Fundamental statistics for understanding environmental data

Charles Turner || 29/10/25

About me

- MSci in Physics at Imperial College London
- PhD in Physical Oceanography at National Oceanography Centre, UK
- Worked as a data scientist at Environment Technologies & Analytics in Perth.
- Currently a senior research software engineer at ACCESS-NRI
- Most of this talk will focus on applying statistical theory I learnt at my time at Imperial & used during my PhD.
- We are going to try to get through a large chunk of a 10 lecture undergraduate stats course in ~30 minutes.
- Plus some extras…

```python
"""
Data source:
https://mesonet.agron.iastate.edu/request/download.phtml?network=AU__ASOS
Hourly temperature data from Perth Airport for 2024.
"""
import polars as pl

first_week = pl.read_csv("YPPH.csv", try_parse_dates=True).with_columns(
    pl.col("valid").alias("Datetime"), pl.col("tmpc").alias("Temperature")
).select("Datetime", "Temperature")


first_week.head(10)
```

shape: (10, 2)

Datetime	Temperature
datetime[μs]	f64
2024-01-01 00:00:00	23.0
2024-01-01 01:00:00	24.0
2024-01-01 02:00:00	25.0
2024-01-01 03:00:00	28.0
2024-01-01 04:00:00	30.0
2024-01-01 05:00:00	27.0
2024-01-01 06:00:00	28.0
2024-01-01 07:00:00	29.0
2024-01-01 08:00:00	27.0
2024-01-01 09:00:00	27.0

```python
first_week.select("Temperature").describe()
```

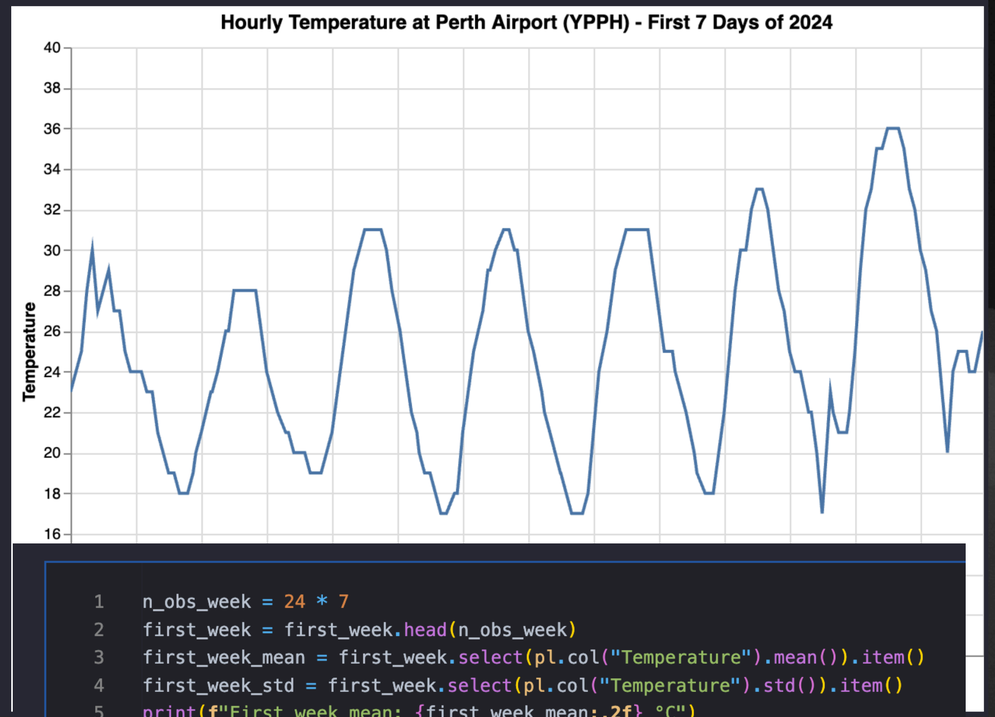
✓ 0.0s

shape: (9, 2)

statistic	Temperature
str	f64
"count"	168.0
"null_count"	0.0
"mean"	24.922619
"std"	4.85354
"min"	17.0
"25%"	21.0
"50%"	25.0
"75%"	29.0
"max"	36.0

```python
import altair as alt
first_week.head(24*7).plot.line("Datetime", "Temperature").encode(
    x='Datetime:T',
    y=alt.Y('Temperature:Q', scale=alt.Scale(domain=[10, 40]))
).properties(
        title='Hourly Temperature at Perth Airport (YPPH) - First 7 Days of 2024',
        width=600,
        height=400
    )
```

Python




```python
n_obs_week = 24 * 7
first_week = first_week.head(n_obs_week)
first_week_mean = first_week.select(pl.col("Temperature").mean()).item()
first_week_std = first_week.select(pl.col("Temperature").std()).item()
print(f"First week mean: {first_week_mean:.2f} °C")
print(f"First week std: {first_week_std:.2f} °C")
# Standard error of the mean
first_week_sem = first_week_std / (n_obs_week) ** 0.5
print(f"First week SEM: {first_week_sem:.2f} °C")
```

```
First week mean: 24.92 °C
First week std. dev: 4.85 °C
First week SEM: 0.37 °C
```

```python
# Upsample to ten minute intervals
df_10min = first_week.upsample("Datetime", every= "10m").with_columns(
    pl.col("Temperature").interpolate().alias("Interpolated Temperature")
)
df_10min.head(10)
```

shape: (10, 3)

Datetime	Temperature	Interpolated Temperature
datetime[μs]	f64	f64
2024-01-01 00:00:00	23.0	23.0
2024-01-01 00:10:00	null	23.166667
2024-01-01 00:20:00	null	23.333333
2024-01-01 00:30:00	null	23.5
2024-01-01 00:40:00	null	23.666667
2024-01-01 00:50:00	null	23.833333
2024-01-01 01:00:00	24.0	24.0
2024-01-01 01:10:00	null	24.166667
2024-01-01 01:20:00	null	24.333333
2024-01-01 01:30:00	null	24.5

```python
# Now get statistics for the 10-min data
first_week_10min = df_10min.head(n_obs_week * 6)
first_week_10min_mean = first_week_10min.select(pl.col("Interpolated Temperature").mean()).item()
first_week_10min_std = first_week_10min.select(pl.col("Interpolated Temperature").std()).item()
print(f"First week (10-min) mean: {first_week_10min_mean:.2f} °C")
print(f"First week (10-min) std: {first_week_10min_std:.2f} °C")
# Standard error of the mean
first_week_10min_sem = first_week_10min_std / (n_obs_week * 6) ** 0.5
print(f"First week (10-min) SEM: {first_week_10min_sem:.2f} °C")
```

```
First week (10-min) mean: 24.94 °C
First week (10-min) std: 4.78 °C
First week (10-min) SEM: 0.15 °C
```

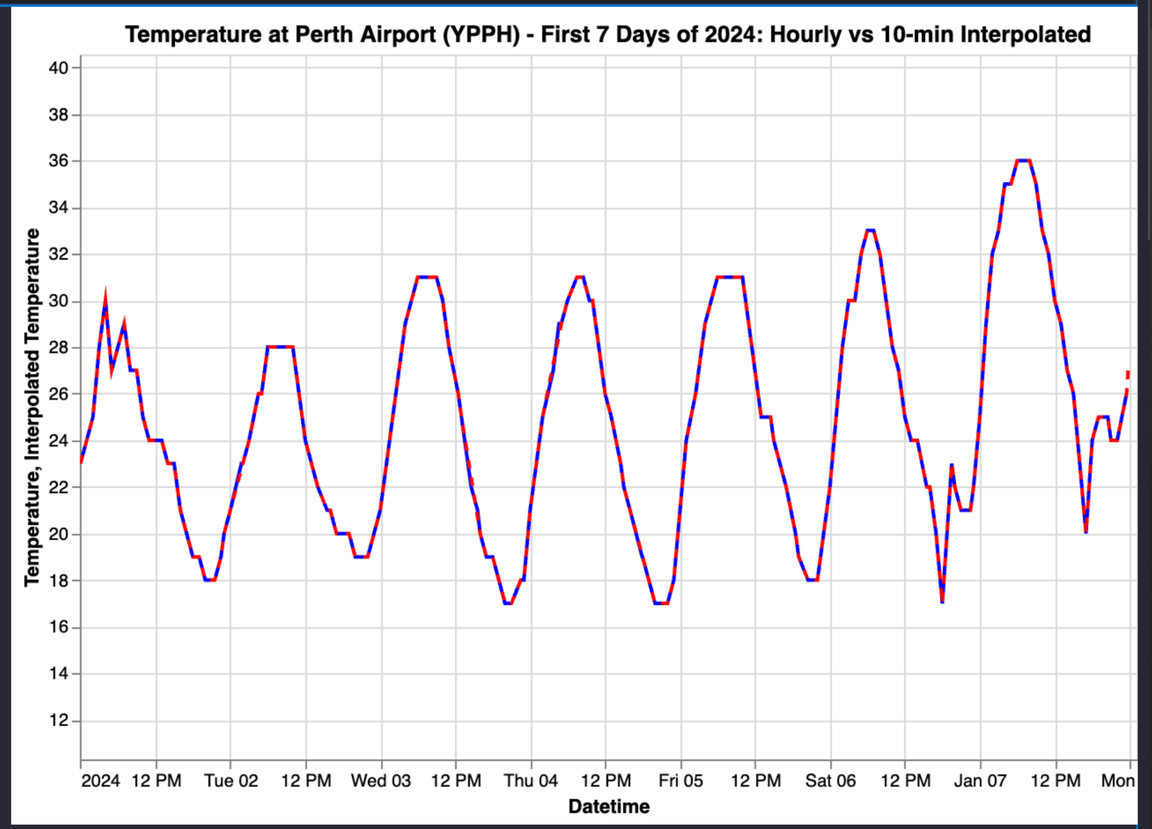
```python
hourly = first_week.head(n_obs_week).plot.line("Datetime", "Temperature").encode(
    x='Datetime:T',
    y=alt.Y('Temperature:Q', scale=alt.Scale(domain=[10, 40])),
    color=alt.value('blue'),
)

_10min = df_10min.head(n_obs_week * 6).plot.line("Datetime", "Interpolated Temperature").encode(
    x='Datetime:T',
    y=alt.Y('Interpolated Temperature:Q', scale=alt.Scale(domain=[10 , 40])),
    color=alt.value('red') ,
    strokeDash=alt.value([5,5])
)

# Layer them together
combined_chart = (hourly + _10min).properties(
    title='Temperature at Perth Airport (YPPH) - First 7 Days of 2024: Hourly vs 10-min Interpolated',
    width=600,
    height=400
).resolve_scale(color='independent')

combined_chart
```

Python



Resolving a paradox

```
n_obs_week = 24 * 7
first_week = first_week.head(n_obs_week)
first_week_mean = first_week.select(pl.col("Temperature").mean()).item()
first_week_std = first_week.select(pl.col("Temperature").std()).item()
print(f"First week mean: {first_week_mean:.2f} °C")
print(f"First week std: {first_week_std:.2f} °C")
# Standard error of the mean
first_week_sem = first_week_std / (n_obs_week) ** 0.5
print(f"First week SEM: {first_week_sem:.2f} °C")
```

```
First week mean: 24.92 °C
First week std. dev: 4.85 °C
First week SEM: 0.37 °C
```

```
# Now get statistics for the 10-min data
first_week_10min = df_10min.head(n_obs_week * 6)
first_week_10min_mean = first_week_10min.select(pl.col("Interpolated Temperature").mean()).item()
first_week_10min_std = first_week_10min.select(pl.col("Interpolated Temperature").std()).item()
print(f"First week (10-min) mean: {first_week_10min_mean:.2f} °C")
print(f"First week (10-min) std: {first_week_10min_std:.2f} °C")
# Standard error of the mean
first_week_10min_sem = first_week_10min_std / (n_obs_week * 6) ** 0.5
print(f"First week (10-min) SEM: {first_week_10min_sem:.2f} °C")
```

```
First week (10-min) mean: 24.94 °C
First week (10-min) std: 4.78 °C
First week (10-min) SEM: 0.15 °C
```

- By upsampling data, *without adding any additional information*, we appear to have reduced our standard error by almost two thirds!
- How can this possibly be?

$$\sigma_{\bar{x}} = \frac{\sigma_x}{\sqrt{N}}$$

Propagation of errors

$$\delta f = \sum_i \frac{\partial f}{\partial x_i} \delta x_i$$

$$\sigma^2(f) = \langle (\delta f)^2 \rangle$$

$$\sigma^2(f) = \sum_i \sum_j \frac{\partial f}{\partial x_i} \frac{\partial f}{\partial x_j} \langle \delta x_i \delta x_j \rangle$$

Deriving the propagation of errors

$$\sigma^2(f) = \sum_i \sum_j \frac{\partial f}{\partial x_i} \frac{\partial f}{\partial x_j} \langle \delta x_i \, \delta x_j \rangle$$

Assume no error covariance: $\langle \delta x_i \, \delta x_j \rangle = \langle \delta x_i \, \delta x_j \rangle \delta_{ij}$ where $\delta_{ij} = 1$ if $i = j$, 0 otherwise.

No error covariance: Coin flips, Dice rolls, Atomic Decays

$$\sigma^2(f) = \sum_i \left(\frac{\partial f}{\partial x_i} \right)^2 \sigma^2(x_i) = \sum_i \left(\frac{\partial f}{\partial x_i} \sigma_i \right)^2$$

Error covariance: Air Temperature, Rainy days, Buses coming past

Standard error of the mean

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i \quad \Longrightarrow \quad \sigma_{\bar{x}}^2 = \sum_{i=1}^{N}\left(\frac{\partial \bar{x}}{\partial x_i}\sigma_i\right)^2$$

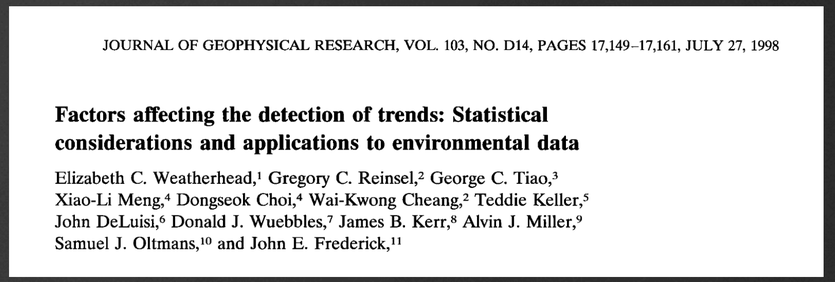
$$\text{since } \frac{\partial \bar{x}}{\partial x_i} = \frac{1}{N}$$

$$\sigma_{\bar{x}}^2 = \frac{1}{N^2}\sum_{i=1}^{N}\sigma_i^2 \quad \Longrightarrow \quad \sigma_{\bar{x}} = \frac{\sigma_x}{\sqrt{N}}$$

Autocorrelation

Assume no error covariance: $\langle \delta x_i \, \delta x_j \rangle = \langle \delta x_i \, \delta x_j \rangle \delta_{ij}$ where $\delta_{ij} = 1$ if $i = j$, 0 otherwise.

- So far, we have assumed all observations are independent.
- This is often a poor assumption for spatial and time series data.
- Reduces ‘effective sample size’ by factor $(1+\phi)/(1-\phi)$ for AR1 process
- https://agupubs.onlinelibrary.wiley.com/doi/epdf/10.1029/98JD00995



JOURNAL OF GEOPHYSICAL RESEARCH, VOL. 103, NO. D14, PAGES 17,149–17,161, JULY 27, 1998

Factors affecting the detection of trends: Statistical considerations and applications to environmental data

Elizabeth C. Weatherhead,[1] Gregory C. Reinsel,[2] George C. Tiao,[3] Xiao-Li Meng,[4] Dongseok Choi,[4] Wai-Kwong Cheang,[2] Teddie Keller,[5] John DeLuisi,[6] Donald J. Wuebbles,[7] James B. Kerr,[8] Alvin J. Miller,[9] Samuel J. Oltmans,[10] and John E. Frederick,[11]

“Detection of long-term, linear trends is affected by a number of factors, including the size of trend to be detected, the time span of available data, and the magnitude of variability and autocorrelation of the noise in the data. The number of years of data necessary to detect a trend is strongly dependent on, and increases with, the magnitude of variance and autocorrelation coefficient of the noise. ***For a typical range of values of variance and autocorrelation coefficients the number of years of data needed to detect a trend of 5%/decade can vary from 10 to >20 years***, implying that in choosing sites to detect trends some locations are likely to be more efficient and cost-effective than others. Additionally, some environmental variables allow for an earlier detection of trends than other variables because of their low variability and autocorrelation”

Emphasis mine

Autocorrelation

```python
first_week.remove(
    pl.col("Temperature").is_nan()
).with_columns(
    pl.col("Temperature").shift(1).alias("Temperature (t-1)")
).select(
    pl.corr("Temperature", "Temperature (t-1)")
).to_series(
).alias("Hourly autocorrelation")


```

✓ 0.0s

shape: (1,)

Hourly autocorrelation
f64
0.946466

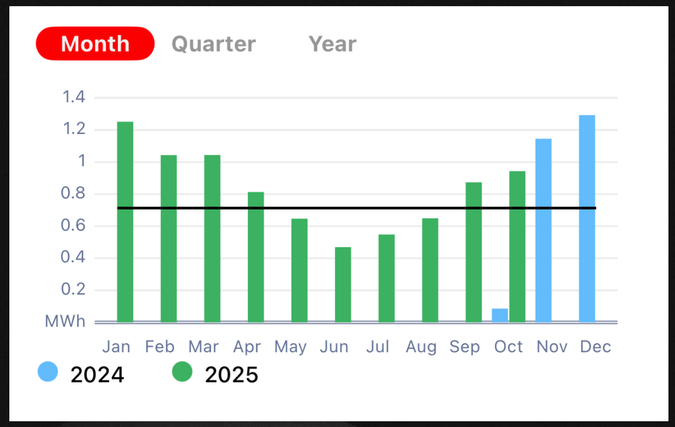
```python
first_week_10min.remove(
    pl.col("Interpolated Temperature").is_nan()
).with_columns(
    pl.col("Interpolated Temperature").shift(1).alias("Interpolated Temperature (t-1)")
).select(
    pl.corr("Interpolated Temperature", "Interpolated Temperature (t-1)")
).to_series(
).alias("10-min autocorrelation")

```

✓ 0.0s

shape: (1,)

10-min autocorrelation
f64
0.998547



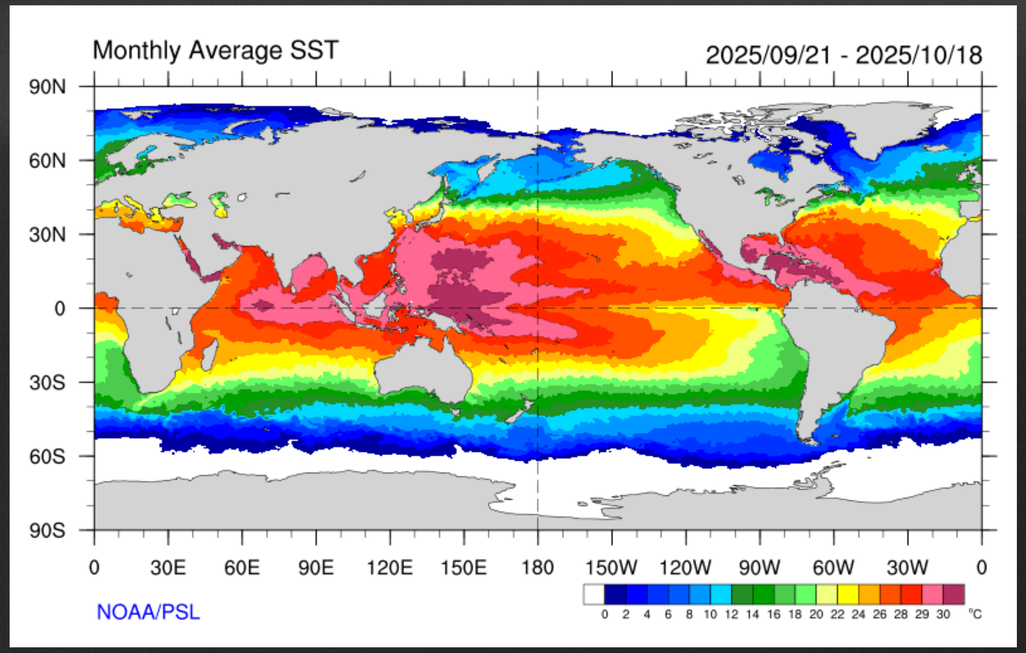
- When we interpolate, we massively increase the autocorrelation.
- This reduces the *effective sample size*

$$N_e = N\frac{1-\phi}{1+\phi}$$

- The above formula is only for an AR1 (Markovian) time series.
- It is a poor assumption for many processes
- Weatherhead et. al 1998 contains a much more complete treatment of the problem.

Spatial Covariance

- The spatial analogue of an autocorrelation is a **correlation length**.
- *“If I measure the sea surface temperature at point x, how far away can point y be where I still know the temperature?”*
- Crucially important in surface fitting



Surface Fitting

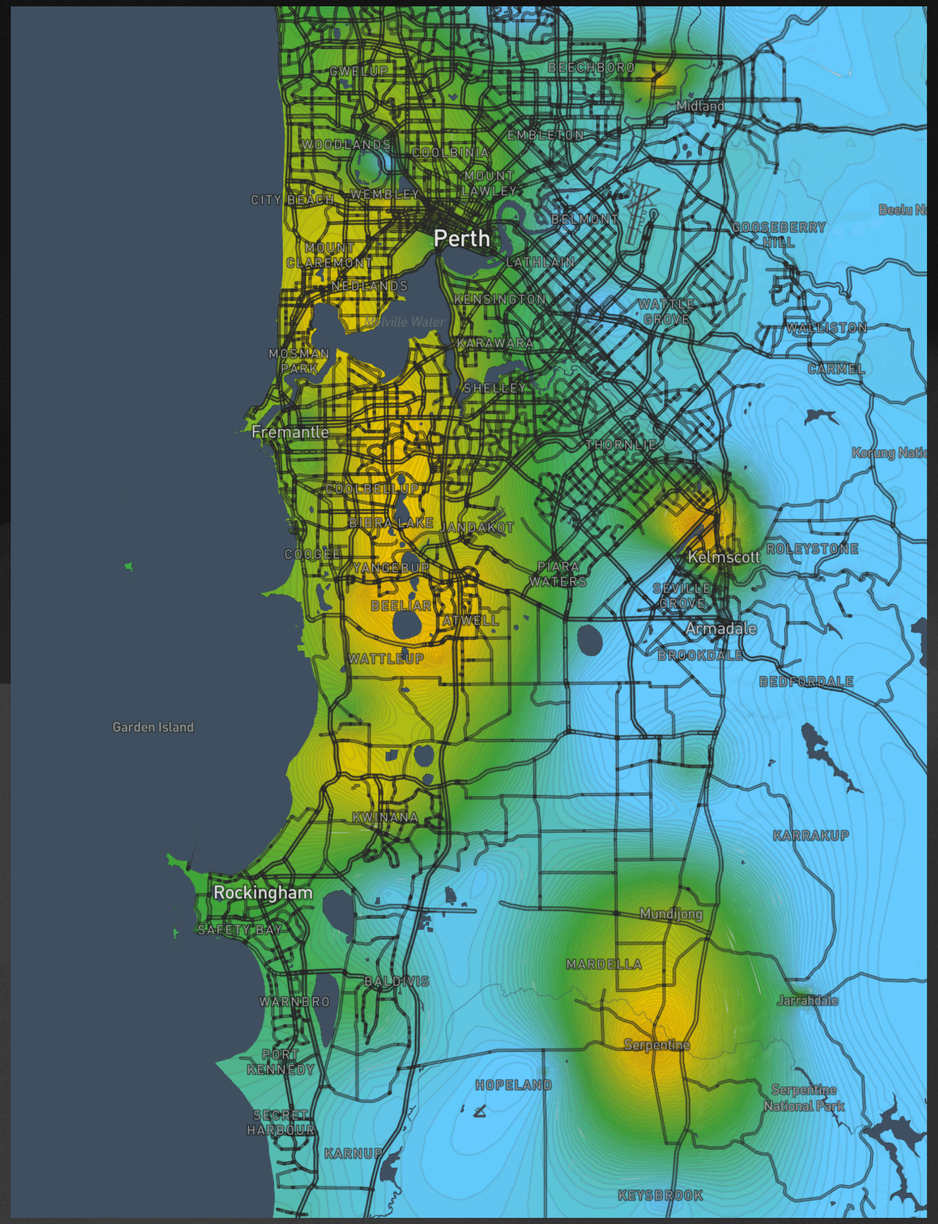
- Commonly used methodologies:
  - Optimal Interpolation
  - Kriging
  - Dynamical Interpolating Variational Analysis (DIVA)

“The correlation length L gives an indication of the distance over which a given data point influences its neighbourhood (Section 2.3.1). Similarly to other interpolation techniques, it is an essential parameter for obtaining meaningful results. The value of L can be provided a priori by the user, or determined using the data distribution itself, as explained in the next Section.

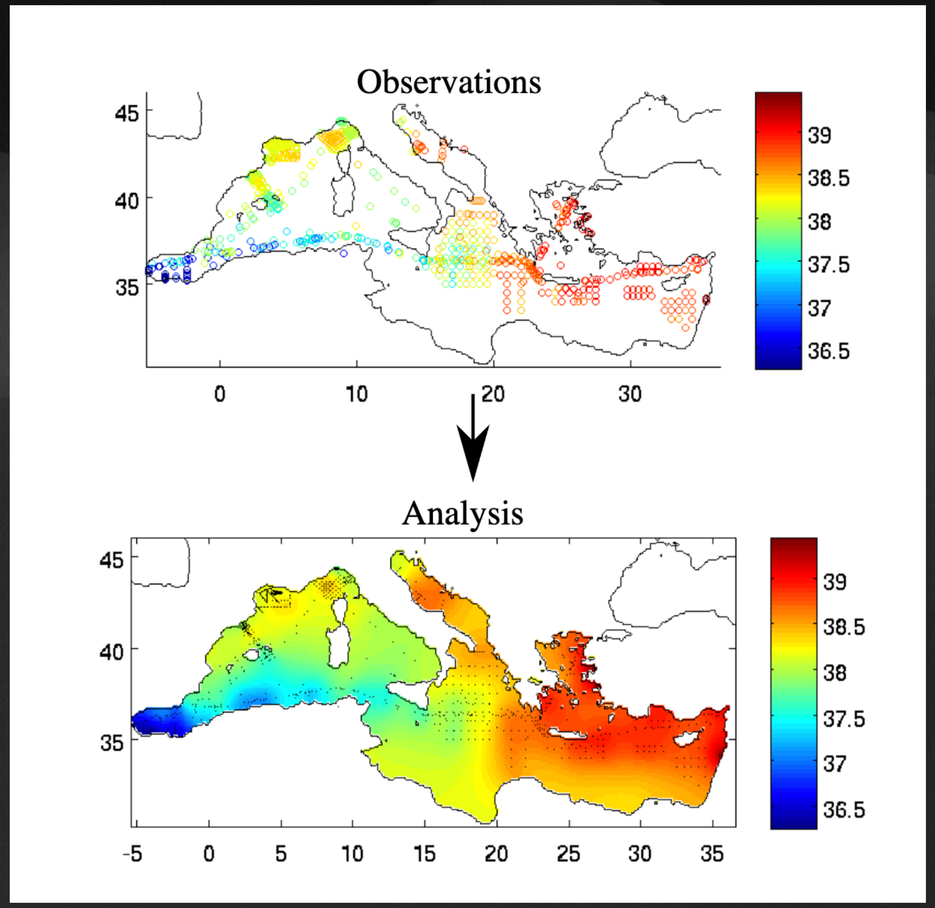
The method to evaluate L is to fit the theoretical kernel of (2.11) (see Fig. 2.5) to the correlation between data assuming spatial isotropy and homogeneity in correlations. The quality of the fit will depend on the number of data points: a value of L obtained with few data points has to be considered with care.”

https://github.com/gher-uliege/Diva-User-Guide

Surface Fitting



Source: https://racairhealthmonitor.ramboll-shair.com/?pollutant=pm25



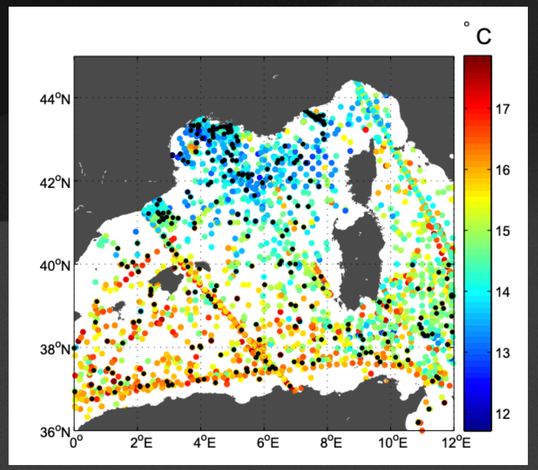
Reproduced from https://gher-uliege.github.io/DIVAnd-presentation/

Detection of wrong analysis with Diva

Charles ctroupin@ulg.ac.be

December 7, 2011

Reproduced from https://orbi.uliege.be/bitstream/2268/114186/1/Diva_misuse_report.pdf



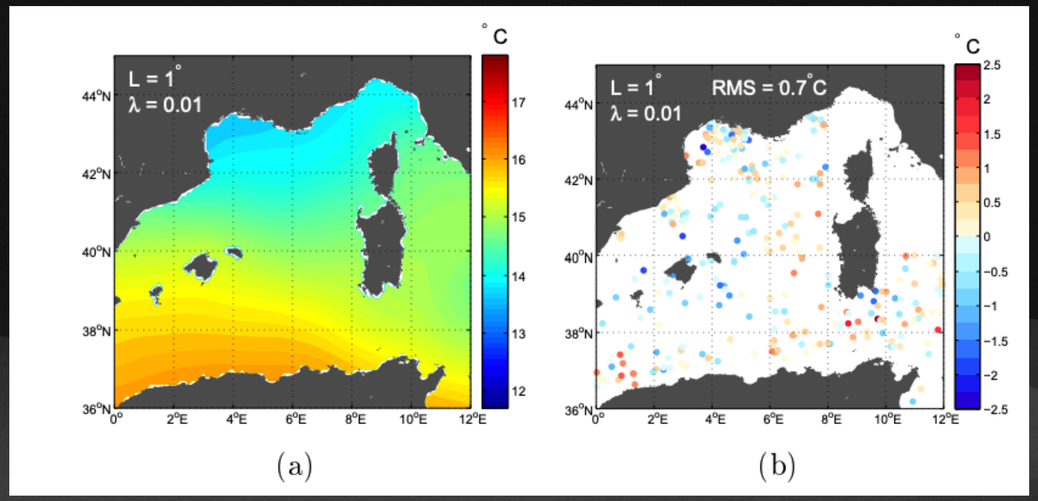
4.1 Signal-to-noise ratio is too low $L = 1°$ and $\lambda = 0.01$

(a) Analysis: the field is very smooth, meaning that the regularisation constraint dominate the data influence. The field only exhibits a meridional gradient of temperature.

(b) Misfits: nothing particular to observe.

(c) Histogram: the distribution is tighter than with the original measurements (Fig. 1b). The mean value is slightly higher, this difference is probably due to the spatial distribution of the data. The standard deviation decreased of almost 40%.

Such an analysis is acceptable when very few data are available, or if the considered period is long.



4.2 Correlation length is too small $L = 0.1°$ and $\lambda = 3$

(d) Analysis: the field is noisy and cruise tracks can be observed (compare with data positions in Fig. 1).

(e) Misfits: again, nothing particular, except that the RMS is decreased with respect to the previous case.

(f) Histogram: as expected, the range of values and the variance is higher than in the previous case, while the mean value is similar. Most of the field values are not between 14°C and 16°C.

In this example the signal-to-noise ratio was set to 3, but analysis with lower values for λ yielded results with similar features. Such an analysis would be acceptable when dealing with measurements taken over a short period (a few hours or days, for instance satellite data). Note that it is reasonable to select a value of L not smaller than the mean distance between data.

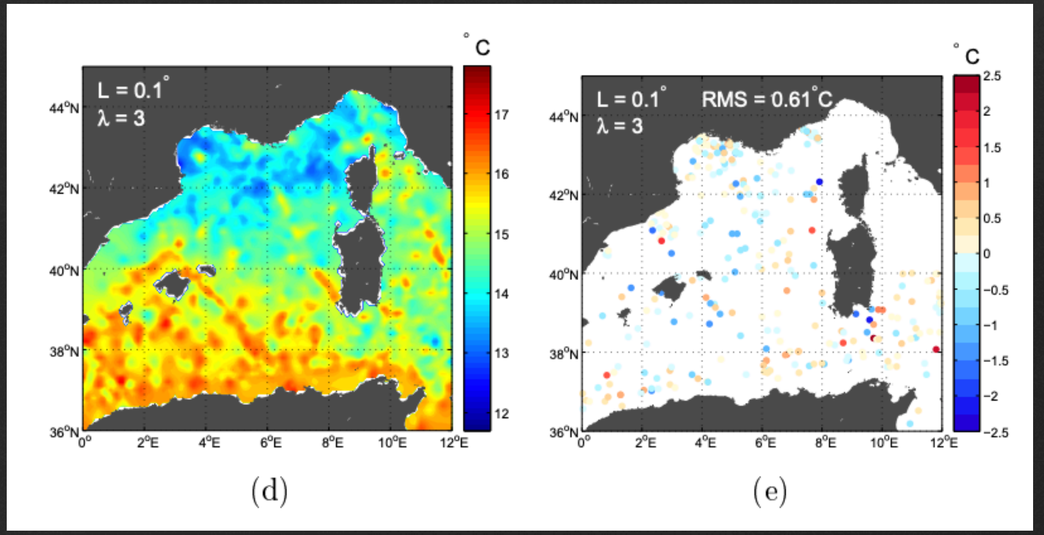


Figure 2.2: Interpolation (black line) provides a solution that goes across all the data points, while the approximation (grey line) has only to be "close" to the measurements, but with a relative "smoothness".

We are looking for the field φ which minimizes the variational principle over our domain of interest D:

$$J[\varphi] = \sum_{j=1}^{Nd} \mu_j \left[d_j - \varphi(x_j, y_j)\right]^2 + \|\varphi\|^2 \tag{2.10}$$

with

$$\|\varphi\| = \int_D \left(\alpha_2 \boldsymbol{\nabla}\boldsymbol{\nabla}\varphi : \boldsymbol{\nabla}\boldsymbol{\nabla}\varphi + \alpha_1 \boldsymbol{\nabla}\varphi \cdot \boldsymbol{\nabla}\varphi + \alpha_0 \varphi^2\right) dD \tag{2.11}$$

where

- α_0 penalizes the field itself (anomalies),
- α_1 penalizes gradients (no trends),
- α_2 penalizes variability (regularization),
- μ penalizes data-analysis misfits (objective).

To wrap up

- Combining errors requires care.
- If you have spatial or temporal structure, *you must* account for it, or statistical estimates, determination of trends, etc will be potentially misleading.
- Autocorrelation & spatial correlation act to reduce the number of effective data points.
- We have barely scratched the surface - this talk has ignored eg. Bayesian priors & much, much more.
- All talk materials can be found at link in QR code
- More maths after this slide for those interested!

Propagation of errors

$$\delta f = \sum_i \frac{\partial f}{\partial x_i} \delta x_i$$

$$\sigma^2(f) = \langle (\delta f)^2 \rangle$$

$$\sigma^2(f) = \sum_i \sum_j \frac{\partial f}{\partial x_i} \frac{\partial f}{\partial x_j} \langle \delta x_i \delta x_j \rangle$$

Deriving the propagation of errors

$$\sigma^2(f) = \sum_i \sum_j \frac{\partial f}{\partial x_i} \frac{\partial f}{\partial x_j} \langle \delta x_i \, \delta x_j \rangle$$

Assume no error covariance: $\langle \delta x_i \, \delta x_j \rangle = \langle \delta x_i \, \delta x_j \rangle \delta_{ij}$ where $\delta_{ij} = 1$ if $i = j$, 0 otherwise.

No error covariance: Coin flips, Dice rolls, Atomic Decays

$$\sigma^2(f) = \sum_i \left(\frac{\partial f}{\partial x_i} \right)^2 \sigma^2(x_i) = \sum_i \left(\frac{\partial f}{\partial x_i} \sigma_i \right)^2$$

Error covariance: Air Temperature, Rainy days,

Composite Quantities - eg. Concentrations

$$f(x) = g(x)\,h(x)$$

$$\delta f(x) = g(x)\,\delta h + h(x)\,\delta g$$

$$(\delta f(x))^2 = (g(x)\,\delta h)^2 + (h(x)\,\delta g)^2 + \cancel{2(h(x)\delta g)\,(\delta h g(x))}$$

$$\frac{\delta f(x)^2}{g(x)^2\,h(x)^2} = \frac{\delta h^2}{h^2} + \frac{\delta g^2}{g^2} = \frac{\delta f(x)^2}{f(x)^2}$$

$$\frac{\delta f}{f} = \sqrt{\left(\frac{\delta g}{g}\right)^2 + \left(\frac{\delta h}{h}\right)^2}$$

Fractional errors add in quadrature for composite quantities

An Example: Wind Direction Uncertainty

$$\sigma^2(f) = \sum_i \left(\frac{\partial f}{\partial x_i}\right)^2 \sigma^2(x_i) = \sum_i \left(\frac{\partial f}{\partial x_i}\sigma_i\right)^2$$

$$\sigma_\theta = \sqrt{\sum_i \left(\frac{\partial \theta}{\partial x_i}\sigma_i\right)^2} \quad \text{but } x_i = \theta, \text{ so } \sigma_\theta = \sqrt{\sum_i \sigma_i^2}$$

Independent uncertainties add in quadrature

Standard error of the mean

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i \quad \Longrightarrow \quad \sigma_{\bar{x}}^2 = \sum_{i=1}^{N}\left(\frac{\partial \bar{x}}{\partial x_i}\sigma_i\right)^2$$

$$\text{since } \frac{\partial \bar{x}}{\partial x_i} = \frac{1}{N}$$

$$\sigma_{\bar{x}}^2 = \frac{1}{N^2}\sum_{i=1}^{N}\sigma_i^2 \quad \Longrightarrow \quad \sigma_{\bar{x}} = \frac{\sigma}{\sqrt{N}} \text{ (if all } \sigma_i = \sigma)$$