

# MantelCorr Package for Bioconductor

Brian Steinmeyer, MS, and William Shannon, PhD  
Department of Internal Medicine  
Division of General Medical Sciences  
Washington University in St. Louis  
School of Medicine  
email: [steinmeb@ilya.wustl.edu](mailto:steinmeb@ilya.wustl.edu), [wshannon@wustl.edu](mailto:wshannon@wustl.edu)

January 22, 2026

## Contents

<b>1</b>	<b>Description</b>	<b>1</b>
<b>2</b>	<b>Example R session with the Golub training data</b>	<b>2</b>
<b>3</b>	<b>Reminder</b>	<b>2</b>
<b>4</b>	<b>GetClusters() function</b>	<b>2</b>
<b>5</b>	<b>DistMatrices() function</b>	<b>2</b>
<b>6</b>	<b>MantelCorrs() function</b>	<b>3</b>
<b>7</b>	<b>PermutationTest() function</b>	<b>3</b>
<b>8</b>	<b>ClusterList() function</b>	<b>3</b>
<b>9</b>	<b>ClusterGeneList() function</b>	<b>3</b>

## 1 Description

The **MantelCorr** package is based on the methodology developed in Shannon et al. [1], for which six functions are used to locate and identify important gene clusters from standard microarray expression data with **p** genes (*rows*) and **n** samples (*columns*). Mantel statistics have been applied with success to correlate gene expression levels with clinical covariates [3]. We also include a real microarray dataset with the package to help illustrate its functionality. Specifically, the package makes use of the **k-means()** function in R (*with arbitrary k, say  $k \in [5, \frac{p}{2}]$* ) to essentially over-partition the gene space into **k** non-overlapping clusters. Next, two types of dissimilarity matrices are computed, one based on the original data **Dfull**, and one for each resultant cluster, **Dsubset(k)**.

Mantel [2] cluster correlations are then found by correlating each **Dsubset(k)** with **Dfull**, resulting in **k** Mantel correlations. In order to destroy the distance dependent nature of **Dfull** and to obtain an empirical null distribution of distance independence, a permutation test is done, where the

number of permutations and  $\alpha$  significance level parameters can be chosen by the user. Specifically, the significance level provides the criterion value (*p-value*) at which a given cluster is considered significant or non-significant. Both significant and non-significant cluster lists can be viewed with the `ClusterList` function. In addition, a summary list of genes within these clusters can also be seen with the `ClusterGeneList` function.

We next introduce a simple application of the `MantelCorr` package with gene-expression training data taken from the Golub et al. [4] leukemia study.

## 2 Example R session with the Golub training data

The Golub training data consists of gene-expression values measured for 38 samples from Affymetrix Hgu6800 chips on 7,129 genes. There are 27 acute lymphoblastic leukemia (ALL) and 11 acute myeloid leukemia (AML) samples. To load the `MantelCorr` package, simply type `library(MantelCorr)`. The data can be loaded by typing `data(GolubTrain)` and a description provided with `?GolubTrain`.

```
> library(MantelCorr)
> data(GolubTrain)
> dim(GolubTrain)
[1] 7129 38
> data <- GolubTrain
```

## 3 Reminder

Help on any of the following `MantelCorr` package functions can be viewed by `?FunctionName`, which provides a complete description and overview of the function's purpose and syntax. In addition, all input 'data' values are **assumed** to be interval-scale (e.g., numeric data), with gene and sample labels assigned from the `dimnames()` function.

## 4 GetClusters() function

The `GetClusters()` function over-partitions the gene-space as described in the package description. We select `k = 500` clusters and store the result in an object called "kmeans.result".

```
> kmeans.result <- GetClusters(data, 500, 100)
```

## 5 DistMatrices() function

A function used to compute distance matrices `Dfull` and `Dsubset(k)` from the `k` non-overlapping clusters stored in "kmeans.result". The result is assigned to "DistMatrices.result".

```
> DistMatrices.result <- DistMatrices(data, kmeans.result$clusters)
```

## 6 MantelCorrs() function

The `MantelCorrs()` function uses `Dfull` and `Dsubset(k)` to compute a Mantel correlation for each `k`th cluster by correlating these two dissimilarity matrices. The result is saved in "MantelCorrs.result".

```
> MantelCorrs.result <- MantelCorrs(DistMatrices.result$Dfull, DistMatrices.result$Dsubsets)
```

## 7 PermutationTest() function

`PermutationTest()` permutes `Dfull` to obtain an empirical null distribution for which cluster significance is determined. We have selected 100 permutations in order to conserve CPU time, and chosen an  $\alpha$ -value of 0.05 for the 38 Golub leukemia samples. The result is stored in an object called "permuted.pval". NOTE: we recommend using at least 1000 permutations for a thorough analysis.

```
> permuted.pval <- PermutationTest(DistMatrices.result$Dfull, DistMatrices.result$Dsubsets, 100,
```

## 8 ClusterList() function

A function used to generate a complete list of both significant and non-significant clusters found by the permutation test and associated level of significance. Cluster size and correlation are provided with each type of cluster. We assign the result to the R object "ClusterLists" as follows:

```
> ClusterLists <- ClusterList(permuted.pval, kmeans.result$cluster.sizes, MantelCorrs.result)
```

## 9 ClusterGeneList() function

A final function that uses information from the "ClusterList" function, coupled with the `dimnames` function to generate a composite list of the genes found in both cluster types (significant and non-significant). We store the result in R object "ClusterGenes".

```
> ClusterGenes <- ClusterGeneList(kmeans.result$clusters, ClusterLists$SignificantClusters, data
```

## References

- [1] Shannon, Steinmeyer, Li, Culverhouse, Grefenstette, Thompson. (2005) *Variable Selection in Cluster Analysis Using k-means and Mantel Correlation*. Computing Science and Statistics (To Appear).
- [2] Mantel, N. *The Detection of Disease and a Generalized Regression Approach*. Cancer Research, 27, 209-220, 1967.
- [3] Shannon W, Watson M, Perry A, Rich K. *Mantel statistics to correlate gene expression levels from microarrays with clinical covariates*. Genetic Epidemiology 2002; 23:87-96.

- [4] Golub, T. et al. *Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring*. Science, 531-537, 1999.