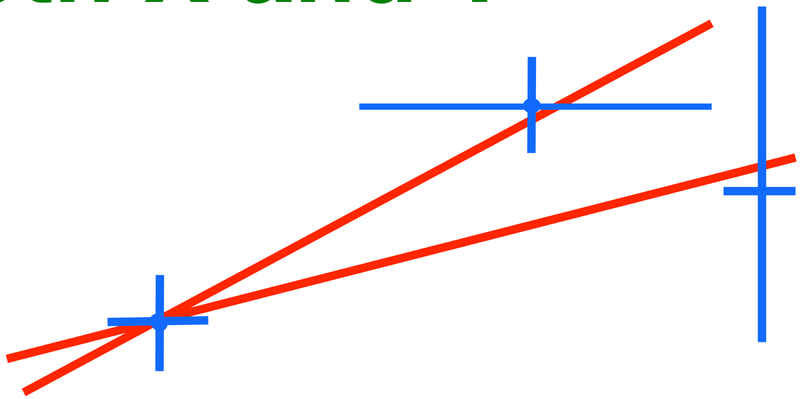
# Error Bars in both X and Y

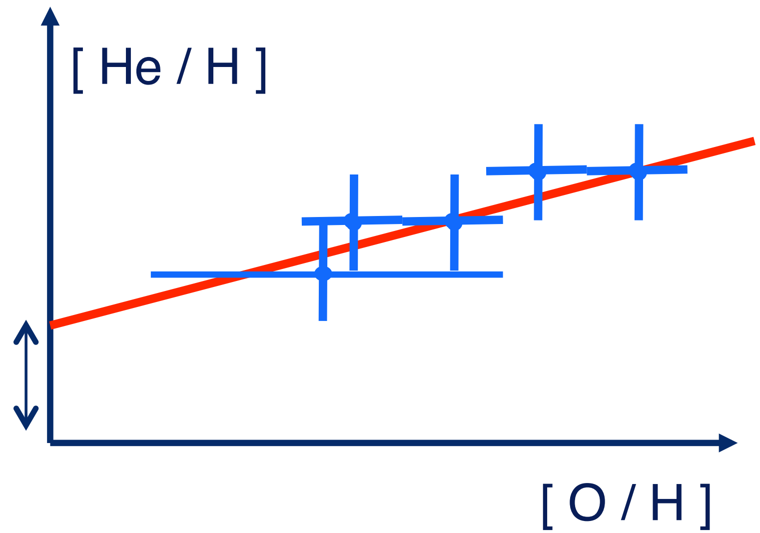
Wrong ways to fit a line:

1. $y(x) = a\,x + b \quad (\sigma_x = 0)$
2. $x(y) = c\,y + d \quad (\sigma_y = 0)$
3. split difference between 1 and 2.



Example: **Primordial He abundance:**

Extrapolate fit line to [ O / H ] = 0.




Correct method is to minimise :

$$\chi^2(a,b) = \sum_{i=1}^{N} \frac{\left(Y_i - (a\,X_i + b)\right)^2}{\sigma^2(Y_i) + a^2\sigma^2(X_i)}$$

***Let's see why.***

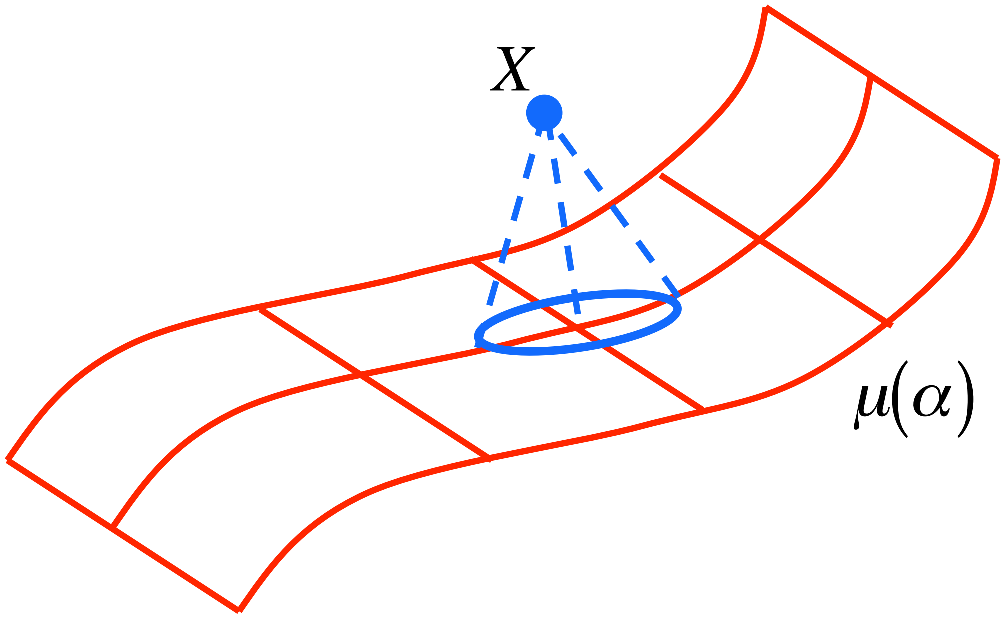
# Vector Space Perspective

$N$ data points, $M$ parameters.

Model $\mu(\alpha)$ defines a parameterised $M$-dimensional surface in the $N$-dimensional data space.

$\chi^2(\alpha)$ = squared distance from the observed data to the model surface.

Best-fit model is the one closest to the data.



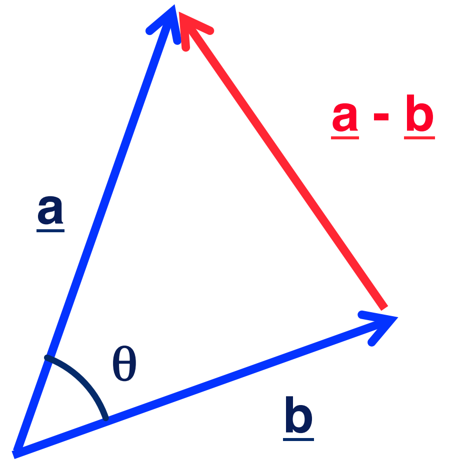

For linear models (scaling patterns), the model surface is a flat $M$-dimensional hyper-plane.

# Review: Vector Spaces

Vectors have a direction and a length.
Addition of vectors gives another vector.
Scaling a vector stretches its length.
Dot product:

$$\underline{\mathbf{a}} \bullet \underline{\mathbf{b}} = |\underline{\mathbf{a}}||\underline{\mathbf{b}}| \cos\theta$$

$$\theta = \text{"angle" between vectors } \underline{\mathbf{a}},\ \underline{\mathbf{b}}.$$




"Length" of a vector: $|\underline{\mathbf{a}}|^2 \equiv \underline{\mathbf{a}} \bullet \underline{\mathbf{a}}$

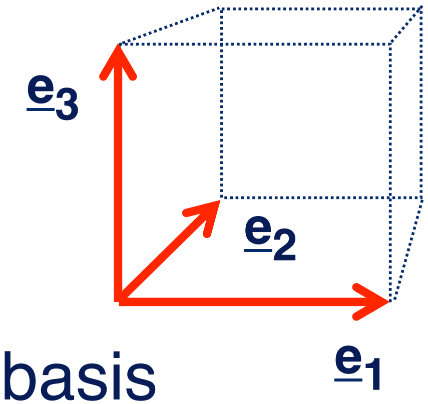
(=distance from base to tip)

"Distance" between 2 vectors: $|\,\underline{\mathbf{a}} - \underline{\mathbf{b}}\,|$

# Ortho-normal Basis Vectors

Ortho-normal basis vectors $\underline{\mathbf{e}}_i$ :

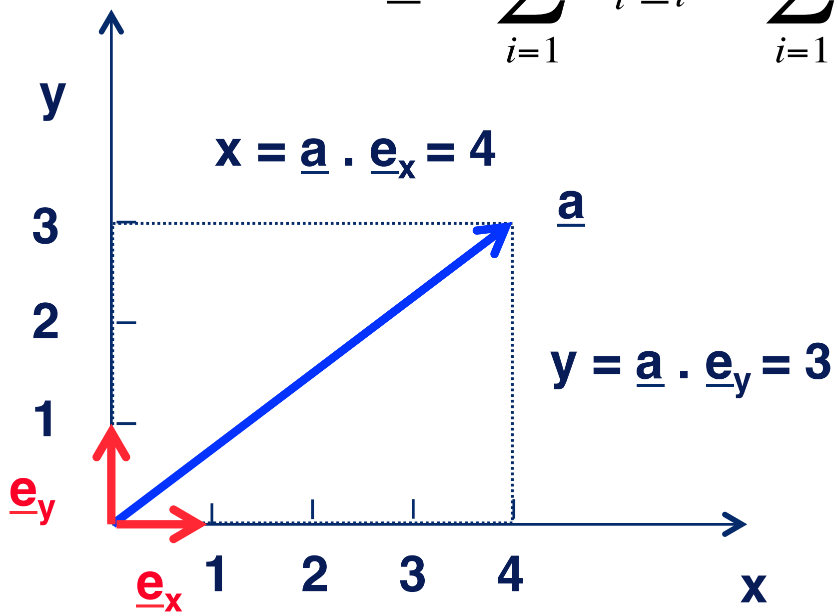
$$\underline{\mathbf{e}}_i \bullet \underline{\mathbf{e}}_j = \delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$




Any vector $\underline{\mathbf{a}}$ is a linear combination of the *N* basis vectors $\underline{\mathbf{e}}_i$ , with scale factors $a_i$

Example:

$$\underline{\mathbf{a}} = \sum_{i=1}^{N} a_i \underline{\mathbf{e}}_i = \sum_{i=1}^{N} \left( \underline{\mathbf{a}} \bullet \underline{\mathbf{e}}_i \right) \underline{\mathbf{e}}_i$$



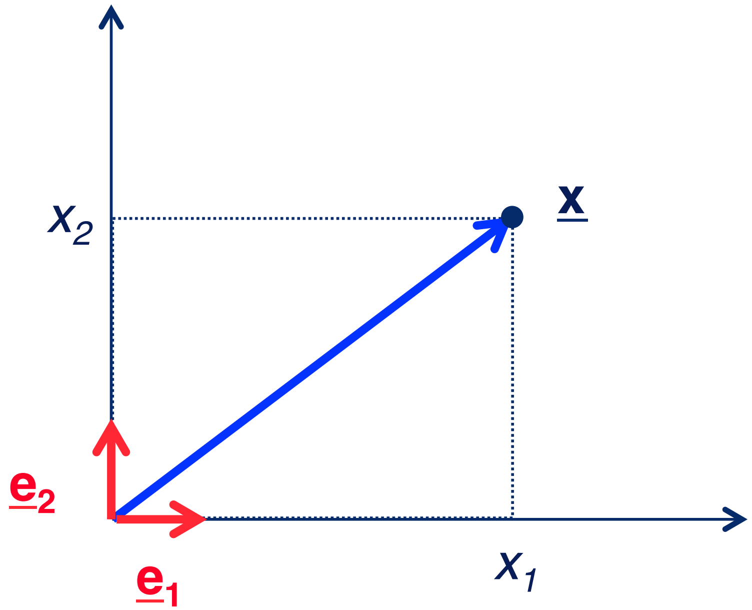
# Data Space is a Vector Space

*N* data points define a vector in *N*-dimensional "data space":

$$\underline{\mathbf{x}} = \{x_1, x_2, \ldots, x_N\}$$
$$= x_1\,\underline{\mathbf{e}}_1 + x_2\,\underline{\mathbf{e}}_2 + \ldots + x_N\,\underline{\mathbf{e}}_N$$

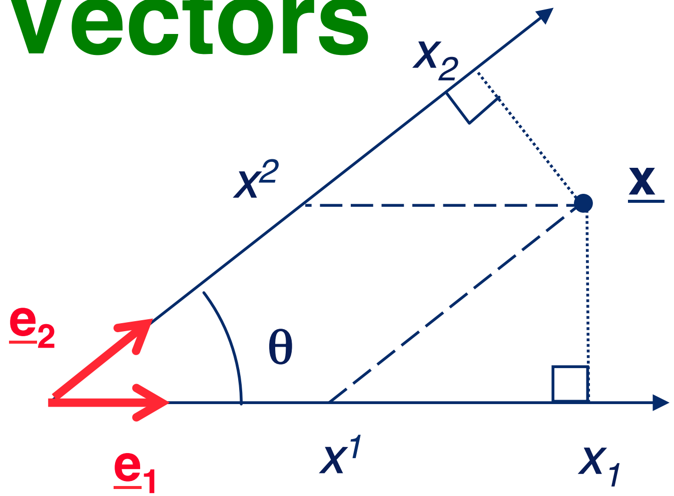
*N* basis vectors:

$$\underline{\mathbf{e}}_1 = \{1, 0, \ldots, 0\}$$
$$\underline{\mathbf{e}}_2 = \{0, 1, \ldots, 0\}$$
$$\ldots$$
$$\underline{\mathbf{e}}_N = \{0, 0, \ldots, 1\}$$




- Basis is ortho-normal if: $\underline{\mathbf{e}}_i \bullet \underline{\mathbf{e}}_j = \delta_{ij}$
- Basis vector $\underline{\mathbf{e}}_i$ selects data point $x_i$: $\underline{\mathbf{x}} \bullet \underline{\mathbf{e}}_i = x_i$
- Data point $x_i$ is the *projection* of data vector $\underline{\mathbf{x}}$ along the basis vector $\underline{\mathbf{e}}_i$ .

# Non-orthogonal Basis Vectors




In the non-orthogonal case, $\underline{\mathbf{e}}_1 \bullet \underline{\mathbf{e}}_2 = \cos\theta \neq 0$

Two ways to measure coordinates:

- **Contravariant** coordinates (index high)**:**
  $x^i$ project **parallel** to basis vectors:

$$\underline{\mathbf{x}} = x^1\, \underline{\mathbf{e}}_1 + x^2\, \underline{\mathbf{e}}_2 + \ldots + x^N\, \underline{\mathbf{e}}_N$$

$$x_1 = x^1 + x^2 \cos\theta$$

$$x_2 = x^2 + x^1 \cos\theta$$

- **Covariant** coordinates (index low):
  $x_i$ project **perpendicular** to basis vectors.

$$x_i = \sum_j g_{ij}\, x^j$$

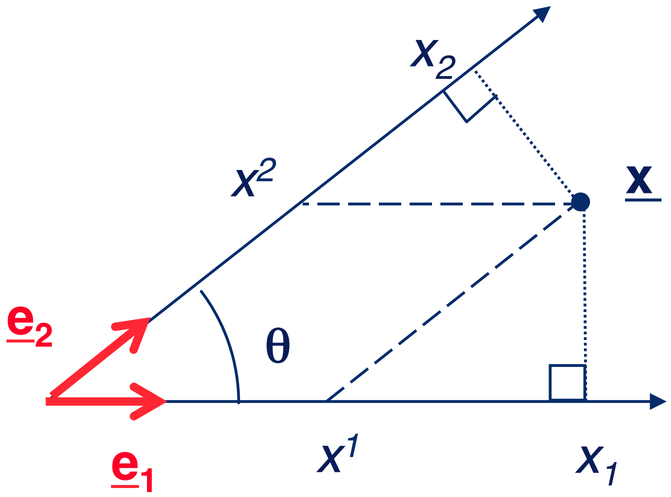
$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 & \cos\theta \\ \cos\theta & 1 \end{bmatrix} \begin{bmatrix} x^1 \\ x^2 \end{bmatrix}$$

- **Metric tensor**: $g_{ij} \equiv \underline{\mathbf{e}}_i \bullet \underline{\mathbf{e}}_j$

Dot product:

$$\underline{\mathbf{x}} \bullet \underline{\mathbf{y}} = \sum_{i,j} x^i\, y^j\, \underline{\mathbf{e}}_i \bullet \underline{\mathbf{e}}_j = \sum_{i,j} x^i\, y^j\, g_{ij} = \sum_i x^i\, y_i = \sum_j x_j\, y^j$$

# Metric for non-orthonormal Basis Vectors




$$g_{ij} \equiv \underline{\mathbf{e}}_i \bullet \underline{\mathbf{e}}_j = \begin{Bmatrix} |\underline{\mathbf{e}}_1|^2 & |\underline{\mathbf{e}}_1||\underline{\mathbf{e}}_2|\cos\theta \\ |\underline{\mathbf{e}}_1||\underline{\mathbf{e}}_2|\cos\theta & |\underline{\mathbf{e}}_2|^2 \end{Bmatrix}$$

Metric is symmetric: $g_{ij} = g_{ji}$ .

Off-diagonal terms vanish if the basis vectors are orthogonal.

Diagonal terms define the lengths of the basis vectors.

# Data sets and Functions as Vector Spaces

- A data set, $X_i$, $i = 1, \ldots, N$, is also an $N$-component vector ( $X_1, X_2, \ldots, X_N$ ), one dimension for each data point.
- The data vector is a single point in the ***N*-dimensional data space**.

---

- A function, $f(t)$, is a vector in an **infinite-dimensional** vector space, one dimension for each value of $t$.
- The "dot product" between 2 functions depends on a **weighting function** $w(t)$:

$$\langle f, g \rangle \equiv \int_{-\infty}^{\infty} f(t)\, g(t)\, w(t)\, dt$$

**Weighting function**

Each weighting function $w(t)$ gives a different dot product, a different distance measure, a different vector space.

***Which w( t ) to use for data analysis?***

# $\chi^2$ as (distance)$^2$ in function space

- The (absolute value)$^2$ of a function *f( t )* :

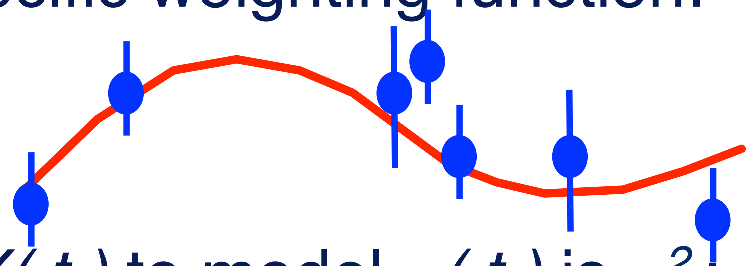
$$\| f \|^2 \equiv \langle f, f \rangle = \int f^2(t)\, w(t)\, dt$$

- The (distance)$^2$ between *f( t )* and *g( t )* :

$$\| f - g \|^2 \equiv \langle f - g, f - g \rangle = \int \big( f(t) - g(t) \big)^2 w(t)\, dt$$

- A dataset ( $X_i$ +/- $\sigma_i$ ) at $t = t_i$ defines a specific weighting function:

$$w(t) \equiv \sum_{i=1}^{N} \frac{\delta(t - t_i)}{\sigma_i^2}$$



- With this *w( t )*, the (distance)$^2$ from data *X( t )* to model $\mu( t )$ is $\chi^2$ :

$$\| X - \mu \|^2 = \sum_{i=1}^{N} \left( \frac{X_i - \mu(t_i)}{\sigma_i} \right)^2 = \chi^2 .$$

***Each dataset defines its own weighting function.***

# The Data-Space Metric:
## $\sigma$ is the unit of distance. $\chi^2$ is (distance)$^2$

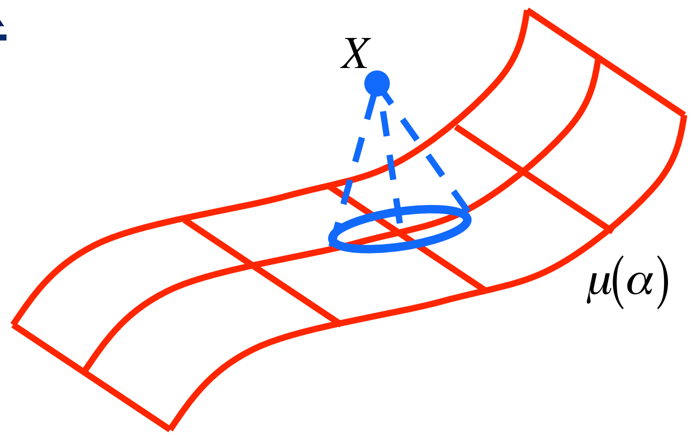
- Define the data-space dot product with inverse-variance weights:

$$w_i = \frac{1}{\sigma_i^2} \quad \Rightarrow \quad \underline{\mathbf{a}} \bullet \underline{\mathbf{b}} = \sum_{i=1}^{N} a_i \, b_i \, w_i = \sum_{i=1}^{N} \frac{a_i \, b_i}{\sigma_i^2}$$

$$\left| \underline{\mathbf{a}} - \underline{\mathbf{b}} \right|^2 = \sum_{i=1}^{N} \left( \frac{a_i - b_i}{\sigma_i} \right)^2 .$$

- Then, the (distance)$^2$ between data $\underline{\mathbf{x}}$ and parameterised model $\underline{\boldsymbol{\mu}}(\alpha)$ is:

$$\chi^2 = \sum_{i=1}^{N} \left( \frac{X_i - \mu_i(\alpha)}{\sigma_i} \right)^2 = \left| \underline{\mathbf{X}} - \underline{\mu}(\alpha) \right|^2 .$$



# Optimal Scaling in Vector Space Notation

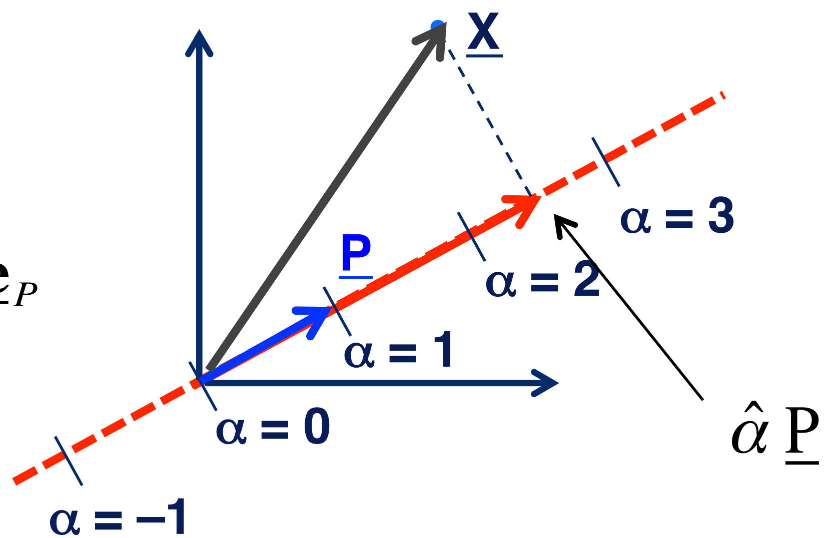
- Minimise $\chi^2$ -> pick model closest to the data.
- Scaling a pattern: $\underline{\mu}(\alpha) = \alpha \underline{\mathbf{P}}$ :

$$\langle X_i \rangle = \mu_i(\alpha) = \alpha P_i$$

- The pattern $\underline{\mathbf{P}}$ is a **vector** in data space.
- The model $\alpha \underline{\mathbf{P}}$ is a **line** in data space, multiples of $\underline{\mathbf{P}}$ .
- The best fit is the point along the line closest to the data $\underline{\mathbf{X}}$

$$\hat{\alpha} = \frac{\sum X_i P_i / \sigma_i^2}{\sum P_i^2 / \sigma_i^2} = \frac{\underline{\mathbf{X}} \bullet \underline{\mathbf{P}}}{\underline{\mathbf{P}} \bullet \underline{\mathbf{P}}}$$

$$\underline{\mu}(\hat{\alpha}) = \hat{\alpha}\, \underline{\mathbf{P}} = \left( \frac{\underline{\mathbf{X}} \bullet \underline{\mathbf{P}}}{\underline{\mathbf{P}} \bullet \underline{\mathbf{P}}} \right) \underline{\mathbf{P}} = \left( \underline{\mathbf{X}} \bullet \underline{\mathbf{e}}_P \right) \underline{\mathbf{e}}_P$$




- Unit vector along $\underline{\mathbf{P}}$ : $\quad \underline{\mathbf{e}}_P \equiv \frac{\underline{\mathbf{P}}}{\left| \underline{\mathbf{P}} \right|}$

# Stretching the Basis Vectors

Using the vector notation,

$$\hat{\alpha} = \frac{\underline{\mathbf{P}} \bullet \underline{\mathbf{X}}}{\underline{\mathbf{P}} \bullet \underline{\mathbf{P}}} = \frac{\sum_i \sum_j X^i\, P^j\, g_{i\,j}}{\sum_i \sum_j P^i\, P^j\, g_{i\,j}} = \frac{\sum_i X^i\, P^i \big/ \sigma_i^2}{\sum_i \left(P^i\right)^2 \big/ \sigma_i^2}$$

$$\underline{\mathbf{e}}_1 = \{1,0,\ldots,0\}$$
$$\underline{\mathbf{e}}_2 = \{0,1,\ldots,0\}$$
$$\ldots$$
$$\underline{\mathbf{e}}_N = \{0,0,\ldots,1\}$$

So the $\underline{\mathbf{e}}_i$ basis vectors are **orthogonal but not unit length**, given the data-space metric

$$g_{i\,j} = \underline{e}_i \bullet \underline{e}_j = \frac{1}{\sigma_i^2}\delta_{i\,j}$$

i.e. $\boldsymbol{\sigma_i}$ **is the natural unit of distance** on the $i_{th}$ axis of data space!

We can "stretch" axis i by factor $\sigma_i$ to define a new set of **ortho-normal basis vectors** $\underline{\mathbf{b}}_i$ :

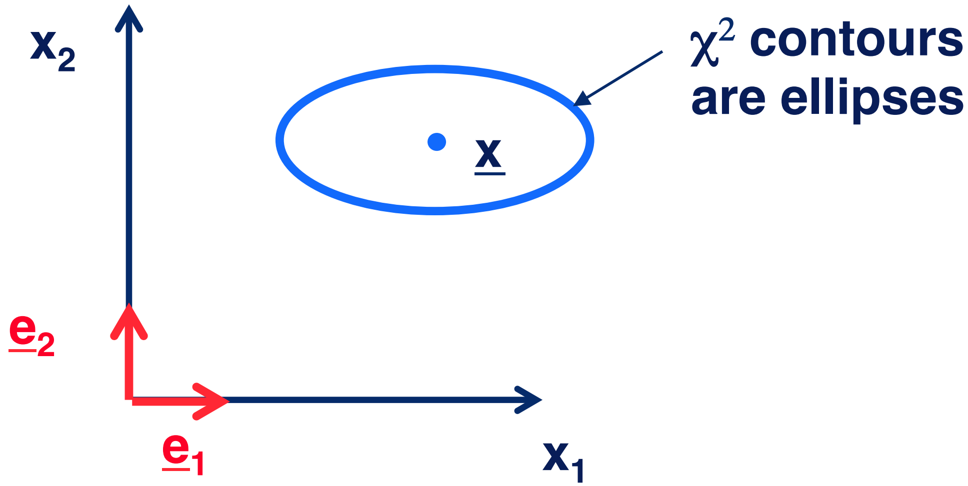
$$\underline{\mathbf{b}}_i \equiv \sigma_i\, \underline{\mathbf{e}}_i \qquad \underline{\mathbf{b}}_i \bullet \underline{\mathbf{b}}_j = \delta_{i\,j}$$

$$\underline{\mathbf{b}}_1 = \{\sigma_1,0,\ldots,0\}$$
$$\underline{\mathbf{b}}_2 = \{0,\sigma_2,\ldots,0\}$$
$$\ldots$$
$$\underline{\mathbf{b}}_N = \{0,0,\ldots,\sigma_N\}$$

# Stretch basis vectors to make $\chi^2$ ellipses become circles

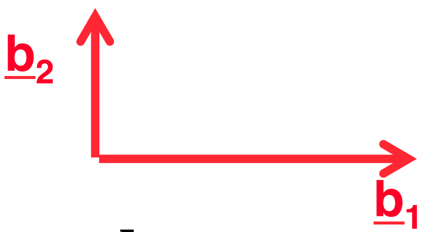
Old basis vectors:

$$\underline{\mathbf{x}} = \sum_{i=1}^{N} x_i \, \underline{\mathbf{e}}_i \qquad g_{ij} = \underline{\mathbf{e}}_i \bullet \underline{\mathbf{e}}_j = \frac{\delta_{ij}}{\sigma_i^2}$$



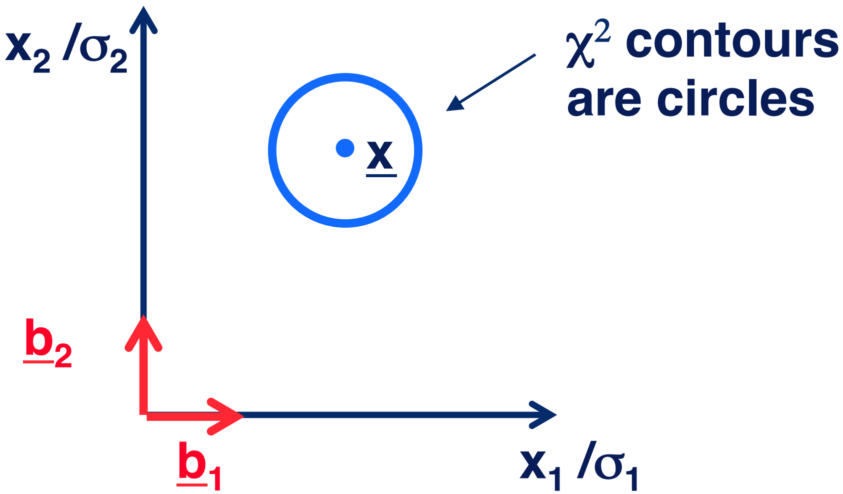

Orthogonal, but not normalised.

"Stretched" basis vectors are orthonormal:

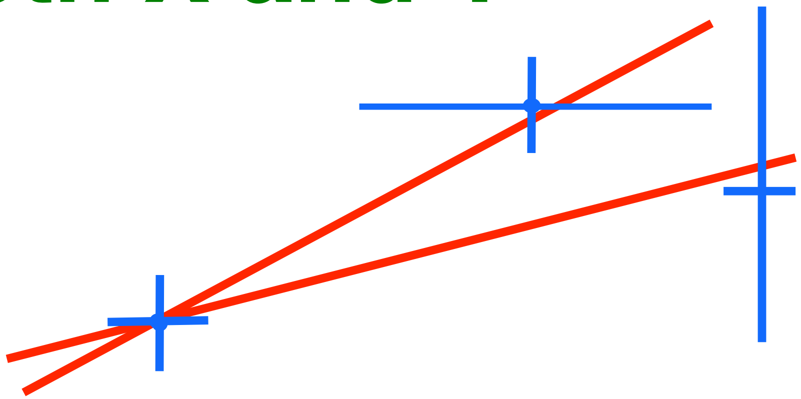



$$\underline{\mathbf{b}}_i \equiv \sigma_i \, \underline{\mathbf{e}}_i \qquad g_{ij} \equiv \underline{\mathbf{b}}_i \bullet \underline{\mathbf{b}}_j = \delta_{ij}$$

$$\underline{\mathbf{x}} = \sum_{i=1}^{N} \langle \mathbf{x}, \underline{\mathbf{b}}_i \rangle \underline{\mathbf{b}}_i = \sum_{i=1}^{N} \frac{x_i}{\sigma_i} \underline{\mathbf{b}}_i$$



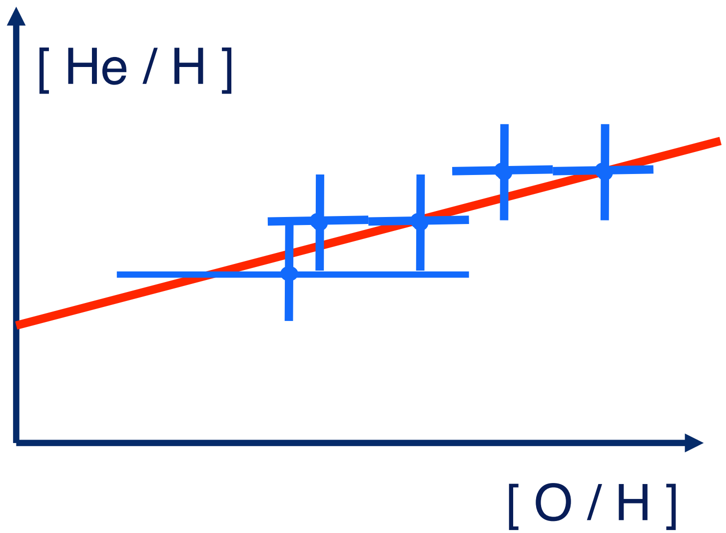
# Error Bars in both X and Y



Wrong ways to fit a line :

1. $y(x) = a\,x + b \quad (\sigma_x = 0)$
2. $x(y) = c\,y + d \quad (\sigma_y = 0)$
3. split difference between 1 and 2.

Example: Primordial He abundance:



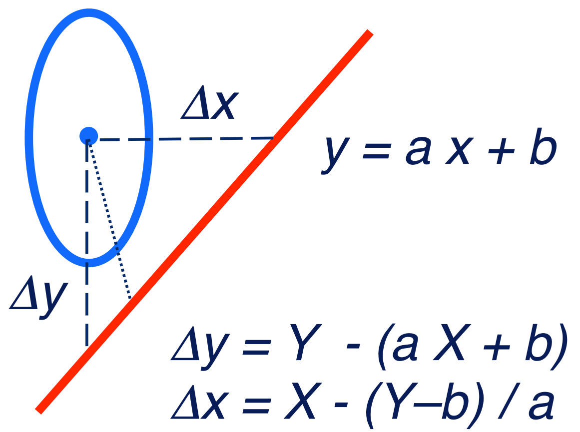

Extrapolate fit line to [ O / H ] = 0.

**Key concept: X +/- $\sigma_X$ and Y +/- $\sigma_Y$ are 2 independent dimensions of the 2N-dimensional data space.**

# Line Fit with error bars in both X and Y

Data: $X \pm \sigma_X \quad Y \pm \sigma_Y$

Model: $y = a\,x + b$

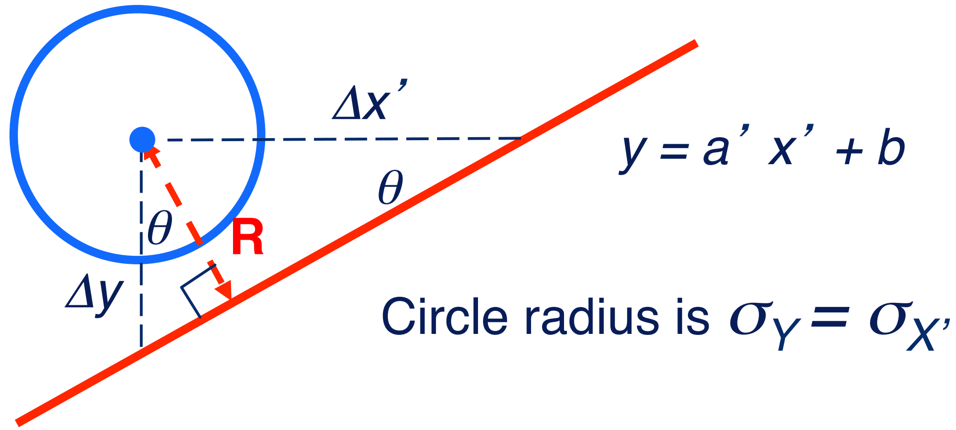



For $\sigma_X \neq \sigma_Y$, **where is the point of closest approach ?**

Not obvious. ☹

***Horizontal stretch by factor*** $\sigma_Y / \sigma_X$
*makes the probability cloud round.*

*Also changes the slope: a => a'*




$$\Delta x' = \frac{\sigma_Y}{\sigma_X}\,\Delta x \qquad a' = \frac{\Delta y}{\Delta x'} = \frac{\sigma_X}{\sigma_Y}\,a = \tan\theta$$

Closest approach at $R = \Delta y \cos\theta$

$$\left(\frac{R}{\Delta y}\right)^2 = \frac{\cos^2\theta}{\cos^2\theta + \sin^2\theta} = \frac{1}{1+\tan^2\theta} = \frac{\sigma_Y^2}{\sigma_Y^2 + a^2\sigma_X^2}$$

$$\left(\frac{R}{\sigma_Y}\right)^2 = \left(\frac{\Delta y}{\sigma_Y}\,\frac{R}{\Delta y}\right)^2 = \frac{\Delta y^2}{\sigma_Y^2 + a^2\sigma_X^2}$$

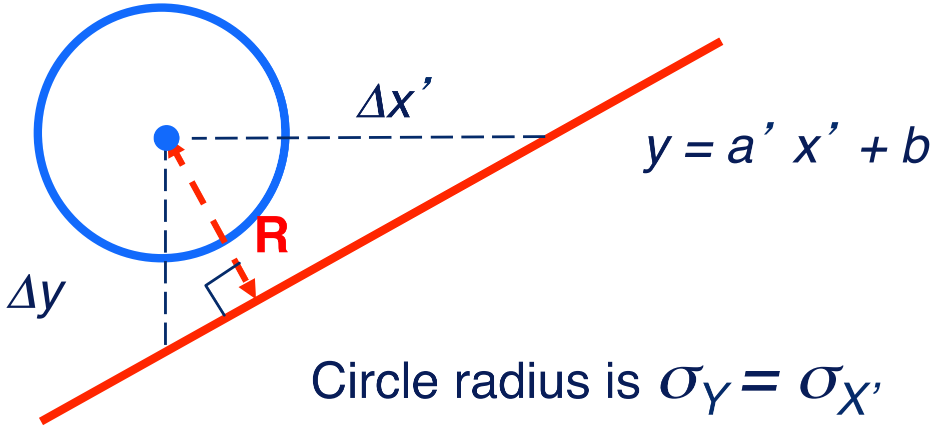
# Defining $\chi^2$ for errors in both X and Y

Horizontal stretch makes probability cloud round.

Circle radius is $\sigma_Y = \sigma_{X'}$ .

Distance $R$ at closest approach is :

$$\left(\frac{R}{\sigma_Y}\right)^2 = \frac{\Delta y^2}{\sigma_Y^2 + a^2 \sigma_X^2}$$




Note: Need a different stretch for each data point.

Total (distance)$^2$ in the *2 N* - dimensional data space:

$$\chi^2 = \sum_{i=1}^{N} \left[ \left( \frac{\varepsilon(Y_i)}{\sigma(Y_i)} \right)^2 + \left( \frac{\varepsilon(X'_i)}{\sigma(X'_i)} \right)^2 \right] = \sum_{i-1}^{N} \left( \frac{\varepsilon(Y_i)^2 + \varepsilon(X'_i)^2}{\sigma^2(Y_i)} \right)$$

$$= \sum_{i=1}^{N} \left( \frac{R}{\sigma(Y_i)} \right)^2 = \sum_{i=1}^{N} \frac{\left(Y_i - (a\,X_i + b)\right)^2}{\sigma^2(Y_i) + a^2 \sigma^2(X_i)}$$