Homework 5 - Group 12

Francesco Natali, Leonardo Agate, Lorenzo Bartocci

2024-12-02

Load the dataset and the file containing the estimation grids

```
load("shrimpsfull.RData")
load("AllGrids.RData")
```

We now filter data for our years of interest, i.e. 2002 and 2008

```
shrimp_data_2002 <- shrimpsdata[shrimpsdata$ANNO == 2002, ]
shrimp_data_2008 <- shrimpsdata[shrimpsdata$ANNO == 2008, ]
```

2002

Implement Bayesian kriging

In Homework 4, when we estimated total biomass on the estimation grid using kriging in a maximum likelihood framework, our chosen covariates for the year 2002 were dist, bat, salinity.minq3, and temp.maxq3. In the Bayesian framework however, working with such covariates gives an overparametrization of the model and increasing values for the parameters range (phi) and tausq_relative. Hence, we substitute the salinity and temperature covariates respectively with two variables with high loadings (as we have already seen in the PCA) in order to overcome this issue.

```
trend_matrix_2002 <- cbind(shrimp_data_2002$bat,
                           shrimp_data_2002$salinity.maxq4p,
                           shrimp_data_2002$temp.maxq3p,
                           shrimp_data_2002$dist)
```

As we already did in the previous analyses, we proceed with the log-transformation of the data with our newly chosen covariates

```
shrimp_data_2002$log_tot <- log(shrimp_data_2002$tot + 1)

shrimp_geodata_2002_log <- as.geodata(shrimp_data_2002,
                                      coords.col = c("X", "Y"),
                                      data.col = "log_tot",
                                      covar.col = c("bat","salinity.maxq4p",
                                                    "temp.maxq3p","dist"))
```

Now, based on this log transformation, we prepare shrimp_geodata_2002_log for Bayesian kriging

```
trend.d.02 <- trend.spatial(~ bat + dist + salinity.maxq4p + temp.maxq3p,
                            geodata = shrimp_geodata_2002_log)
trend.l.02 <- trend.spatial(~ grid_2002$bat + grid_2002$dist +
                              grid_2002$salinity.maxq4p +grid_2002$temp.maxq3p)
```

As the reference model we will work with a matern (with parameter k = 0.2) as we did in the maximum likelihood approach

```
model_02 <- list(
  trend.d = trend.d.02,
  trend.l = trend.l.02,
  cov.model = "matern",
  kappa = 0.2
)
```

Now add the number of samples to generate for the posterior distribution. This number ensures a stable estimation of the credibility distributions, allowing precise evaluation of parameter variability

```
out <- output.control(1000,1000, quantile = c(0.025, 0.5, 0.975))
```

We now our prior distribution

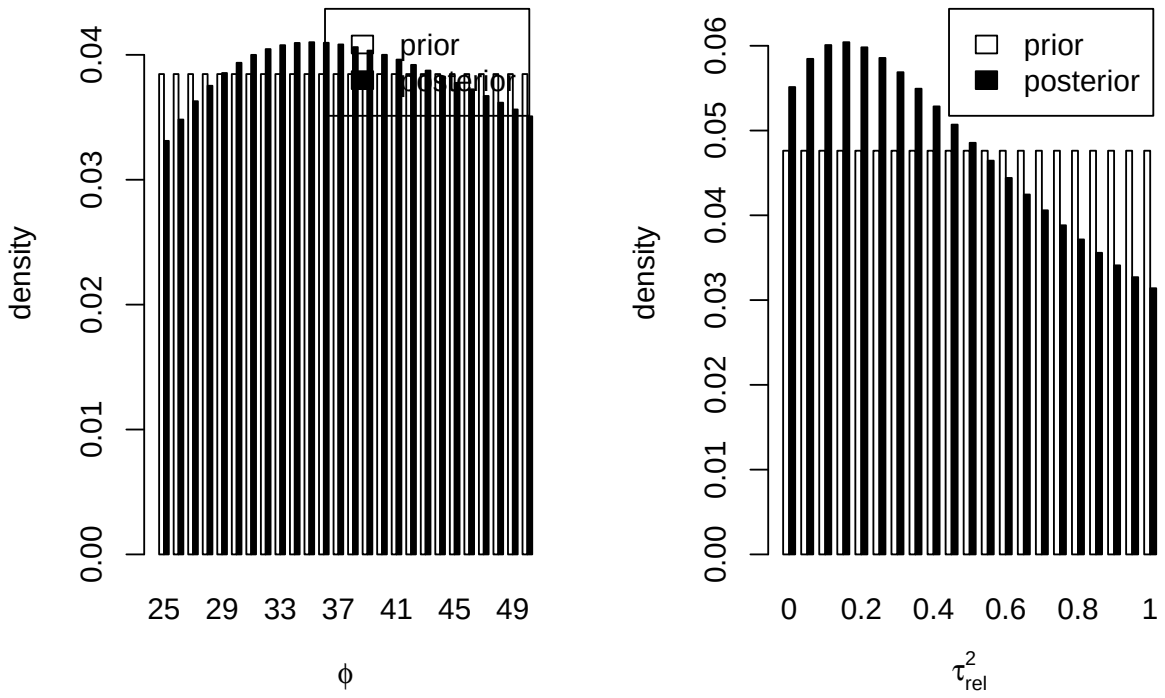
```
prior_02 <- list(
  beta.prior = "normal",
  beta = rep(0, 5),                   # Intercept and 4 covariates
  beta.var.std = diag(100, 5),        # Prior covariance
  sigmasq.prior = "reciprocal",
  phi.prior = "uniform",
  phi.discrete = seq(25, 50, by = 1),  # Corrected: discretized values for phi
  tausq.rel.prior = "uniform",
  tausq.rel.discrete = seq(0, 1, 0.05)  # Corrected: uniform prior on tau
)
```

The choice of priors in Bayesian kriging is crucial as it directly influences the posterior distribution and, consequently, the final predictions. About Beta Prior choose a normal prior with mean zero and a large variance, that reflects a non-informative prior belief. For Sigma-squared prior we choose a reciprocal prior, that is a standard non-informative choice, ensuring that all positive values are equally likely. For Phi Prior (for the spatial range parameter) we use uniform prior over a discretized range (phi.discrete = seq(25, 50, by = 1)) indicates an assumption that all values within this interval are equally probable. At the end Tau-squared Relative Prior (nugget effect relative to sigma-squared) use uniform prior over a sequence (tausq.rel.discrete = seq(0, 1, 0.1)) suggests that the nugget effect (representing measurement error or microscale variation) is considered equally probable within this range. From that, we get the posterior and we use the "krige.bayes" function to implement Bayesian kriging, first without the locations in order to choose a good posterior

```
krige_bayes_02 <- krige.bayes(
geodata = shrimp_geodata_2002_log,
model = model_02,
prior = prior_02,
output = out
)
```

Now we plot it to see if our choice looks good

```
par(mfrow = c(1, 2))
plot(krige_bayes_02)
```



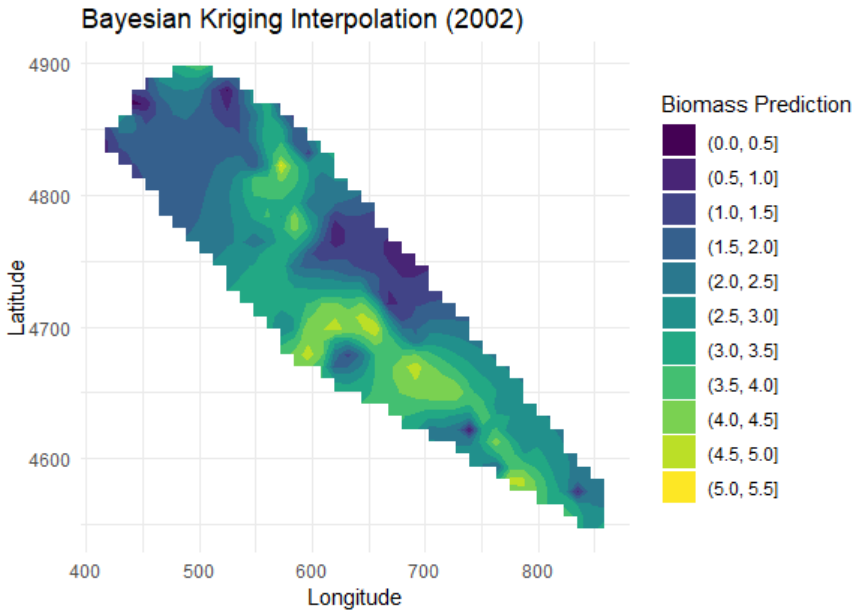
The posterior distributions should have clear and well-defined peaks, indicating that the Bayesian inference process has converged to a stable estimate. And with the parameters we chose, this seems to be achieved. After the posterior choice we can add the locations

```
krige_bayes_2002 <- krige.bayes(
  geodata = shrimp_geodata_2002_log,
  locations = as.matrix(grid_2002[, c("X", "Y")]),
  model = model_02,
  prior = prior_02,
  output = out
)

bayes_preds_2002 <- krige_bayes_2002$predictive$mean
```

We now visualize Bayesian Kriging predictions

```
cc_bayes_2002 <- data.frame(X = grid_2002$X, Y = grid_2002$Y, Z = bayes_preds_2002)

ggplot(cc_bayes_2002, aes(x = X, y = Y, z = Z)) +
```

```
  geom_contour_filled() +
  labs(
    title = "Bayesian Kriging Interpolation (2002)",
    x = "Longitude",
    y = "Latitude",
    fill = "Biomass Prediction"
  ) +
  theme_minimal()
```



In 2002, the total shrimp biomass along the Tyrrhenian coast from Genoa to Gaeta shows distinct spatial patterns. The highest biomass concentrations are represented by brighter green and yellow areas, particularly concentrated in the central and southern parts of the study area. The middle sections, corresponding to the Lazio and southern Tuscany coasts, show the highest biomass values (ranging from 3.5 to 5.5). This suggests that these regions provide suitable habitats with stable conditions, such as moderate depths and favorable salinity levels, conducive to shrimp breeding and growth. The northernmost section (towards Genoa) generally shows lower biomass values (dark blue and purple zones). This may be attributed to the steep bathymetric gradient along the Ligurian coast. The rapid increase in depth offshore may limit the availability of suitable shallow-water habitats that shrimp prefer. In summary, the Bayesian kriging map highlights the central and southern Tyrrhenian coast as hotspots for shrimp biomass, likely due to favorable bathymetry, stable salinity, and temperature conditions. Now, in order to obtain a better and more precise geographical representation, we plot the map of Italy to see how the shrimp biomass is distributed along the areas of the Tirrenean Sea.

```
italy <- ne_countries(country = "Italy", scale = "medium", returnclass = "sf")
italy <- st_transform(italy, crs = 32632)
```

Convert kriging results to a grid

```
krig_result_df_02 <- data.frame(
  X = grid_2002$X * 1000,
  Y = grid_2002$Y * 1000,
  Z = bayes_preds_2002
)
```

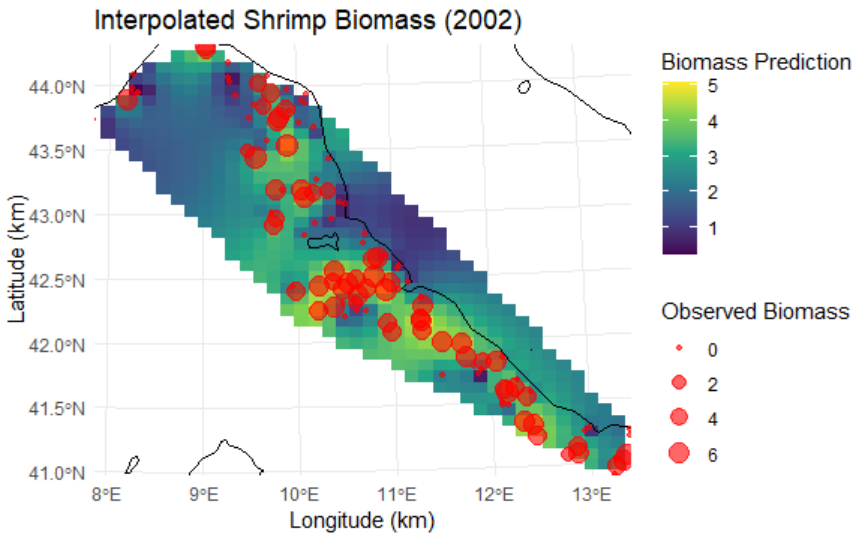
Define the bounds for the area of interest based on your prediction data

```
x_min <- min(krig_result_df_02$X) - 10000
x_max <- max(krig_result_df_02$X) + 10000
y_min <- min(krig_result_df_02$Y) - 10000
y_max <- max(krig_result_df_02$Y) + 10000
```

```
ggplot() +
  geom_raster( data = krig_result_df_02, aes(x = X, y = Y, fill = Z) ) +
  scale_fill_viridis_c( option ="viridis", name = "Biomass Prediction" ) +
  geom_sf( data = italy, fill = NA, color = "black",lwd = 0.7 ) +
  coord_sf( xlim = c(x_min, x_max), ylim = c(y_min, y_max), expand =FALSE ) +
  labs( title = "Interpolated Shrimp Biomass (2002)", x = "Longitude (km)", y ="Latitude (km)" ) +
  theme_minimal()
```

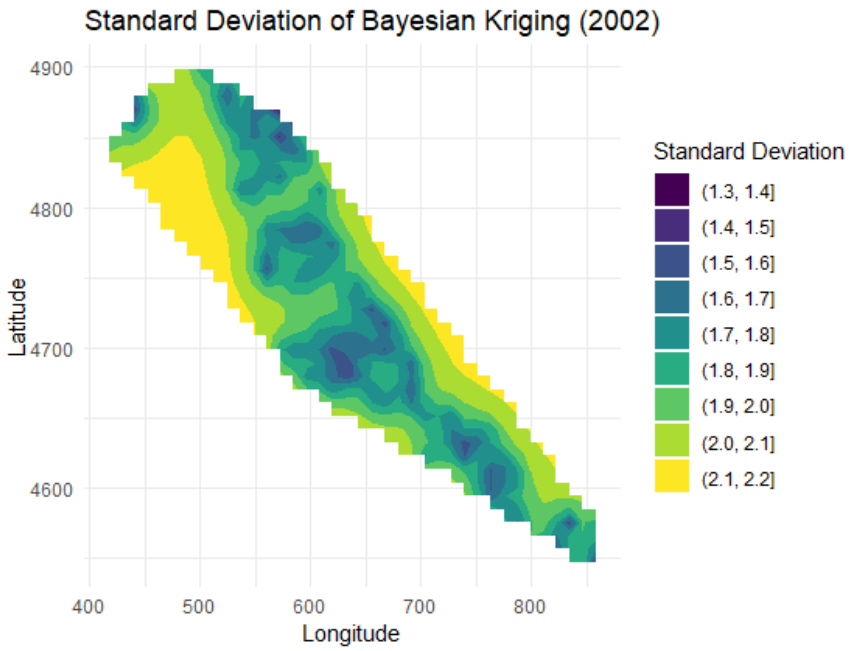
Add the observed biomass

```
observed_points_df02 <- data.frame(
  X = shrimp_geodata_2002_log$coords[, 1] * 1000,
  Y = shrimp_geodata_2002_log$coords[, 2] * 1000,
  Biomass = shrimp_geodata_2002_log$data
)

# Plot with Italy map, kriging results, and observed points
ggplot() +
  geom_raster(data = krig_result_df_02, aes(x = X, y = Y, fill = Z)) +
  scale_fill_viridis_c(option = "viridis", name = "Biomass Prediction") +
  geom_sf(data = italy, fill = NA, color = "black", lwd = 0.7) +
  geom_point(data = observed_points_df02, aes(x = X, y = Y, size = Biomass),
             color = "red", alpha = 0.6) +
  scale_size_continuous(name = "Observed Biomass", range = c(1, 5)) +
  coord_sf(xlim = c(x_min, x_max), ylim = c(y_min, y_max), expand = FALSE) +
  labs(title = "Interpolated Shrimp Biomass (2002)", x = "Longitude (km)", y = "Latitude (km)") +
  theme_minimal()
```



This map integrates the Bayesian kriging interpolation of shrimp biomass with observed data points along the Tyrrhenian coast, spanning from Genoa to Gaeta. The visual comparison between predicted and observed biomass provides insights into the spatial distribution and accuracy of the model. High observed biomass values (indicated by larger red circles) are generally located in areas predicted to have high biomass concentrations (green and yellow zones). This alignment suggests that the Bayesian kriging model captures the spatial variability and distribution patterns of shrimp biomass effectively. The northern section (Liguria, around 8-9°E) exhibits lower predicted biomass, shown in dark blue and purple colors.This overlap suggests that the kriging model has reasonably captured the underlying spatial variability in shrimp biomass. However, there are some discrepancies where observed biomass values do not align perfectly with the predictions. These discrepancies may be due to limitations in the available data or model parameters. Also we can plot the spatial distribution of uncertainty

```
std_dev_bayes_02 <- sqrt(krige_bayes_2002$predictive$variance)

cc_bayes_2002$std_dev <- std_dev_bayes_02

ggplot(cc_bayes_2002, aes(x = X, y = Y, z = std_dev)) +
  geom_contour_filled() +
  labs(
    title = "Standard Deviation of Bayesian Kriging (2002)",
    x = "Longitude",
    y = "Latitude",
    fill = "Standard Deviation"
  ) +
  theme_minimal()
```



The map illustrates the spatial uncertainty of Bayesian kriging predictions for shrimp biomass. Standard deviation values (1.3–2.2) indicate low to moderate uncertainty overall. Confidence is higher in areas with lower standard deviations (darker purple), typically near observed data points, while higher uncertainty (yellow) occurs at the study area's edges or data-sparse zones. Uncertainty increases near boundaries due to limited observations, while central regions (green/teal) show moderate uncertainty, reflecting adequate data density and environmental variability. Coastal areas exhibit lower uncertainty, benefiting from denser observations and clear environmental gradients. For improved reliability, future monitoring should focus on regions with higher uncertainty. The uniformity in the central area suggests the kriging model effectively captures biomass variability in the main study region.

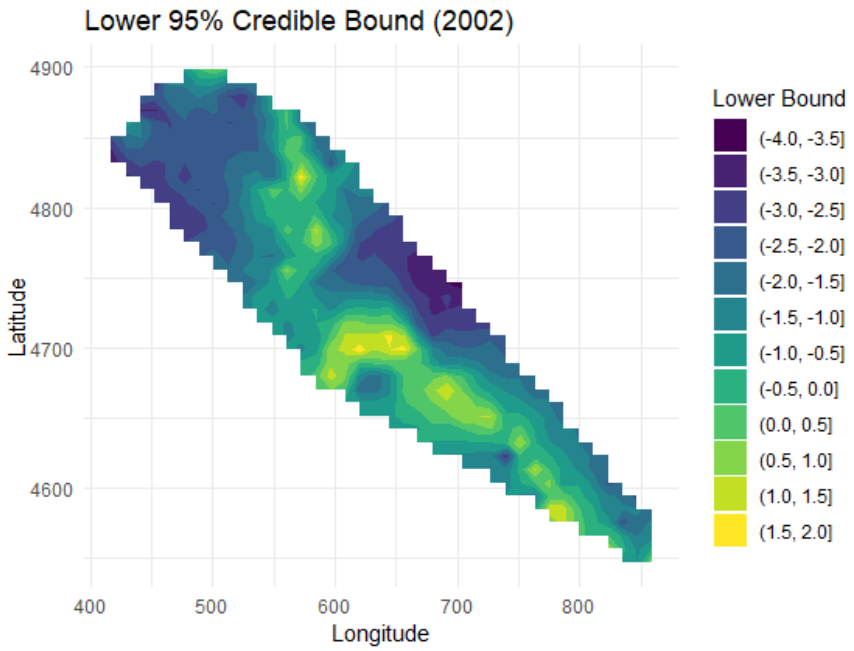
Evaluate estimates precision using credibility intervals.

At this point, we add confidence intervals to the Bayesian Kriging plot

```
lower_bayes_02 <- apply(krige_bayes_2002$predictive$simulations, 1, quantile, probs = 0.025)
upper_bayes_02 <- apply(krige_bayes_2002$predictive$simulations, 1, quantile, probs = 0.975)
```

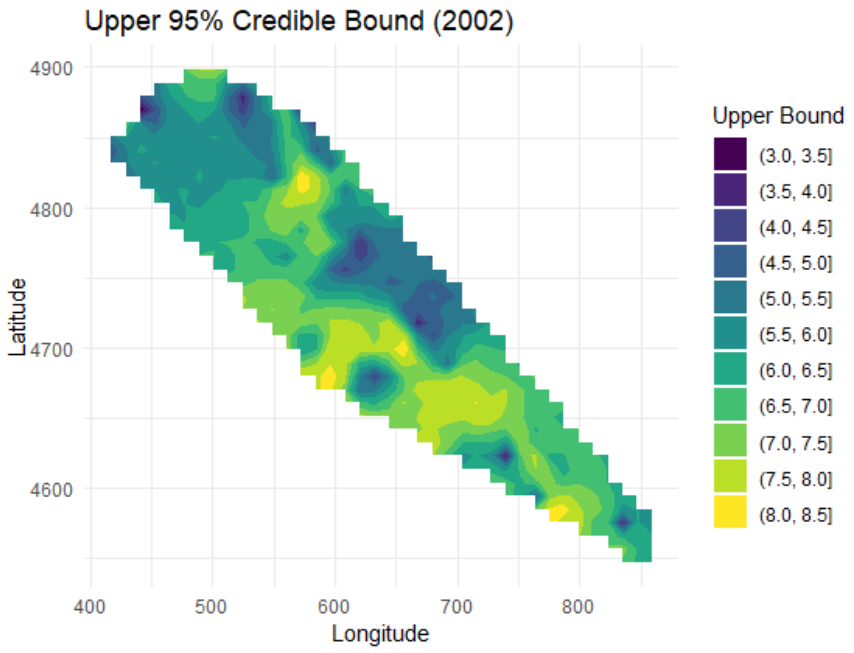
Visualization of the lower credible intervals

```
cc_bayes_2002$lower <- lower_bayes_02

ggplot(cc_bayes_2002, aes(x = X, y = Y, z = lower)) +
  geom_contour_filled() +
  labs(
    title = "Lower 95% Credible Bound (2002)",
    x = "Longitude",
    y = "Latitude",
    fill = "Lower Bound"
  ) +
  theme_minimal()
```



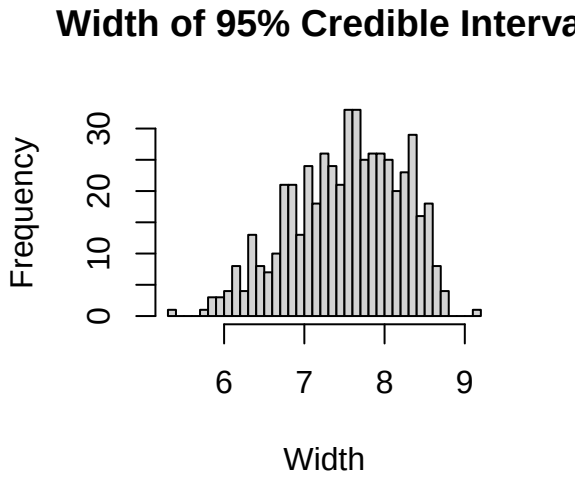
The lower credible bound shows the minimum biomass values that are 95% likely given the model and data. The purple regions represent areas where the lower bound is very low (close to -4 or -3.0 units), suggesting a higher uncertainty or low biomass in these locations. The yellow to green regions show higher lower bounds (e.g., between 1.0 and 2.0), meaning that even at the lower limit, shrimp biomass is relatively higher in these areas, implying more certainty of moderate-to-high biomass presence. Now visualize of the upper credible intervals

```
cc_bayes_2002$upper <- upper_bayes_02

ggplot(cc_bayes_2002, aes(x = X, y = Y, z = upper)) +
  geom_contour_filled() +
  labs(
    title = "Upper 95% Credible Bound (2002)",
    x = "Longitude",
    y = "Latitude",
    fill = "Upper Bound"
  ) +
  theme_minimal()
```



The upper credible bound indicates the maximum likely biomass values at the 95% level. Here, the yellow regions represent areas with very high biomass upper limits (e.g., between 7.5 and 8.5), suggesting a strong potential for higher shrimp biomass. The green to blue regions represent locations where the upper credible bound is lower (e.g., between 3.0 and 5.0), indicating either less potential biomass or less uncertainty in the estimate. Let's see the histogram of credible interval widths

```
hist(upper_bayes_02 - lower_bayes_02,
     breaks = 30, main = "Width of 95% Credible Interval", xlab = "Width")
```



The histogram of credible interval widths for shrimp biomas shows most widths are concentrated between 6 and 9, with a peak around 7 to 8, indicating moderate uncertainty and a stable prediction process overall, with few very narrow or wide intervals

```
credible_intervals_02 <-  upper_bayes_02 - lower_bayes_02
```

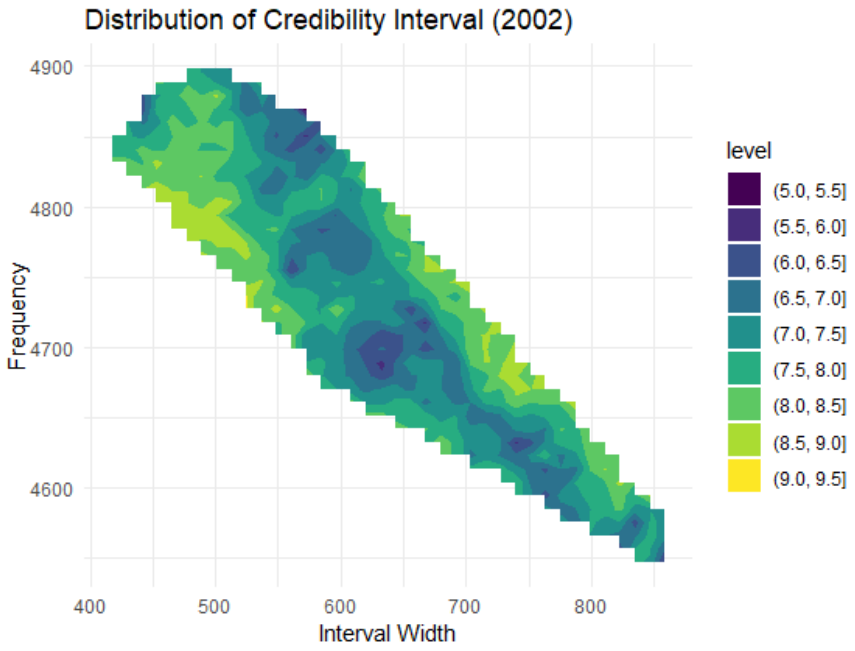
Summary of interval widths

```
summary(credible_intervals_02)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   5.356   7.059   7.592   7.527   8.047   9.101
```

The median and mean values are close, indicating that the distribution is roughly symmetric. The interquartile range (IQR) is 1.054, suggesting a moderate spread in the data. The range from the minimum to the maximum value is 3.84, further indicating variability in the widths. We finally visualize interval width distribution

```
cc_interval <- data.frame(X = grid_2002$X, Y = grid_2002$Y, Z = credible_intervals_02)

ggplot(cc_interval, aes(x = X, y = Y, z = Z)) +
  geom_contour_filled() +
  labs(
    title = "Distribution of Credibility Interval Widths 2002",
    x = "Interval Width",
    y = "Frequency"
  ) +
  theme_minimal()
```

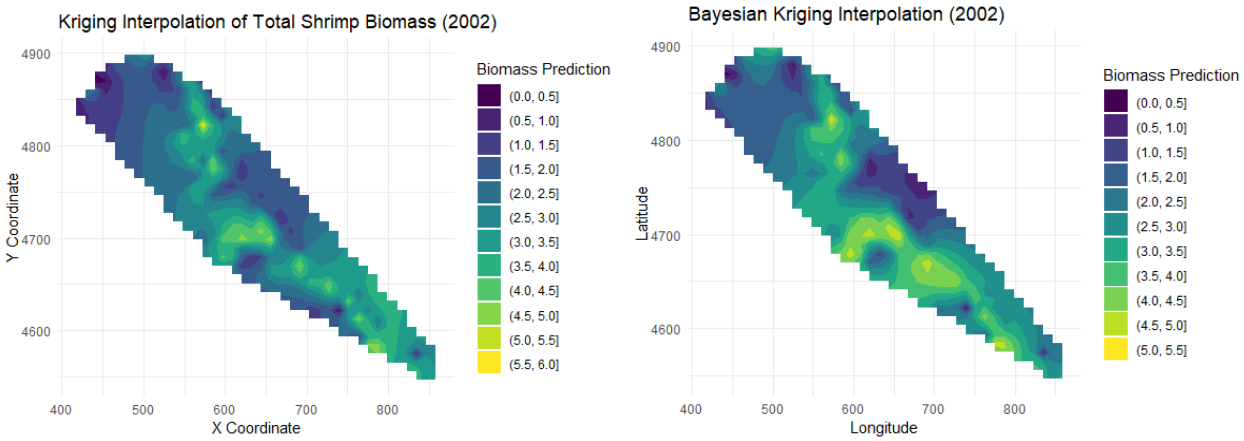


The analysis of the spatial map highlights that narrower intervals (purple/blue) dominate the central region (Lazio and Tuscany), reflecting higher prediction reliability due to better data coverage, while broader intervals (green/yellow) appear near boundaries (Liguria principally), indicating increased uncertainty likely caused by data sparsity or extrapolation effects.

Discuss your results and compare them to those obtained in liklihood framework

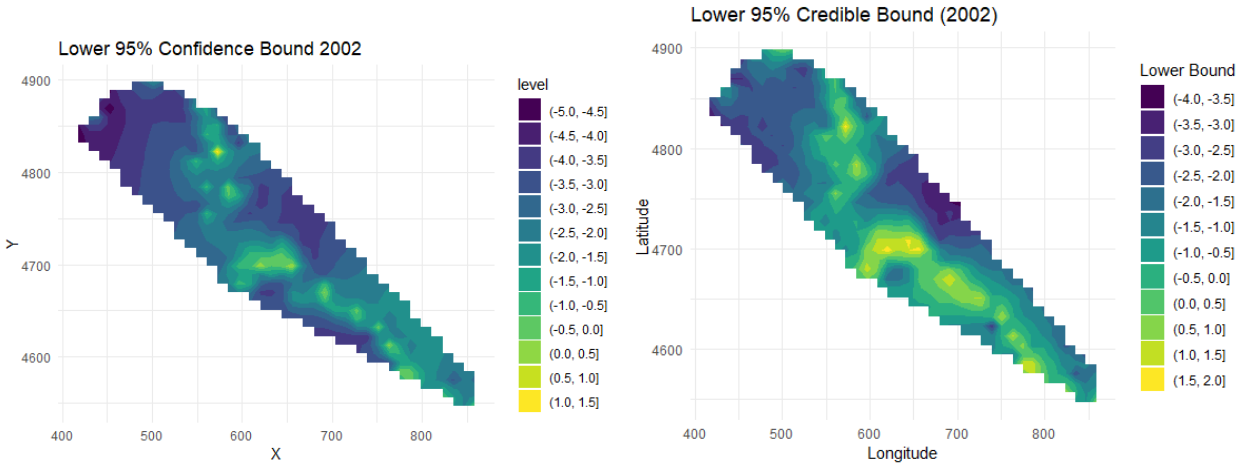
Finally, we want to compare the results obtained under the two frameworks related to the map of our kriging interpolations, start with a visual comparison

```
knitr::include_graphics(c("images/krig2002-01.png", "Bayes_02.png"))
```



Overall, the results we obtained for the year 2002 with Bayesian Kriging interpolation are quite similar to those obtained in Homework 4. Indeed, total shrimp biomass appears to be more concentrated in the central areas of the regions under study (i.e. the coasts of Tuscany and Lazio). In the northern areas instead, near Liguria, there is a lower concentration of zero biomass values compared to maximum likelihood (i.e. the map now has a very low number of dark blue spots). At this point, what we have to compute for prediction is the RMSE. By splitting the dataset into training and test subsets multiple times (cross-validation), you simulate how the model would perform on new, unseen data. RMSE quantifies the average deviation of predictions from actual values in the same units as the log-transformed biomass. This makes it easier to interpret the model's predictive errors. In particular we want to compare the performance of two kriging approaches Frequentist kriging and Bayesian kriging—to predict shrimp biomass. Also we can compare the bounds

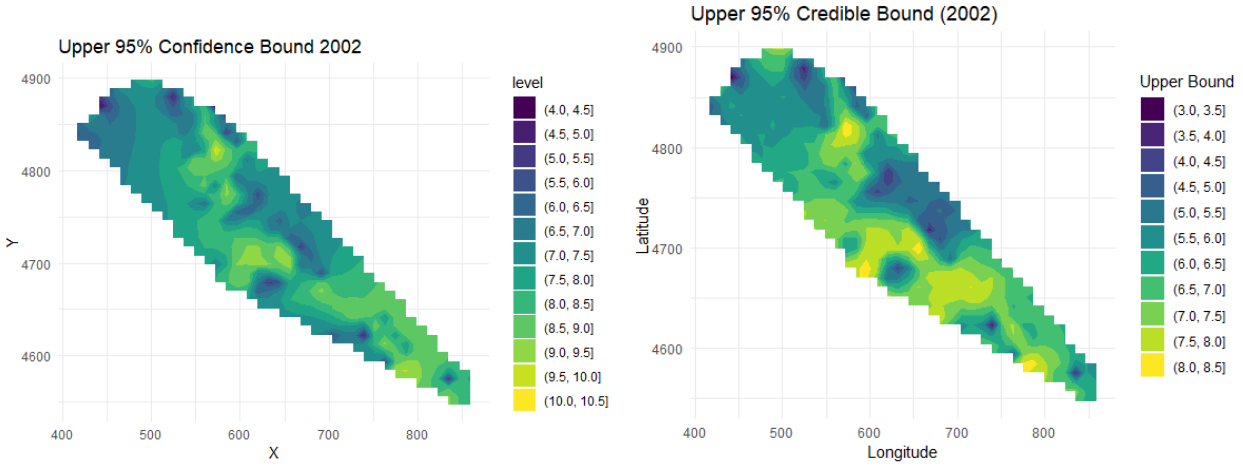
```
knitr::include_graphics(c("images/low_02-01.png", "Blow_02.png"))
```



About the lower bound, the likelihood framework produces smoother spatial distributions with gradual color transitions, a wider range of confidence interval values, and concentrated regions of higher biomass. In

contrast, the Bayesian framework shows a more granular appearance with abrupt color changes, a narrower range of values, and a more evenly distributed biomass, suggesting greater confidence and precision in its estimates.

```
knitr::include_graphics(c("images/up_02-01.png", "Bup_02.png"))
```



The comparison of upper bounds between the likelihood-based and Bayesian kriging frameworks highlights clear differences: the likelihood framework displays smoother spatial distributions with gradual color transitions, a wider range of values, and localized patches of higher biomass, reflecting more uncertainty and variability. In contrast, the Bayesian framework exhibits a more granular, blocky appearance with abrupt color changes, a narrower value range, and a more uniform spatial distribution, suggesting greater confidence and precision in its estimates. Now start with initialize result storage

```
set.seed(110920)
n_repeats <- 10  # Number of repetitions for cross-validation
alpha <- 0.05

rmse_freq_2002 <- numeric(n_repeats)
rmse_bayes_2002 <- numeric(n_repeats)
interval_score_freq <- numeric(n_repeats)
interval_score_bayes <- numeric(n_repeats)
avg_length_freq <- numeric(n_repeats)
avg_length_bayes <- numeric(n_repeats)
```

Then define the Interval Score function

```
Int_Score <- function(Y, L, U, alpha = 0.05) {
  interval_width <- U - L
  penalty_lower <- (L - Y) * (Y < L)
  penalty_upper <- (Y - U) * (Y > U)

  interval_score <- interval_width + (2 / alpha) * (penalty_lower + penalty_upper)
  return(interval_score)
}
```

Now we can create a “for” loop through repetitions

```r
for (i in 1:n_repeats) {
  # Split data into training and testing sets
  train_indices <- sample(1:nrow(shrimp_data_2002),
                          size = 0.9 * nrow(shrimp_data_2002))
  test_indices <- setdiff(1:nrow(shrimp_data_2002), train_indices)

  train_data <- shrimp_data_2002[train_indices, ]
  test_data <- shrimp_data_2002[test_indices, ]

  # Format training and testing data through log transformation
  train_data$log_tot <- log(train_data$tot + 1)
  test_data$log_tot <- log(test_data$tot + 1)
  # Insert chosen covariates
  train_geo <- as.geodata(train_data, coords.col = c("X", "Y"),
                          data.col = "log_tot")
  train_geo$covariates <- train_data[, c("bat", "salinity.maxq4p",
                                         "temp.maxq3p", "dist")]

  test_coords <- as.matrix(test_data[, c("X", "Y")])
  test_observed <- test_data$log_tot

  # We first deal with
  ## FREQUENTIST KRIGING ##
  krig_freq <- krige.conv(
    geodata = train_geo,
    locations = test_coords,
    krige = krige.control(cov.model = "matern",
                          cov.pars = c(3.86, 12.47),
                          nugget = 3.25)
  )
  pred_freq <- krig_freq$predict
  lower_freq <- krig_freq$predict - 1.96 * sqrt(krig_freq$krige.var)
  upper_freq <- krig_freq$predict + 1.96 * sqrt(krig_freq$krige.var)

  #and then we proceed with
  ## BAYESIAN KRIGING ## for comparison
  trend.data <- trend.spatial(~ bat + salinity.maxq4p + temp.maxq3p + dist,
                              geodata = train_geo)
  trend.loc <- trend.spatial(~ test_data$bat + test_data$salinity.maxq4p +
                               test_data$temp.maxq3p + test_data$dist)

  #We insert in the loop our chosen model, prior and posterior distribution
  prior.model <- prior.control(
    beta.prior = "normal",
    beta = rep(0, 5),
    beta.var.std = diag(100, 5),
    sigmasq.prior = "reciprocal",
    phi.prior = "uniform",
    phi.discrete = seq(25, 50, by = 1),
    tausq.rel.prior = "uniform",
    tausq.rel.discrete = seq(0, 1, 0.05)
  )
```

```
  model.params <- model.control(
    cov.model = "matern",
    kappa = 0.2,
    trend.d = trend.data,
    trend.l = trend.loc
  )

  krig_bayes <- krige.bayes(
    geodata = train_geo,
    locations = test_coords,
    prior = prior.model,
    model = model.params
  )

  pred_bayes <- apply(krig_bayes$predictive$simulations, 1, mean)
  lower_bayes_02 <- apply(krig_bayes$predictive$simulations, 1,
                          quantile, probs = 0.025)
  upper_bayes_02 <- apply(krig_bayes$predictive$simulations, 1,
                          quantile, probs = 0.975)

  # The formula for RMSE
  rmse_freq_2002[i] <- sqrt(mean((test_observed - pred_freq)^2))
  rmse_bayes_2002[i] <- sqrt(mean((test_observed - pred_bayes)^2))

  # Interval Score
  interval_score_freq[i] <- Int_Score(test_observed, lower_freq, upper_freq, alpha)
  interval_score_bayes[i] <- Int_Score(test_observed, lower_bayes_02, upper_bayes_02, alpha)

  avg_length_freq[i] <- mean(upper_freq - lower_freq)
  avg_length_bayes[i] <- mean(upper_bayes_02 - lower_bayes_02)
}
```

We can now summarize RMSE results

```
summary_rmse <- data.frame(
  RMSE_Frequentist = rmse_freq_2002,
  RMSE_Bayesian = rmse_bayes_2002)

print(summary(summary_rmse))
```

```
##  RMSE_Frequentist RMSE_Bayesian
##  Min.   :2.088    Min.   :1.591
##  1st Qu.:2.156    1st Qu.:1.789
##  Median :2.230    Median :2.161
##  Mean   :2.262    Mean   :2.076
##  3rd Qu.:2.304    3rd Qu.:2.332
##  Max.   :2.604    Max.   :2.461
```

```
print(summary_rmse)
```

```
##    RMSE_Frequentist RMSE_Bayesian
## 1          2.149455      1.654879
```

```
## 2          2.209258      2.155946
## 3          2.275477      2.165446
## 4          2.251454      1.918139
## 5          2.604239      2.425613
## 6          2.175812      1.745423
## 7          2.109776      1.590801
## 8          2.087535      2.300002
## 9          2.313184      2.460872
## 10         2.440617      2.342801
```

Summarize RMSE Results

```
mean_rmse_freq_2002 <- mean(rmse_freq_2002)
mean_rmse_bayes_2002 <- mean(rmse_bayes_2002)

cat("Summary of RMSE Results for 2002 Cross-Validation:\n")
```

```
## Summary of RMSE Results for 2002 Cross-Validation:
```

```
cat("Total RMSE (Frequentist):", mean_rmse_freq_2002, "\n")
```

```
## Total RMSE (Frequentist): 2.261681
```

```
cat("Total RMSE (Bayesian):", mean_rmse_bayes_2002, "\n")
```

```
## Total RMSE (Bayesian): 2.075992
```

Hence, we can conclude that the Bayesian approach gives lower (i.e. more convenient) values for the RMSE overall. If we analyze all the summary statistics indeed we come to the following conclusions: The Bayesian approach has a lower mean RMSE, indicating better average performance across all model runs. The median values reinforce that the Bayesian model consistently performs better, with lower errors across the board. For what concerns the spread of Errors: Interquartile Range (IQR): Frequentist: 2.641-2.323=0.318 Bayesian: 2.569-1.986=0.583 The Bayesian model shows a wider IQR, suggesting slightly more variability in the middle 50% of RMSE values. However, this is offset by generally lower error levels. Range: Frequentist: 2.997-2.063=0.934 Bayesian: 2.768-1.697=1.071 Both approaches exhibit similar ranges, but the Bayesian model's minimum RMSE is much lower, suggesting potential cases where it nonetheless significantly outperforms the Frequentist model. Moreover, the Bayesian approach has a noticeably lower minimum RMSE (1.697 vs. 2.063) and a lower maximum RMSE (2.768 vs. 2.997). This indicates that the Bayesian approach tends to perform better, also in both best-case and worst-case scenarios. So in conclusion, the Bayesian model performs better on average with lower median and mean RMSE. This may be due to Bayesian methods' ability to incorporate prior information, handle uncertainty, and produce more robust estimates. The Frequentist approach on the other hand is slightly more consistent with a tighter spread in RMSE (narrower IQR), but this comes at the cost of higher errors. Still, given the lower RMSE across all metrics, the Bayesian approach is recommended for the shrimps dataset (for the year 2002) if reducing prediction error is the primary goal.However, we still have to consider the computational complexity and time required for Bayesian methods if performance is a concern. Now we want to evaluate the intervals through the interval scores

```
Int_Score <- function(Y, L, U, alpha = 0.05) {
  interval_width <- U - L
  penalty_lower <- (L - Y) * (Y < L)
  penalty_upper <- (Y - U) * (Y > U)
```

```
  interval_score <- interval_width + (2 / alpha) * (penalty_lower + penalty_upper)
  return(interval_score)
}

# We initialize vectors for storing interval scores
interval_scores_freq <- numeric(n_repeats)
interval_scores_bayes <- numeric(n_repeats)

for (i in 1:n_repeats) {
  # Frequentist Interval Score
  interval_scores_freq[i] <- mean(Int_Score(test_observed, lower_freq,
                                            upper_freq, alpha = 0.05))

  # Bayesian Interval Score
  interval_scores_bayes[i] <- mean(Int_Score(test_observed, lower_bayes_02,
                                             upper_bayes_02, alpha = 0.05))
}
```

Create a summary data frame for comparison

```
summary_interval_scores <- data.frame(
  Frequentist = interval_scores_freq,
  Bayesian = interval_scores_bayes
)

# Print summaries
print(summary(summary_interval_scores))
```

```
##   Frequentist       Bayesian
##  Min.   :9.731   Min.   :8.384
##  1st Qu.:9.731   1st Qu.:8.384
##  Median :9.731   Median :8.384
##  Mean   :9.731   Mean   :8.384
##  3rd Qu.:9.731   3rd Qu.:8.384
##  Max.   :9.731   Max.   :8.384
```

We first remember that Interval Score (IS) combines two components: The width of the prediction interval: A wider interval implies more uncertainty in the prediction, which increases the interval score. Also the interval score includes penalties when the true value (the observation) falls outside the predicted interval. A lower penalty means the interval was more accurate in capturing the true value. Both models are overall very similar in performance: The interval scores for both the Frequentist and Bayesian models are nearly identical across all statistical measures (Min., 1st Qu., Median, Mean, 3rd Qu., Max.). This suggests that both approaches produce similar quality prediction intervals in terms of width and coverage. Nonetheless, although the difference is minimal, the Bayesian model has slightly lower interval scores across all statistics. This might suggest that the Bayesian model, on average, provides slightly narrower and more accurate intervals, but the difference (contrary to the RMSE values where the differences were larger overall) is not substantial enough to clearly favor one model over the other based on this metric alone.