MultiGS-R (v1.0)

Java Pipeline for Genomic Selection of Multiple Single Traits Using R-Based Models and Diverse Marker Types

MultiGS-R is a powerful, flexible, and user-friendly Java-based pipeline for performing genomic selection (GS) analysis. It seamlessly integrates a wide range of popular R packages implementing both classical statistical and modern machine learning models, providing a unified platform for cross-validation and across-population prediction in plant and animal breeding programs.

The pipeline supports multiple genomic marker types (SNPs, haplotypes, and principal components) and a comprehensive suite of GS modeling algorithms, making it an all-in-one solution for breeders and researchers.

A detailed tutorial is available in the file MultiGS-R_v1.0_tutorial.pdf.

Table of Contents

Introduction

Genomic Selection accelerates genetic improvement by predicting the genetic-estimated breeding values (GEBVs) of individuals based on their genomic markers. MultiGS-R automates the complex workflow of GS, which includes data preprocessing, quality control, imputation, model training, and validation. By leveraging the robust statistical capabilities of R within a managed Java pipeline, MultiGS-R ensures reproducibility, scalability, and ease of use for both small-scale studies and large breeding populations.

Key Features

- **Flexible Analysis Modes:** Supports both **cross-validation** (for model evaluation) and independent **across-population prediction** (using a training set to predict a new test set).
- **Multiple Marker Views:**
    - **SNP:** Direct use of Single Nucleotide Polymorphisms (SNPs).
    - **HAP:** Conversion of SNPs into haplotype blocks using RTM-GWAS SNP-LD for potentially capturing epistatic effects.
    - **PCA:** Use of Principal Components as markers to reduce dimensionality and address multicollinearity.
- **Comprehensive Data Preprocessing:** Includes sample alignment, genotype harmonization between training and test lines, and missing data imputation.
- **Diverse GS Modeling Methods:** Integrates several state-of-the-art models via R packages:
    - **Linear Models:** Ridge-Regression BLUP (`RR-BLUP`) via `rrBLUP` and Genomic Best Linear Unbiased Predictio (`GBLUP`) via BGLR.
    - **Kernel Methods:** Reproducing Kernel Hilbert Spaces (`RKHS`).
    - System R**Bayesian Approaches:** BL (Bayesian LASSO), BRR (Bayesian Ridge Regression), BayesA, BayesB, BayesC via `BGLR`.
    - **Machine Learning:** Random Forest for Regression (`RFR`) and Classification (`RFC`), Support Vector Regression (`SVR`) and Classification (`SVC`).

Requirements & Installation

Prerequisites

1. **Java Runtime Environment (JRE):** Version 21 or higher must be installed. You can check by running `java -version` in your terminal.
2. **R:** Version 3.5 or higher must be installed and accessible from the command line. Check with `R --version`.
3. **Rscript:** This executable (included with R) must be in your system's PATH.

Installing R Libraries

Before running MultiGS-R, you must install the required R packages. Start an R session and run the following commands:

r

```
# Install required packages from CRAN

install.packages(c("rrBLUP", "BGLR", "randomForest", "e1071", "ade4", "sommer", "ggplot2"))

An additional G2P package needs to be installed through source file. Please download it from GitHub and follow installation instruction:

https://github.com/cma2015/G2P
```

Installing `rtm-gwas-snpldb` tool

The latest executable can be downloaded from:

https://github.com/njau-sri/rtm-gwas

Obtaining MultiGS-R

Download the latest release of the MultiGS-R repository from https://github.com/AAFC-ORDC-Crop-Bioinfomatics/MultiGS-R

Quick Start

1. **Prepare your data:** Have your VCF marker files and phenotypic data files ready.
2. **Create a configuration file:** Copy the sample below and modify the paths to match your system and data.
3. **Run the pipeline:**

bash

```
java -jar MultiGS-R_1.0.jar /path/to/your/config.ini
```

Configuration File

The pipeline is controlled by a single configuration file using an INI-style format.

Sample Configuration

ini

```
# This is a configuration file for MultiGS-R pipeline.


[Tools]

# Haplotype block identification tool (included with MultiGS-R or can be
downloaded)

rtm_gwas_snpldb_path = /home/user/MultiGS-R/rtm_gwas/rtm-gwas-snpldb


# R path

RScriptPath = /usr/bin/Rscript


[General]

# variance explained for selection of number of principal components

pca_variance_explained = 0.95


# result output folder

result_folder = sample_results_CV


# Number of threads for parallel computation

threads = 7


# number of replicates in CROSS-VALIDATION mode

Replicates = 2


[GS_Mode]

# Mode: CROSS-VALIDATION | PREDICTION
```

```
mode = CROSS-VALIDATION

[Feature_view]
# Three marker types: raw SNPs (SNP), haplotypes (HAP) and principal components (PC)
marker_type = PC

[Data]
# (training) marker file (for cross_validation or Prediction)
marker_file=/path/to/training_markers.vcf

# test marker file (required for PREDICTION mode, optional for CROSS-VALIDATION)
test_marker_file=/path/to/test_markers.vcf

# training phenotypic data file for both modes
training_pheno_file=/path/to/training_pheno.txt

# test phenotypic data file (optional, for PREDICTION mode only)
test_pheno_file=/path/to/test_pheno.txt

[Models]
# Choose GS modeling methods: True | False
# Parametric/linear models
RR-BLUP = True
GBLUP = True
BRR = True
BL = True
BayesA = True
BayesB = True
BayesC = True

# Non-parametric machine learning methods
RFR = True
SVR = True
```

```
RKHS = True


# Classifiers
RFC = True
SVC = True


[Hyperparameters]
# Model parameters for Bayesian methods
nIter = 12000
burnIn = 2000
```

Parameter Details

Section	Parameter	Description	Values
Tools	`rtm_gwas_snpldb_path`	Path to haplotype block identification tool	File path
	`RScriptPath`	Path to RScript executable	File path
General	`pca_variance_explained`	Variance cutoff for PCA component selection	0.0-1.0 (e.g., 0.95)
	`result_folder`	Output directory for results	Directory path
	`threads`	Number of CPU threads for parallel processing	Integer
	`Replicates`	Number of CV replicates	Integer
GS_Mode	`mode`	Analysis mode	`CROSS-VALIDATION` or `PREDICTION`
Feature_view	`marker_type`	Type of markers to use	`SNP`, `HAP`, or `PC`
Data	`marker_file`	Training population VCF file	File path
	`test_marker_file`	Test population VCF file (Prediction mode)	File path
	`training_pheno_file`	Training phenotype data	File path
	`test_pheno_file`	Test phenotype data	File path

Section	Parameter	Description	Values
		(optional)	
Models	Various	Enable/disable specific GS models	`True` or `False`
Hyperparameters	`nIter`	MCMC iterations for Bayesian models	Integer (e.g., 12000)
	`burnIn`	MCMC burn-in period	Integer (e.g., 2000)

Input Files

Genotypic Data (Markers)

- **Format:** standard VCF (Variant Call Format) with header - can be compressed (.vcf.gz) or uncompressed
- **Requirements:**
    - For **Cross-Validation:** One VCF file for the training population
    - For **Prediction:** Two VCF files (training and test)

Phenotypic Data

- **Format:** Tab-delimited or CVS text file **with a header row**
- **Structure:**
    - First column: Individual/Sample IDs
    - Subsequent columns: Phenotypic values for different traits

Example `training_pheno.txt`**:**

text

```
SampleID        Yield   Height
sample_1        5.6     112
sample_2        4.8     105
sample_3        NA      108
```

Missing values should be coded as `NA`*. The pipeline will handle them automatically.*

Usage

1. **Prepare your configuration file** following the template above
2. **Run the pipeline:**

bash

```
java -jar MultiGS-R_1.0.jar /path/to/your/config.ini
```

3. **Monitor progress:** The pipeline will display progress in the console and write detailed logs to the output directory

For large datasets, you may need to increase memory allocation:

bash

```
java -Xmx8g -jar MultiGS-R_1.0.jar config.ini
```

Output

The pipeline generates a well-organized directory structure:

text

```
result_folder/
├── gs_<timestamp>.log              # Detailed log file
├── all_CV_results.txt              # Detailed CV results (CV mode)
├── CV_summary_statistics.csv       # Summary statistics (CV mode)
├── prediction_detailed_results.txt # Model results (Prediction mode)
│
├── trait_predictions/              # Predicted values for test set
│   └── <Trait>_<Model>_prediction_data.txt
├── plots/                          # Diagnostic plots
│   ├── MDS_plot.png               # Population structure
│   └── ...                        # Other visualizations
├── intermediate_data/              # Processed intermediate files
└── pheno_data/                     # Preprocessed phenotypic data
```

Troubleshooting

- **"RScript not found":** Verify the `RScriptPath` in your configuration file is correct
- **Missing R packages:** Check the log file for package errors and install missing packages in R
- **Memory errors:** Use `-Xmx` parameter to increase Java heap space (e.g., `-Xmx8g` for 8GB)
- **VCF file errors:** Ensure your VCF files are properly formatted and have necessary header

Citation

If you use MultiGS-R in your research, please cite:

You FM, Zheng C, Zagariah Daniel JJ, Li P, Jackle K, House M, Tar'an T, Cloutier S. Genomic selection for seed yield prediction achieved through versatile pipelines for breeding efficiency in Flax. (In preparation).

License