

Workshop Homework: OOP, Shiny & Rcpp

Homework Assignment

Please complete the following tasks within one week and submit as a GitHub repository named `knn_workshop_hw`.

1. Extend the k-NN S3 Class (30 pts)

- **Inverse-Distance Weights:** Modify your `knn_s3` implementation to support a `weight = c("uniform", "inverse")` argument. When `inverse`, use weights proportional to $1/\text{distance}$, normalized to sum to 1.
 - **Unit Tests:** Write `testthat` tests verifying both weighting schemes, and that uniform results match when distances are equal.
 - **Documentation:** Add roxygen comments for the new argument and regenerate help files with `devtools::document()`.
-

2. Shiny App Enhancement (30 pts)

- **Dynamic Predictors:** In the k-NN app, add a multi-select input to choose predictor variables at runtime.
 - **Nearest-Neighbor Table:** Display the indices and distances of the k nearest neighbors for the first test observation in a table.
 - **Download Button:** Enable downloading the test-set predictions as a CSV file.
 - **Input Validation:** Disable or warn if no predictors are selected.
-

3. Rcpp Performance Study (30 pts)

- **Benchmark Script:** Create `benchmark_knn_dims.R` to measure runtimes for pure-R vs Rcpp across combinations of sample size $n \in \{500, 1000, 2000\}$ and feature count $p \in \{3, 5, 10\}$. Plot a heatmap of speed-up ratios.
- **C++ Optimization:** Modify `knn_pred.cpp` to use `std::partial_sort` instead of `std::nth_element`, re-benchmark, and report any differences.
- **Summary Report:** Produce a one-page PDF summarizing your benchmarks, code changes, and recommendations.

Deliverables

Your GitHub repository `knn_workshop_hw` should include:

```
knn_workshop_hw/  
  R/                # R scripts and tests  
  app/              # Shiny app files  
  src/              # C++ source for Rcpp  
  benchmark_knn_dims.R  
  report.pdf        # summary of benchmarks and findings  
  README.md         # overview and instructions
```

Include a **README** with setup instructions and a sample run of each component.

Good luck, and feel free to reach out with questions!