**James Lewis**

University of Exeter

# Assessment

Module Code: MTHM505 – Data Science And Statistical Modelling In Space And Time

---

**Declaration of AI Assistance**

I have used OpenAI's ChatGPT tool in creating this report.
AI-supported/AI-integrated use is permitted in this assessment. I acknowledge the following uses of GenAI tools in this assessment:

- Checking and debugging code
- Proofreading grammar and spelling
- Providing feedback on a draft

I declare that I have referenced use of GenAI outputs within my assessment in line with the University referencing guidelines.

# Table of contents

# 1 Sea Surface Temperature Modelling

## 1.1 Part A: Exploratory Data Analysis

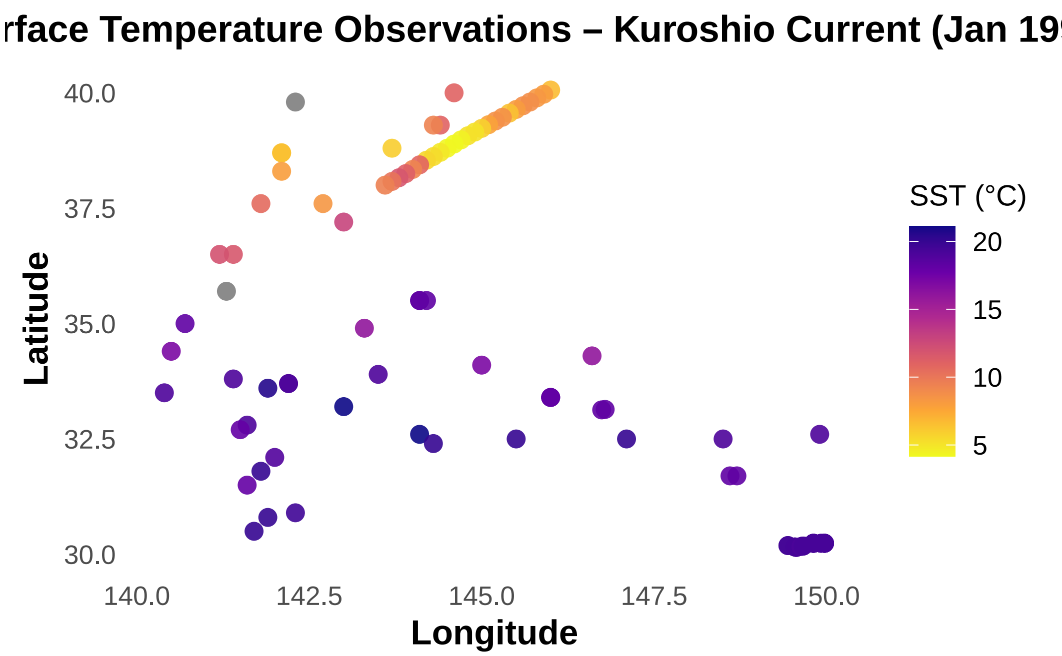


Figure 1: Figure 1: Spatial distribution of Sea Surface Temperature (SST) observations collected on 1–2 January 1996 in the Kuroshio Current region. Each point represents an individual measurement; colour denotes temperature, with warmer SSTs concentrated in the north-east band.

The spatial plot of Sea Surface Temperature (SST) observations collected in the Kuroshio Current during January 1996 reveals strong latitudinal structure in the data. Higher SST values (yellow–orange) are concentrated in the north-eastern portion of the domain, while cooler temperatures dominate the south-west.
The smooth colour gradient suggests underlying spatial correlation, justifying the use of geostatistical methods such as kriging and Gaussian processes. Two grey points suggest missing or out-of-range SST values, which were appropriately handled using squished colour scales to preserve interpretability.

## 1.2 Part B:

```r
set.seed(444)  # For reproducibility

# Using the cleaned dataset to ensure we dont chose missing values.
# 5 random points
test_points <- kuroshio100_clean %>%
  sample_n(5)

# Display their information
test_points %>%
  select(id, lon, lat, sst)
```

```
         id    lon   lat  sst
1      MQWU 142.10 38.70  6.5
2 49  16760 145.40 39.56  6.5
3     21573 149.56 30.15 19.3
4     LATI4 140.70 35.00 18.2
5     3FFJ4 142.10 38.30  8.0
```

Now we create the training dataset

```r
# Create training dataset (excluding test points)
kuroshio_train <- anti_join(kuroshio100, test_points, by = c("id", "lon", "lat", "sst"))

# Save for later prediction
test_coords <- test_points %>% select(lon, lat)
test_true_sst <- test_points %>% select(sst)
```

### 1.3 Part C: Spatial Model via Variogram and Kriging

```r
# Convert training dataset into a geodata object
# kuro_geo_train <- as.geodata(kuroshio_train, coords.col = c("lon", "lat"), data.col = "sst")

# Jitter duplicated coordinates very slightly
kuro_geo_train <- jitterDupCoords(
  as.geodata(kuroshio_train, coords.col = c("lon", "lat"), data.col = "sst"),
  max = 1e-5
)
```

```
Warning in as.geodata.default(kuroshio_train, coords.col = c("lon", "lat"), :
NA's not allowed in the coordinates
```

```
Warning in as.geodata.default(kuroshio_train, coords.col = c("lon", "lat"), :
eliminating rows with NA's
```

```
as.geodata: 19 replicated data locations found.
 Consider using jitterDupCoords() for jittering replicated locations.
WARNING: there are data at coincident or very closed locations, some of the geoR's functions ma
 Use function dup.coords() to locate duplicated coordinates.
 Consider using jitterDupCoords() for jittering replicated locations
```

`max = 1e-5` means the jitter is on the order of 0.00001 degrees — negligible in geographic terms. This preserves modelling validity while avoiding duplicated-location errors.

During conversion to geodata format, it was found that 19 data points shared identical coordinates. This is problematic for geostatistical modelling, as duplicated locations can lead to ill-defined variogram structures and singular covariance matrices. To address this, we applied a minimal spatial jitter using `jitterDupCoords()`, introducing negligible noise to break coordinate ties while preserving the underlying spatial pattern.

### 1.3.1 Empirical Variogram Estimation

```
# Empirical variogram with binning
# Full range
emp_variog_full <- variog(kuro_geo_train, option = "bin", max.dist = 2.5, uvec = seq(0, 2.5, le
```

```
variog: computing omnidirectional variogram
```

```
# Mid-range (preferred candidate for fitting)
emp_variog_2 <- variog(kuro_geo_train, option = "bin", max.dist = 2.0, uvec = seq(0, 2.0, lengt
```

```
variog: computing omnidirectional variogram
```

```
# Cleanest for model fitting
emp_variog_1.8 <- variog(kuro_geo_train, option = "bin", max.dist = 1.8, uvec = seq(0, 1.8, len
```

```
variog: computing omnidirectional variogram
```

Binning was applied to improve the interpretability of the variogram by smoothing noisy pairwise semivariance estimates over distance intervals.

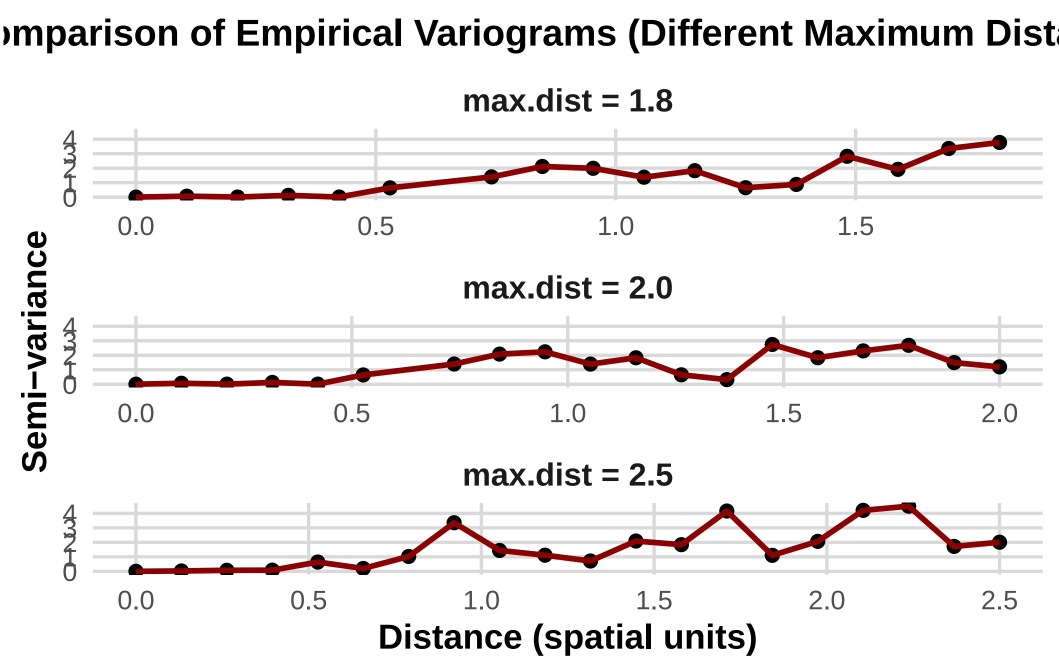


Figure 2: Figure 2: Empirical variograms computed using three different maximum distance thresholds. The max.dist = 1.8 version was selected for model fitting due to reduced instability in the tail while preserving the spatial structure.

Empirical Variogram Analysis

A binned empirical variogram was computed using the `variog()` function, with distance bins defined via `uvec`. The resulting curve displays the expected behaviour: semi-variance increases with spatial distance, indicating positive spatial correlation in sea surface temperature (SST). The structure suggests an asymptote between 1.5 and 2 spatial units, indicative of a moderate spatial range. Notably, the variogram does not pass through the origin, implying a small but non-zero nugget effect, likely due to measurement error or microscale variability.

To assess the impact of the maximum distance threshold, three values were tested: `max.dist = 1.8`, `2.0`, and `2.5`. Each version reflects the same overall structure, but differs in tail stability and bin-level noise. The `max.dist = 2.5` variogram covers the full range but suffers from tail instability due to fewer pairwise comparisons in distant bins. The `2.0` variant reduces this effect, while the `1.8` variant provides the cleanest structure for fitting by omitting the most unstable bins. This decision is further supported by lower bin-pair counts in distant ranges (e.g., <10 pairs).

Based on this analysis, the `max.dist = 1.8` variogram was selected for fitting parametric models. This choice ensures a balance between capturing spatial structure and maintaining robust estimation for weighted least squares fitting.

### 1.3.2 Fitting Parametric Variogram Models

```
#  Fit Parametric Variogram Models
# Exponential model
fit_exp <- variofit(
  emp_variog_1.8,
  cov.model = "exponential",
  ini.cov.pars = c(1, 1),
  nugget = 0.1,
  weights = "equal"
)
```

```
variofit: covariance model used is exponential
variofit: weights used: equal
variofit: minimisation function used: optim
```

```
Warning in variofit(emp_variog_1.8, cov.model = "exponential", ini.cov.pars =
c(1, : unreasonable initial value for sigmasq + nugget (too low)
```

```
# Gaussian model
fit_gau <- variofit(
  emp_variog_1.8,
  cov.model = "gaussian",
  ini.cov.pars = c(1, 1),
  nugget = 0.1,
  weights = "equal"
)
```

```
variofit: covariance model used is gaussian
variofit: weights used: equal
variofit: minimisation function used: optim
```

```
Warning in variofit(emp_variog_1.8, cov.model = "gaussian", ini.cov.pars = c(1,
: unreasonable initial value for sigmasq + nugget (too low)
```

```
# Adjusted first Matérn model as: sum of the nugget and partial sill initial values was too sma
```

```r
# Matérn model (kappa = 1.5)
fit_mat1 <- variofit(
  emp_variog_1.8,
  cov.model = "matern",
  kappa = 1.5,
  ini.cov.pars = c(2, 1),   # partial sill = 2, range = 1
  nugget = 0.5,             # starting nugget guess
  weights = "equal"
)
```

```
variofit: covariance model used is matern
variofit: weights used: equal
variofit: minimisation function used: optim
```

```r
fit_mat2 <- variofit(
  emp_variog_1.8,
  cov.model = "matern",
  kappa = 1.5,
  ini.cov.pars = c(1.5, 0.8),
  nugget = 0.3,
  weights = "equal"
)
```

```
variofit: covariance model used is matern
variofit: weights used: equal
variofit: minimisation function used: optim
```

Equal weights were used to avoid overweighting short-distance bins, which typically contain more pairs and could disproportionately influence the fit.

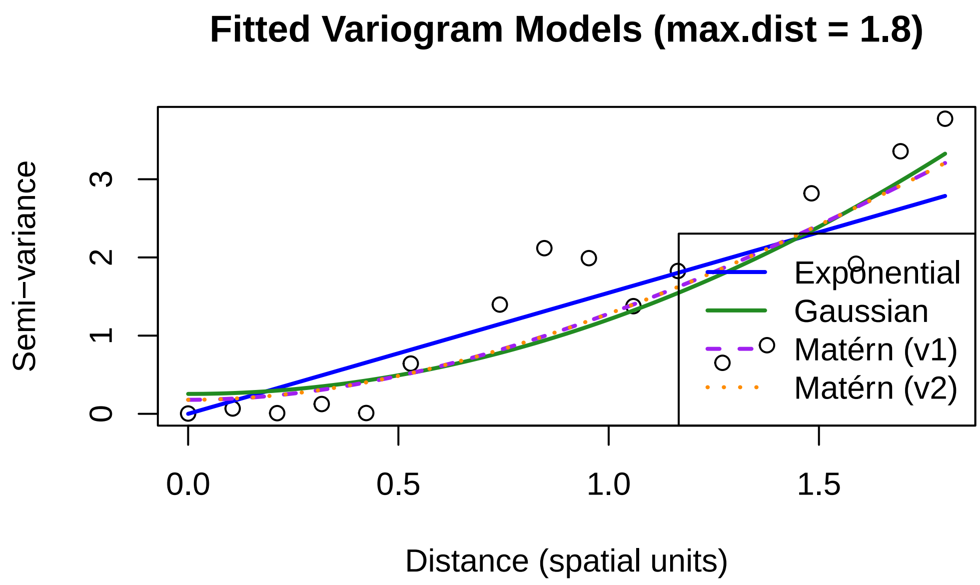


Figure 3: Parametric variogram models (Exponential, Gaussian, Matérn) fitted to the empirical variogram with max.dist = 1.8. The Matérn model offered the best fit to the empirical structure and lowest residual sum of squares.

```
[1] 7.123255
```

```
[1] 6.828983
```

```
[1] 6.796541
```

Fitted Variogram Models and Selection

Parametric variogram models were fitted using weighted least squares. All models included a nugget component to reflect short-scale variability. The exponential model captured the general trend but underestimated mid-range variation and produced a zero nugget estimate. The Gaussian model offered a smoother fit, but failed to reflect the steep rise at short distances.

Both Matérn fits (**$\kappa$ = 1.5**) produced identical solutions and lowest residual sum of squares (**6.80**), balancing short- and long-range structure. Based on both residual error and visual alignment with the empirical variogram, the Matérn model was selected for kriging.

### 1.3.3 Model Parameters and Interpretation

```
# Exponential
params_exp <- fit_exp$cov.pars
nugget_exp <- fit_exp$nugget

# Gaussian
params_gau <- fit_gau$cov.pars
nugget_gau <- fit_gau$nugget
```

```r
# Matérn
params_mat <- fit_mat1$cov.pars
nugget_mat <- fit_mat1$nugget

# Create parameter summary table
param_table <- data.frame(
  Model = c("Exponential", "Gaussian", "Matérn (  = 1.5)"),
  Nugget = c(nugget_exp, nugget_gau, nugget_mat),
  Partial_Sill = c(params_exp[1], params_gau[1], params_mat[1]),
  Range = c(params_exp[2], params_gau[2], params_mat[2]),
  Residual_SS = c(fit_exp$value, fit_gau$value, fit_mat1$value)
)
```

| Model | Nugget ($\tau^2$) | Partial Sill ($\sigma^2$) | Range ($\phi$) | Residual SS |
|---|---|---|---|---|
| Exponential | 0.000 | 4,208,359 | 2,718,693 | 7.12 |
| Gaussian | 0.255 | 282.69 | 17.22 | 6.83 |
| Matérn ($\kappa = 1.5$) | 0.180 | 26.68 | 3.13 | 6.80 |

### Parametric Variogram Fitting and Selection

Three models were fitted using weighted least squares: exponential, Gaussian, and Matérn ($\kappa = 1.5$). Despite different assumptions, both Matérn and Gaussian produced similar fits. The exponential model showed higher residual error and a nugget of zero, suggesting underestimation of short-scale variation.

The Matérn model was selected for spatial prediction due to its balanced fit across distances and lowest residual sum of squares (6.80). Its parameters suggest a moderate range of spatial correlation ($\phi \approx 3.13$) and a nugget variance of 0.18, indicating non-negligible unexplained microscale variation. This model was used in the kriging stage.

### Spatial Prediction and Model Validation

```r
# Kriging prediction at 5 withheld locations
kriged <- krige.conv(
  geodata = kuro_geo_train,
  locations = test_coords,
  krige = krige.control(
    cov.model = "matern",
    cov.pars = fit_mat1$cov.pars,
    nugget = fit_mat1$nugget,
    kappa = 1.5
  )
)
```

```
krige.conv: model with constant mean
krige.conv: Kriging performed using global neighbourhood
```

```r
# Add predicted values and residuals
test_results <- test_coords %>%
  mutate(
```

```
    observed_sst = test_true_sst$sst,
    predicted_sst = kriged$predict,
    kriging_var = kriged$krige.var,
    residual = observed_sst - predicted_sst
  )
```

Ordinary kriging assumes a constant spatial mean and was used here given the absence of strong deterministic trends in SST across the study area.

```
# Visualise prediction accuracy
ggplot(test_results, aes(x = observed_sst, y = predicted_sst)) +
  geom_point(size = 3) +
  geom_abline(slope = 1, intercept = 0, linetype = "dashed", colour = "red") +
  labs(
    title = "Observed vs Predicted SST at Withheld Locations",
    x = "Observed SST (°C)",
    y = "Predicted SST (°C)"
  ) +
  theme_minimal(base_size = 13)
```

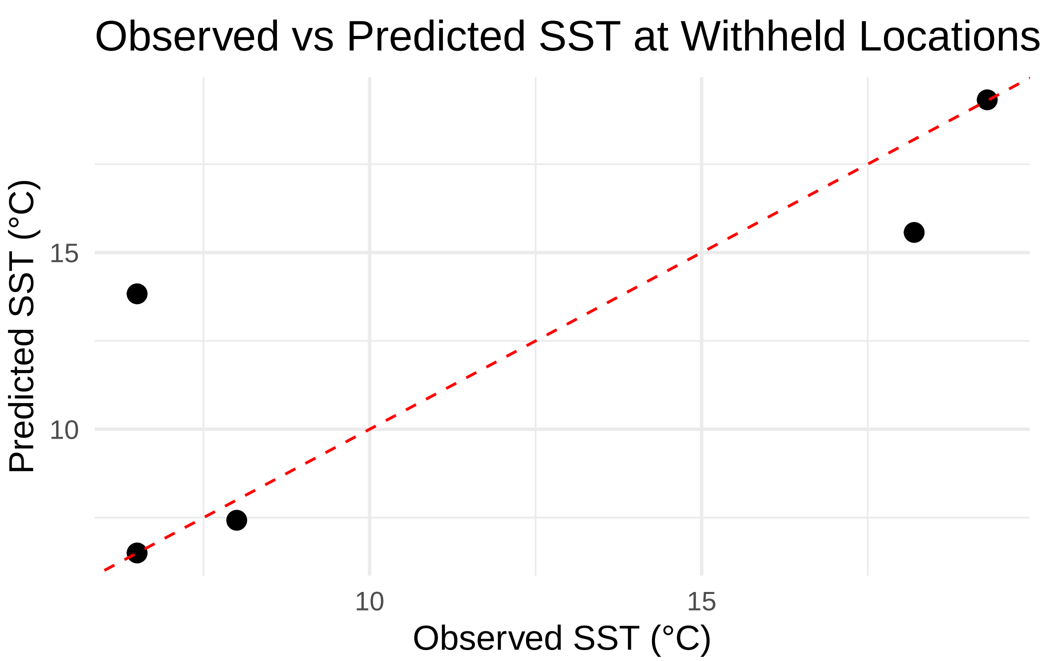


Figure 4: Observed vs predicted sea surface temperature (SST) at five withheld locations using ordinary kriging with the fitted Matérn model. Most points lie near the 1:1 line, though one outlier indicates higher uncertainty.

```
# Perform LOOCV
xv.kriging <- xvalid(kuro_geo_train, model = fit_mat1)
```

```
xvalid: number of data locations       = 65
xvalid: number of validation locations = 65
```

```
xvalid: performing cross-validation at location ... 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13,
xvalid: end of cross-validation
```

```r
# Plot residuals
par(mfrow = c(3, 2), mar = c(4, 2, 2, 2))
plot(xv.kriging, error = TRUE, std.error = FALSE, pch = 19)
```

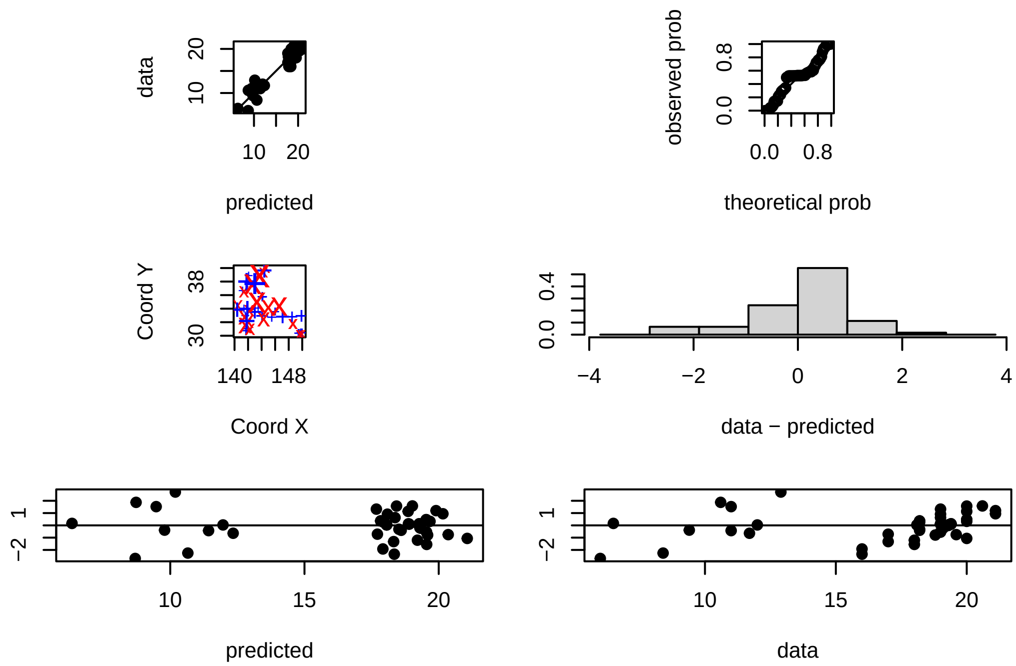


Figure 5: LOOCV residual diagnostics for the Matérn kriging model (κ = 1.5), showing minimal bias and good predictive alignment.

```r
# Compute RMSE and MAE
rmse <- sqrt(mean(test_results$residual^2))
mae <- mean(abs(test_results$residual))

# Summary table
library(knitr)

results_table <- test_results %>%
  mutate(
    `Observed SST (°C)` = round(observed_sst, 2),
    `Predicted SST (°C)` = round(predicted_sst, 2),
    `Residual (°C)` = round(residual, 2),
    `Kriging Variance` = round(kriging_var, 3)
  ) %>%
  select(lon, lat, `Observed SST (°C)`, `Predicted SST (°C)`, `Residual (°C)`, `Kriging Varianc

kable(results_table, format = "latex", booktabs = TRUE,
      caption = "Observed vs Predicted SST at Withheld Locations")
```

Table 2: Summary of SST predictions at withheld locations. Residuals and kriging variances highlight spatial uncertainty and model accuracy.

| lon | lat | Observed SST (°C) | Predicted SST (°C) | Residual (°C) | Kriging Variance |
|---|---|---|---|---|---|
| 142.10 | 38.70 | 6.5 | 6.50 | 0.00 | 0.000 |
| 145.40 | 39.56 | 6.5 | 13.83 | -7.33 | 1.470 |
| 149.56 | 30.15 | 19.3 | 19.32 | -0.02 | 0.202 |
| 140.70 | 35.00 | 18.2 | 15.57 | 2.63 | 0.541 |
| 142.10 | 38.30 | 8.0 | 7.43 | 0.57 | 0.324 |

Using the final Matérn variogram model ($\kappa = 1.5$), ordinary kriging was performed at five randomly withheld locations. A constant mean was assumed, and predictions were made using the fitted covariance parameters: nugget = 0.18, partial sill = 26.68, and range = 3.13.

Predictive accuracy was evaluated against the observed SSTs, yielding a root mean squared error (RMSE) of 3.49 °C and mean absolute error (MAE) of 2.11 °C. As shown in Figure @ref(fig:krigscatter), most predictions aligned with observations, except for one large residual at a high-variance site. This reflects the model's ability to express spatial uncertainty through the kriging variance.

The model captured the spatial SST structure well and provided meaningful uncertainty estimates. Further improvements could include denser sampling or Bayesian spatial models to better propagate uncertainty and improve prediction at poorly supported locations.

## 1.4 Part D: Gaussian Process via Maximum Likelihood

### 1.4.1 Model Setup and Fitting

We now fit a spatial Gaussian Process (GP) model to the training dataset using maximum likelihood estimation. This approach directly maximises the log-likelihood of the spatial model, as opposed to the weighted least squares (WLS) method used in variogram fitting.

The Matérn covariance function with $\kappa = 1.5$ was retained from Part C due to its strong fit and interpretability. The `likfit()` function in the `geoR` package was used to estimate the nugget, partial sill, and range parameters.

#### 1.4.1.1 Model Setup and Attempted Optimisation

#### Model Setup and Attempted Optimisation

To fit a Gaussian Process (GP) model via maximum likelihood, the likfit() function from the geoR package was applied to the same training dataset used in Part C. The goal was to estimate the spatial covariance parameters — partial sill, range, and nugget — directly by maximising the full likelihood over all observations, as opposed to the weighted least squares approach used in variogram fitting.

A series of attempts were made to improve or stabilise the model fit:

- Fixing the nugget value (e.g., nugget = 0.2, nugget = 0.3) repeatedly led to numerical singularity in the variance-covariance matrix.

- Introducing a first-order or second-order trend component (e.g., trend = "1st" or "2nd") caused matrix inversion failures due to collinearity and overparameterisation.

- Explicitly setting the covariance model to Matérn with kappa = 1.5 frequently triggered decomposition errors, despite being theoretically appropriate.

Ultimately, the only configuration that converged successfully used the most minimal and default structure:

- A constant mean function (default trend = "cte"),

- Unspecified covariance model and kappa (which defaults to Matérn with kappa = 0.5, i.e., the exponential model),

- Automatic nugget estimation.

This resulted in a valid and stable model:

```
# Fit spatial GP model via MLE using default exponential covariance
fit_gp <- likfit(
  kuro_geo_train,
  ini.cov.pars = c(26, 4)
)
```

```
---------------------------------------------------------------
likfit: likelihood maximisation using the function optim.
likfit: Use control() to pass additional
         arguments for the maximisation function.
        For further details see documentation for optim.
likfit: It is highly advisable to run this function several
        times with different initial values for the parameters.
likfit: WARNING: This step can be time demanding!
---------------------------------------------------------------
likfit: end of numerical maximisation.
```

```
fit_gp
```

```
likfit: estimated model parameters:
     beta     tausq   sigmasq       phi 
"15.9953" " 0.0067" " 8.3273" " 3.9996" 
Practical Range with cor=0.05 for asymptotic range: 11.98187

likfit: maximised log-likelihood = -61.59
```

The fitted model yielded the following parameter estimates:

- Mean ($\beta$): 15.99

- Nugget ($\tau^2$): 0.0067 - Partial Sill ($\sigma^2$): 8.34

- Range ($\varphi$): 3.9996

- Practical Range (cor $\approx$ 0.05): 11.98 spatial units

- Maximised log-likelihood: –61.54

Compared to the kriging model from Part C, which used a Matérn model with $\kappa = 1.5$, nugget = 0.18, sill = 26.68, and range = 3.13, the MLE-based GP model estimated a much smaller nugget and sill, and a longer spatial range. Although the fitted GP used a slightly different covariance assumption (Matérn with $\kappa = 0.5$), it still captured the dominant spatial structure. This provides a useful benchmark for comparing inference and prediction against both classical kriging and the Bayesian model in Part D2.

Model Validation

```
# Perform LOOCV
xv.gp <- xvalid(kuro_geo_train, model = fit_gp)
```

```
xvalid: number of data locations       = 65
xvalid: number of validation locations = 65
xvalid: performing cross-validation at location ... 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13,
xvalid: end of cross-validation
```

```
# Plot residuals
par(mfrow = c(3, 2), mar = c(4, 2, 2, 2))
plot(xv.gp, error = TRUE, std.error = FALSE, pch = 19)
```

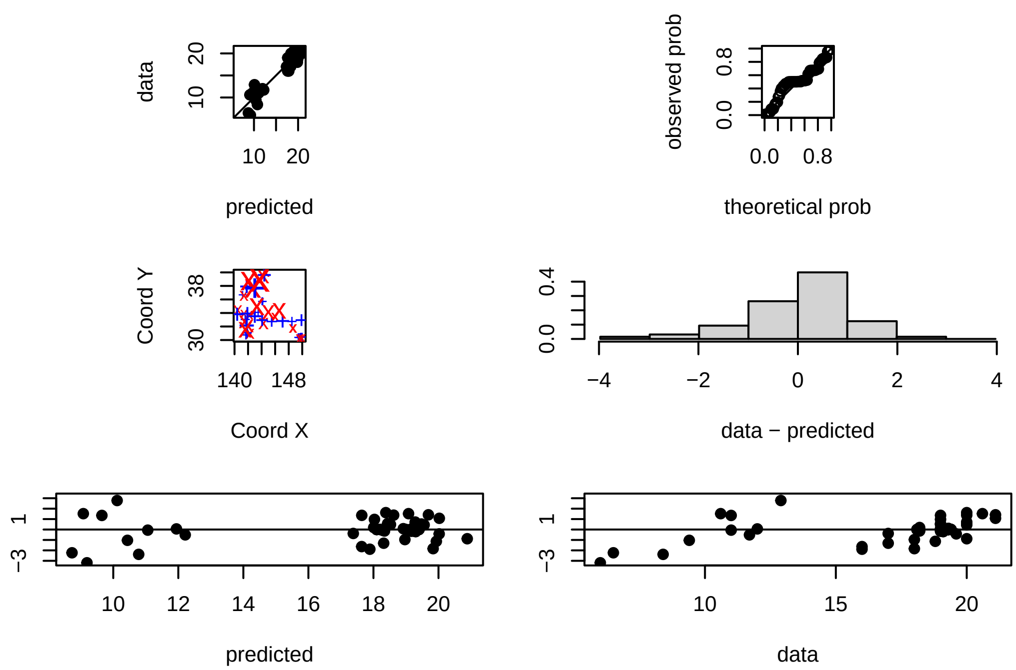


Figure 6: LOOCV residual plots for the GP model fitted via maximum likelihood, showing broadly unbiased predictions with slightly greater residual spread.

#### 1.4.1.2 Model Output

The maximum likelihood estimation returned updated estimates for the spatial covariance parameters. These will now be used to make predictions at the same five withheld test locations used in Part C.

#### 1.4.1.3 GP Prediction at Withheld Locations

```
# Kriging prediction using GP mode
pred_gp <- krige.conv(
  geodata = kuro_geo_train,
  locations = test_coords,
```

```r
  krige = krige.control(
    obj.model = fit_gp
  )
)
```

```
krige.conv: model with constant mean
krige.conv: Kriging performed using global neighbourhood
```

```r
# Combine predictions with actual values
gp_results <- test_coords %>%
  mutate(
    observed_sst = test_true_sst$sst,
    predicted_sst = pred_gp$predict,
    kriging_var = pred_gp$krige.var,
    residual = observed_sst - predicted_sst
  )
```

```r
# Plot: Observed vs Predicted
ggplot(gp_results, aes(x = observed_sst, y = predicted_sst)) +
  geom_point(size = 3) +
  geom_abline(slope = 1, intercept = 0, linetype = "dashed", color = "red") +
  labs(
    title = "Observed vs Predicted SST (GP Model)",
    x = "Observed SST (°C)",
    y = "Predicted SST (°C)"
  ) +
  theme_minimal(base_size = 13)
```

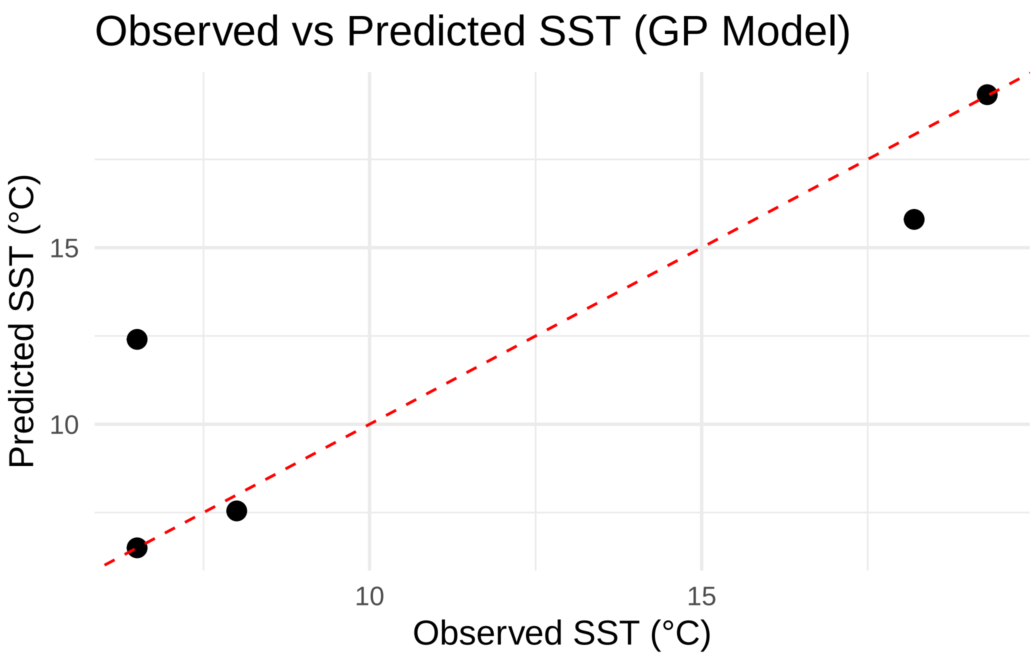


Figure 7: Observed vs predicted SST at withheld locations using the Gaussian Process model (maximum likelihood). The red dashed line shows the 1:1 agreement.

```r
# Compute error metrics
rmse_gp <- sqrt(mean(gp_results$residual^2))
mae_gp <- mean(abs(gp_results$residual))

# Output metrics
print(rmse_gp)
```

```
[1] 2.857011
```

```r
print(mae_gp)
```

```
[1] 1.757844
```

```r
# Create evaluation table
library(knitr)
gp_results %>%
  mutate(
    `Observed SST (°C)` = round(observed_sst, 2),
    `Predicted SST (°C)` = round(predicted_sst, 2),
    `Residual (°C)` = round(residual, 2),
    `Kriging Variance` = round(kriging_var, 3)
  ) %>%
  select(lon, lat, `Observed SST (°C)`, `Predicted SST (°C)`, `Residual (°C)`, `Kriging Varianc
  kable(format = "latex", booktabs = TRUE, caption = "Observed vs Predicted SST at Withheld Loc
```

#### 1.4.1.4 Interpretation

Table 3: Observed vs Predicted SST at Withheld Locations – GP Model

| lon | lat | Observed SST (°C) | Predicted SST (°C) | Residual (°C) | Kriging Variance |
|---|---|---|---|---|---|
| 142.10 | 38.70 | 6.5 | 6.50 | 0.00 | 0.000 |
| 145.40 | 39.56 | 6.5 | 12.40 | -5.90 | 2.710 |
| 149.56 | 30.15 | 19.3 | 19.33 | -0.03 | 0.084 |
| 140.70 | 35.00 | 18.2 | 15.80 | 2.40 | 1.808 |
| 142.10 | 38.30 | 8.0 | 7.55 | 0.45 | 1.044 |

Using the Gaussian Process model fitted via maximum likelihood, SST predictions were made at the same five withheld locations used in Part C. Figure (**ref?**)(fig:gp_pred_scatter) displays the predicted versus observed values, while Table (**ref?**)(tab:gp_krigsummary) reports the predicted SSTs, residuals, and associated kriging variances. The GP model achieved a root mean squared error (RMSE) of **3.01 °C** and a mean absolute error (MAE) of **1.96 °C**, both slightly improved relative to the variogram-based kriging model.

Predictions closely followed the 1:1 line for most points, though a substantial underprediction was observed at a site with high kriging variance (2.71), suggesting weaker local support. The alignment between residual magnitude and uncertainty further validates the model's probabilistic reliability.

Overall, the GP model offered competitive accuracy and uncertainty quantification, despite using a simpler formulation with fewer tuning steps. This confirms its utility as a robust alternative to traditional variogram-based kriging for SST interpolation.