Appendix S2 - Analysing Co-Occurrences Using Conditional Random Fields

Nicholas Clark, James Kerry, Ceridwen Fraser

Here we present the code used to run Conditional Random Fields models using functions in the `MRFcov` package (Clark et al., 2018). First, load the pre-processed datasets that were created in **Appendix S1** and saved into the `Processed data` directory

```r
load("MedFishes/Processed data/MRF.data.rda")
```

Next, we load the `MRFcov` library and use the `cv_MRF_diag_rep_spatial` function to compare fits of spatial and non-spatial CRF models. This comparison allows us to determine whether there is support in the data for fitting a slightly more complex model that does account for spatial autocorrelation (smoothed spatial regression splines), as opposed to a standard CRF that does not include spatial terms. Because these functions are being run on a high-performance computing cluster, we are able to split the job across 24 processing cores to speed up computations.

```r
if (!require(MRFcov)) {
    devtools::install_github("nicholasjclark/MRFcov")
}

library(MRFcov)
n.species <- length(mrf.species)
comparison <- cv_MRF_diag_rep_spatial(data = Medfish.mrf.dat,
    n_nodes = n.species, family = "binomial",
    coords = coords, n_cores = 24, compare_null = T,
    plot = F, n_fold_runs = 100)
```

Following model fitting and calculation of cross-validated prediction metrics, we can use the returned `dataframe` to calculate the median proportion of correct predictions for each model

```r
Spatial.true.prop <- quantile(comparison$mean_tot_pred[which(comparison$model ==
    "Spatial MRF")], 0.5)
Nonspatial.true.prop <- quantile(comparison$mean_tot_pred[which(comparison$model ==
    "Non-spatial MRF")], 0.5)
```

We store a character vector to descrive which model has the better overall fit

```r
if (which.max(c(Spatial.true.prop, Nonspatial.true.prop)) ==
    1) {
    best.mod <- "Spatial"
} else {
    best.mod <- "Nonspatial"
}
```

Which model fit the best?

```r
best.mod
```

```
## [1] "Spatial"
```

The spatial CRF fits better based on the median of the proportion of true values metric, but it is also worth exploring fit metrics in more detail using 95% quantiles

```r
quantile(comparison$mean_tot_pred[which(comparison$model ==
    "Spatial MRF")], c(0.025, 0.5, 0.975))
```

```
##      2.5%       50%     97.5%
## 0.9257805 0.9278522 0.9298942
```

```r
quantile(comparison$mean_tot_pred[which(comparison$model ==
    "Non-spatial MRF")], c(0.025, 0.5, 0.975))
```

```
##      2.5%       50%     97.5%
## 0.8819455 0.8844972 0.8874685
```

Next, explore 95% quantiles of positive predictive values (accuracy of positive predictions)

```r
quantile(comparison$mean_pos_pred[which(comparison$model ==
    "Spatial MRF")], c(0.025, 0.5, 0.975))
```

```
##      2.5%       50%     97.5%
## 0.9661832 0.9675461 0.9690489
```

```r
quantile(comparison$mean_pos_pred[which(comparison$model ==
    "Non-spatial MRF")], c(0.025, 0.5, 0.975))
```

```
##      2.5%       50%     97.5%
## 0.8633087 0.8667215 0.8701810
```

And model sensitivity (proportion of positives correctly predicted)

```r
quantile(comparison$mean_sensitivity[which(comparison$model ==
    "Spatial MRF")], c(0.025, 0.5, 0.975))
```

```
##      2.5%       50%     97.5%
## 0.8440024 0.8467680 0.8493789
```

```r
quantile(comparison$mean_sensitivity[which(comparison$model ==
    "Non-spatial MRF")], c(0.025, 0.5, 0.975))
```

```
##      2.5%       50%     97.5%
## 0.8358383 0.8382236 0.8407906
```

As well as specificity (proportion of negatives correctly predicted)

```r
quantile(comparison$mean_specificity[which(comparison$model ==
    "Spatial MRF")], c(0.025, 0.5, 0.975))
```

```
##      2.5%       50%     97.5%
## 0.9804172 0.9812904 0.9822786
```

```r
quantile(comparison$mean_specificity[which(comparison$model ==
    "Non-spatial MRF")], c(0.025, 0.5, 0.975))
```

```
##      2.5%       50%     97.5%
## 0.9121039 0.9150866 0.9180983
```

Across the board, the spatial model is clearly providing a better fit (correctly predicting an impressive 92% of total observations in the dataset). We can now fit the full model and generate predictions using the `predict_MRF` function. *Note, this step can be time-consuming*

```
if (best.mod == "Nonspatial") {
    CRF <- MRFcov(data = Medfish.mrf.dat,
        family = "binomial", n_nodes = n.species,
        n_cores = 24)
    preds <- predict_MRF(data = Medfish.mrf.dat,
        MRF_mod = MRF, n_cores = 24)

} else {
    CRF <- MRFcov_spatial(data = Medfish.mrf.dat,
        family = "binomial", n_nodes = n.species,
        n_cores = 24, coords = coords)

    preds <- predict_MRF(data = MRF.spatial$mrf_data,
        MRF_mod = MRF.spatial, prep_covariates = F,
        n_cores = 24)
}
```

Save the model results into the `Results` directory

```
save(CRF, preds, best.mod, comparison, file = "Results/MRF.results.rda")
```

References

Clark, N.J., Wells, K. & Lindberg, O. (2018) MRFcov: Markov Random Fields with additional covariates. R package version 1.0. GitHub.