(variogramme)*

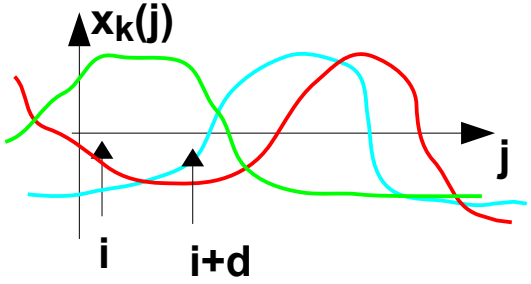
données nécessaires pour calculer des covariances:

- échantillon de champs x = climatologie historique, prévisions numériques, ou réanalyses
- taille d'échantillon: >10 pour une moyenne, >30-100 pour variances et corrélations

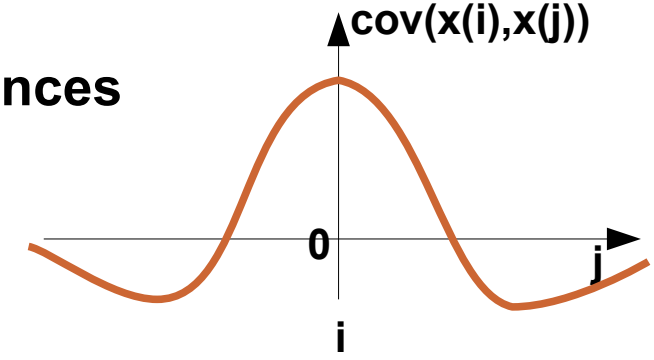


champs → covariances





champs → covariances



variogramme d'un champ

- **autocovariance générale**: cov(x,x) = covariances entre tous les couples de positions (i,j): cov(x(i) , x(j))

Pour améliorer l'échantillonnage on fait souvent des **hypothèses simplificatrices:**

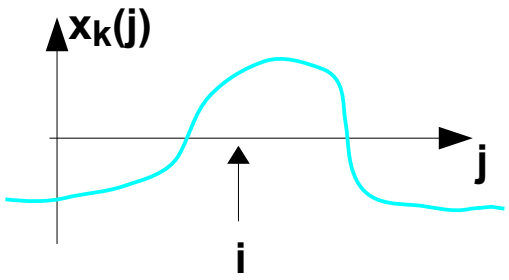
- **hypothèse d'homogénéité:** on suppose que la fonction j → cov(x(i) , x(j)) ne dépend plus de i, seulement de la position relative de i et j:

f(d) = cov(x(i) , x(i+d)) = moyenne des cov(x(i),x(j)) pour tous les couples (i,j) tels que le vecteur d=j-i

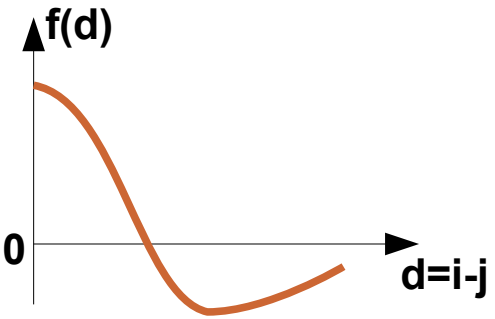
- **hypothèse d'isotropie:** f ne dépend que de la distance entre les points i et j, pas de la direction

Ces hypothèses facilitent les calculs statistiques mais ne sont valables que si le système physique représenté par x respecte ces hypothèses:

- souvent vrai en cas de turbulence libre
- souvent faux près des limites géométriques (parois, fronts) ou dans les fluides stratifiés (anisotropie)

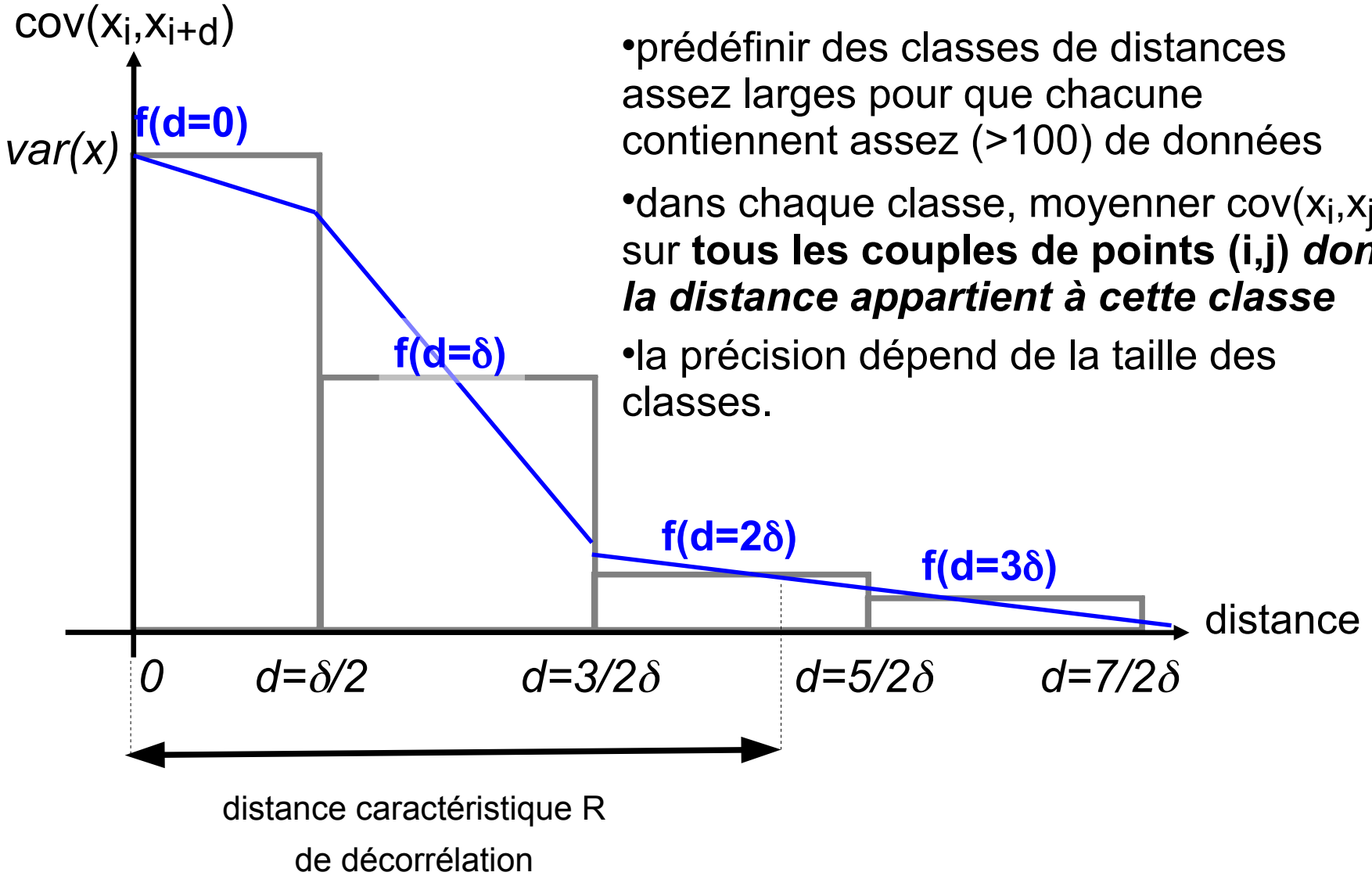


champ(s) → **variogramme**



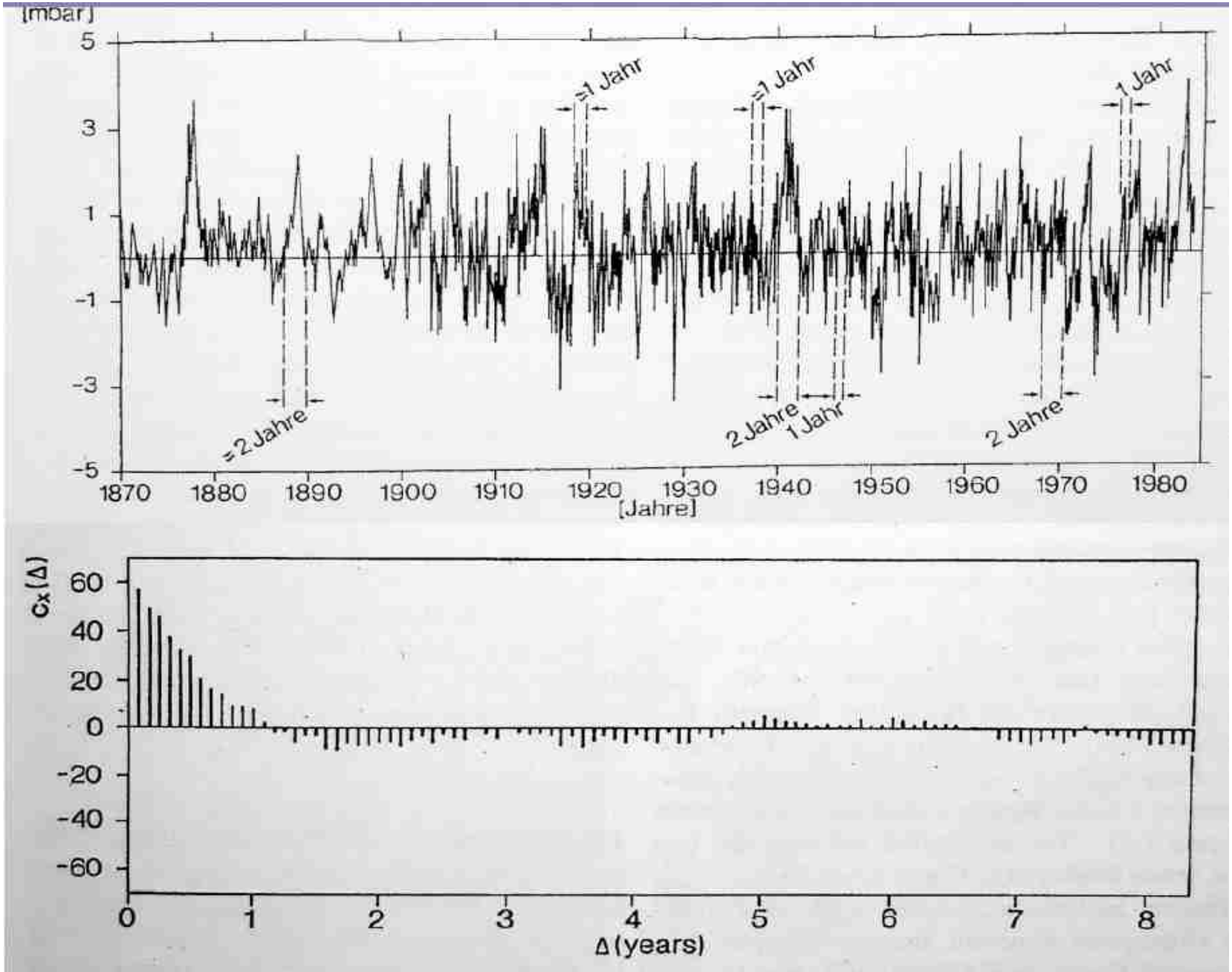
calcul statistique d'un variogramme:

c'est un **histogramme des cov(x_i,x_j) = f(distance d_{ij})**



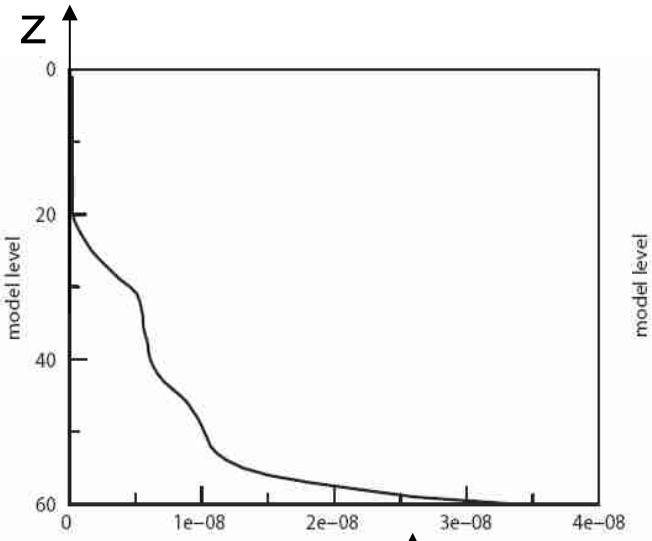
- prédéfinir des classes de distances assez larges pour que chacune contiennent assez (>100) de données
- dans chaque classe, moyenner cov(x_i,x_j) sur **tous les couples de points (i,j)** ***dont la distance appartient à cette classe***
- la précision dépend de la taille des classes.

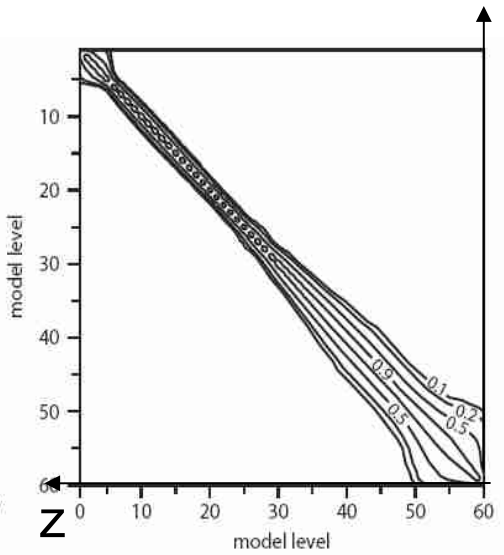
ex de variogramme (série temporelle d'indice ENSO)



Exemple de variances & corrélations 3D en chimie atmosphérique (CO)

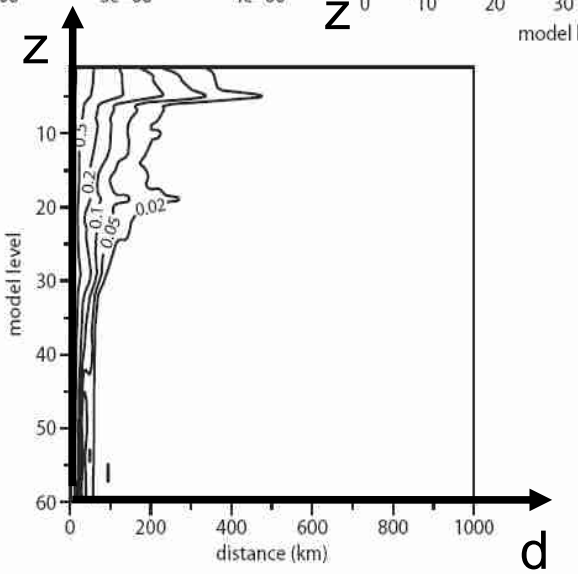
distribution verticale des variances





matrice des corrélations verticales

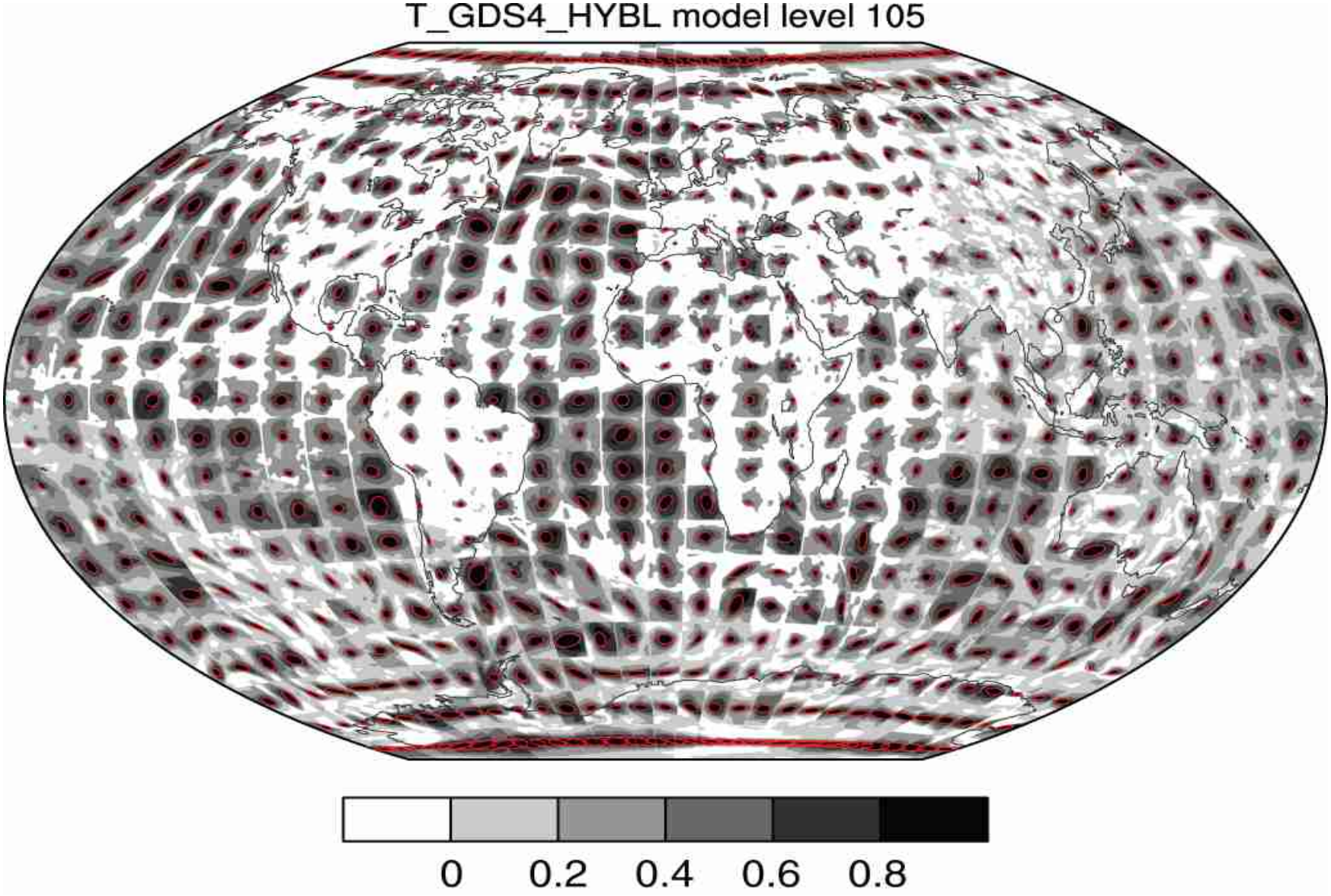
distribution verticale des corrélations horizontales (1 variogramme par altitude)



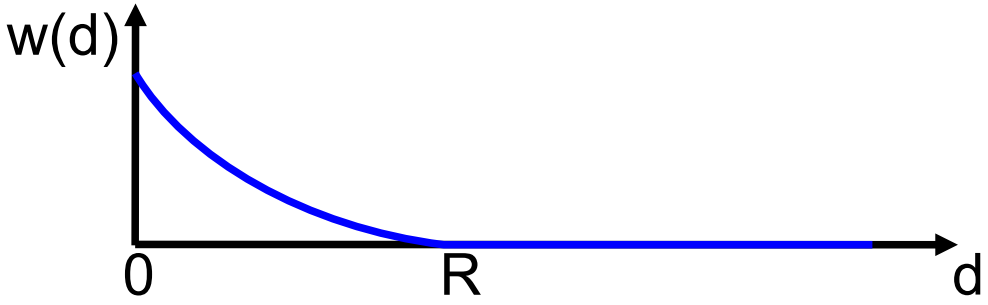
(calcul à partir d'un grand nombre de simulations 3D du champ de CO)

Figure 2: CO background error standard deviation profile (top left) in kg/kg, vertical correlations of CO background errors at (top right), and horizontal correlations of CO background errors (bottom) at 50° N, 10° E.

Corrélations météorologiques



Retour sur l'analyse par noyau (krigeage simplifié)



- on peut utiliser un variogramme pour estimer R ou même construire la fonction de poids dans l'équation:

$$x_a(i) = \Sigma_j \{w(d_{ij})\, y_j\} \,/\, \Sigma_j\, w(d_{ij})$$

- le variogramme peut être estimé directement à partir des obs si elles sont assez nombreuses
- produit une analyse dont les structures "ressemblent" à celles vues par les obs.
- (suite du cours: étendre cette idée en la combinant avec un modèle numérique, pour analyser les *écarts* entre obs et modèle)

Analyse en composantes principales (ACP)
Empirical Orthogonal Functions (EOFs)

- très utilisée en sciences de la Terre, notamment en climatologie
- idée:
  - identifier les modes de variabilité principaux d'un historique de champs (les EOFs)
  - projeter les champs dans le sous-espace défini par ces modes
  - *(c'est donc une analyse dans un sous-espace de fonctions définies statistiquement)*
- exemple:
  - l'ENSO est le mode de variabilité principal du Pacifique tropical
  - le climat tropical est influencé par la phase de l'ENSO où l'on se trouve (El Nino / La Nina)
  - on peut étudier l'influence de tels modes de variabilité à grande échelle sur les climats locaux (= il y a des sources naturelles d'inhomogénéité dans les séries de données climatiques)

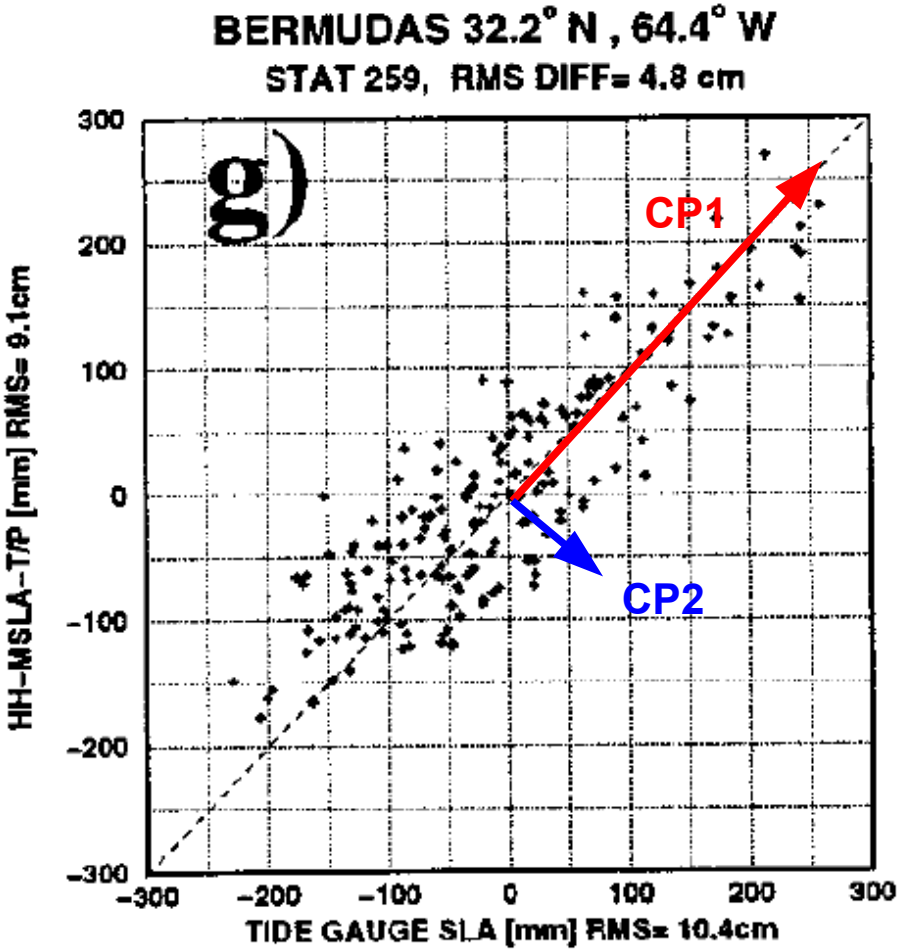
Analyse en composantes principales (ACP)
Empirical Orthogonal Functions (EOFs)

méthode:

- données = un ensemble de réalisations d'un champ (ou de séries temporelles, etc.) = K vecteurs $(\mathbf{x}_k)_{k=1...K}$, chacun de dimension n
- 1) calculer la **matrice d'autocovariance** A=cov(x,x)
- 2) la diagonaliser: $A = M^T D M$ (ou mieux: la décomposer en valeurs singulières)
- 3) **trier les valeurs propres** (diagonale de D) par ordre décroissant
- 4) garder les *p* plus grandes valeurs propres nécessaires pour expliquer (par ex.) 90% de la variance totale (=trace de A ou de D)
- 5) les lignes correspondantes de M (=vecteurs propres de A) sont les p **composantes principales**: ce sont les champs "qui varient le plus"
- 6) ils définissent une **base orthonormée** sur laquelle on peut:
  - projeter n'importe quel champ
  - voir sa coordonnée par rapport à chacune des composantes principales

Ex trivial: ACP avec 2 variables

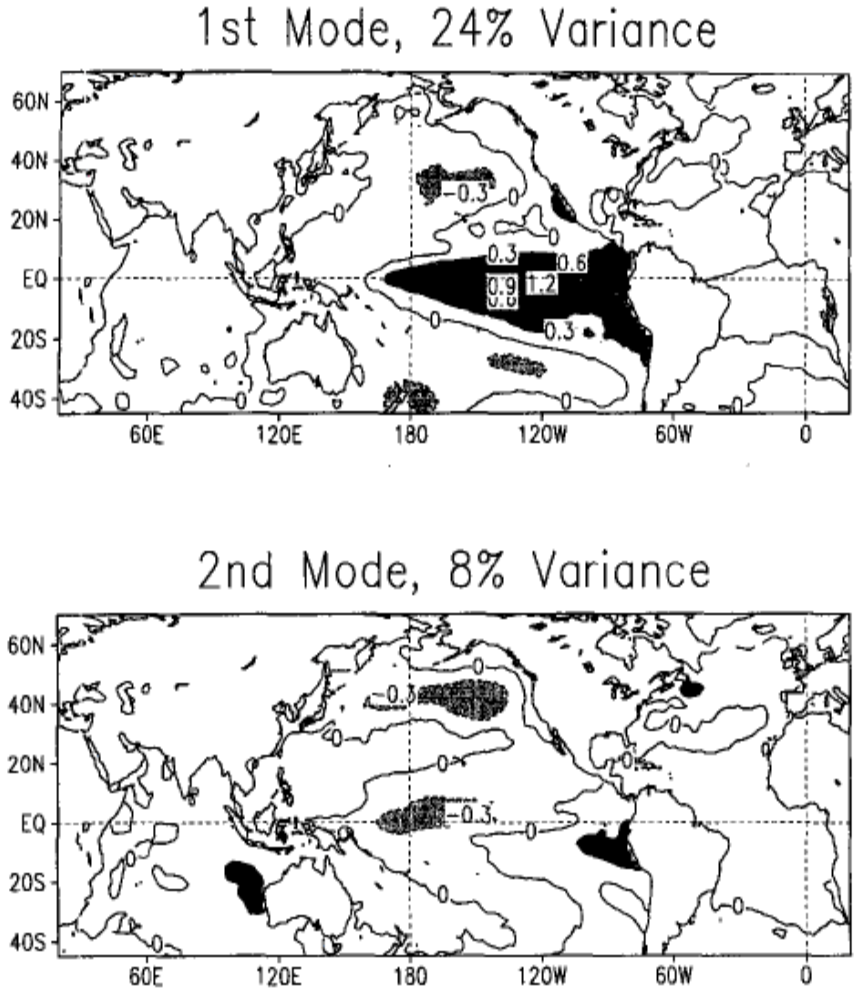
hauteur de la mer mesurée par satellite et par marégraphe

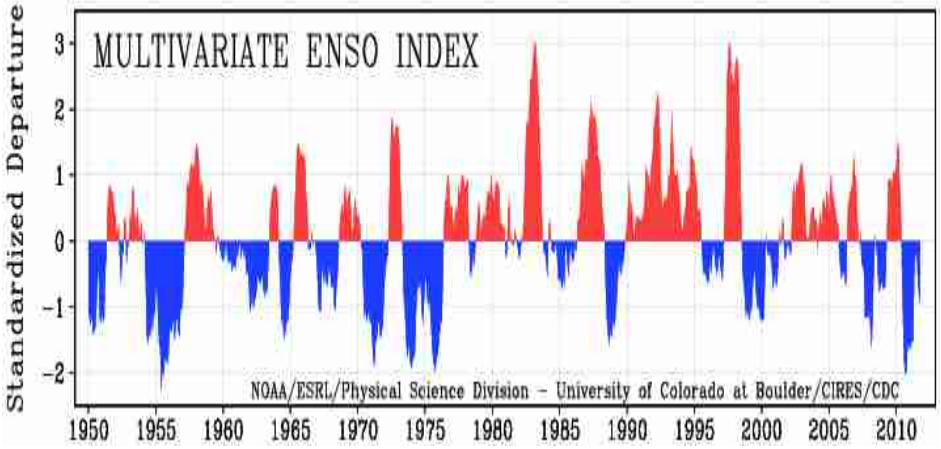


Ex moins trivial: analyse de la phase ENSO

température de surface de la mer

(Smith et al, J Clim 1996)





Merci pour votre attention