

How to use the MiPP Package

Mat Soukup, HyungJun Cho, and Jae K. Lee

May 18, 2026

Contents

| | | |
|----------|--|----------|
| 1 | Introduction | 1 |
| 2 | Misclassification-Penalized Posteriors (MiPP) | 1 |
| 3 | Examples | 2 |
| 3.1 | Acute Leukemia Data | 2 |
| 3.2 | Colon Cancer Data | 4 |
| 3.3 | Sequential selection | 7 |

1 Introduction

The *MiPP* package is designed to sequentially add genes to a classification gene model based upon the Misclassification-Penalized Posteriors (MiPP) as discussed in Section 2. The construction of the model is based upon a training data set and the estimated actual performance of the model is based upon an independent data set. When no clear distinction between the training and independent data sets exists, the cross-validation technique is used to estimate actual performance. For the detailed algorithms, see Soukup, Cho, and Lee (2005) and Soukup and Lee (2004). The *MiPP* package employs libraries *MASS* for LDA/QDA (linear/quadratic discriminant analysis) and *e1071* for SVM (support vector machine). Users should install the *e1071* package from the main web page of R (<http://www.r-project.org/>).

2 Misclassification-Penalized Posteriors (MiPP)

In the above section, estimated actual performance is mentioned a number of times. Classically, the accuracy of a classification model is done by reporting its estimated actual error rate. However, error rate fails to take into account how likely a particular sample belongs to a given class and dichotomizes the data into yes the sample was

correctly classified or no the sample was NOT correctly classified. Although error rate, plays a key role in how well a classification model performs, it fails to take into account all the information that is available from a classification rule.

The Misclassification-Penalized Posteriors (MiPP) takes into account how likely a sample belongs to a given class by using a posterior probability of correct classification. MiPP also adjusts its definition any time a sample is misclassified by subtracting a 1 from the posterior probability of correct classification resulting in a negative value of MiPP. If we define the posterior probability of correct classification using genes \mathbf{x} as $\hat{f}(\mathbf{x})$, MiPP can be calculated as

$$\psi_p = \sum_{correct} \hat{f}(\mathbf{x}) + \sum_{wrong} (\hat{f}(\mathbf{x}) - 1). \quad (1)$$

Here, *correct* refers to the subset of samples that are correctly classified and *wrong* refers to the subset of samples that are misclassified. By introducing a random variable that takes into account whether a sample is misclassified or not MiPP can be shown to be the sum of posterior probabilities of correct classification minus the number of misclassified samples. As a result, MiPP increases whenever the sum of posterior probabilities of correction classification increase, the number of misclassified samples decreases, or both of these occur.

We standardize the MiPP score divided by the number of samples in each data set, denoted as sMiPP. Thus, the range of sMiPP is from -1 to 1. Note that as accuracy increases, sMiPP converges to 1.

Some basic properties of MiPP are that the maximum value it can take is equal to the sample size (or $sMiPP = 1$), and on the flip side, the minimum value is equal to the negation of the sample size (or $sMiPP = -1$). Under a pure random model, the expected value of MiPP is equal to zero (or $sMiPP = 0$). The variance is derived and is available from the first author for the two class case, however an explicit value for more than two classes can not be derived analytically. Thus, a bootstrapped estimate is the preferred method of estimating the variance.

3 Examples

3.1 Acute Leukemia Data

This data set has been frequently used for testing various methods in classification and prediction of cancer sub-types. Two distinct subsets of array data for AML and ALL leukemia patients are available: a training set of 27 ALL and 11 AML samples and a test set of 20 ALL and 14 AML samples. The independent set was from adult bone marrow samples, whereas the independent set was from 24 bone marrow samples, 10 from peripheral blood samples, and 4 of the AML samples from adults. Gene expression levels contain probes for 6817 human genes from AffymetrixTM oligonucleotide microarrays. Note that a subset of genes (713 probe sets) was stored into the *MiPP* package.

To run *MiPP*, the data can be prepared as follows.

```
data(leukemia)

#IQR normalization
leukemia <- cbind(leuk1, leuk2)
leukemia <- mipp.preproc(leukemia, data.type="MAS4")

#Train set
x.train <- leukemia[,1:38]
y.train <- factor(c(rep("ALL",27),rep("AML",11)))
#Test set
x.test <- leukemia[,39:72]
y.test <- factor(c(rep("ALL",20),rep("AML",14)))
```

Since two distinct data sets exist, the model is constructed on the training data and evaluated on the test data set as follows.

```
out <- mipp(x=x.train, y=y.train, x.test=x.test, y.test=y.test,
            n.fold=5, percent.cut=0.05, rule="lda")
```

This sequentially selects genes one gene at a time with the LDA rule (*rule="lda"*) and 5-fold cross-validation (*n.fold=5*) on the training set. To reduce computing time, it pre-selects the most plausible 5% out of 713 genes by the two-sample t-test (*percent.cut=0.05*), and then performs gene selection. To utilize all genes without pre-selection, set the argument *percent.cut=1*. The above command generates the following output.

```
out$model
```

| | Order | Gene | Tr.ER | Tr.MiPP | Tr.sMiPP | Te.ER | Te.MiPP | Te.sMiPP | Select |
|---|-------|------|--------|---------|----------|--------|---------|----------|--------|
| 1 | 1 | 571 | 0.0526 | 30.86 | 0.8122 | 0.1176 | 23.92 | 0.7035 | |
| 2 | 2 | 436 | 0.0000 | 36.89 | 0.9707 | 0.0294 | 30.41 | 0.8945 | |
| 3 | 3 | 366 | 0.0000 | 37.95 | 0.9988 | 0.0294 | 31.35 | 0.9222 | |
| 4 | 4 | 457 | 0.0000 | 38.00 | 0.9999 | 0.0294 | 32.14 | 0.9453 | |
| 5 | 5 | 413 | 0.0000 | 38.00 | 1.0000 | 0.0294 | 32.18 | 0.9464 | |
| 6 | 6 | 635 | 0.0000 | 38.00 | 1.0000 | 0.0000 | 33.75 | 0.9927 | ** |
| 7 | 7 | 648 | 0.0000 | 38.00 | 1.0000 | 0.0000 | 33.57 | 0.9874 | |

The gene models are evaluated by both train (denoted by **Tr**) and test (denoted by **Te**) sets; however, we select the final model based on the test set independent of the train

set used for gene selection. The gene model with the maximum sMiPP is indicated by one star (*) and the parsimonious model (indicated by **) contains the fewest number of genes with sMiPP greater than or equal to (max sMiPP - 0.01). In this example, the maximum and parsimonious models (indicated by **) are the same. Thus, the final model with sMiPP 0.993 contains genes 571, 436, 366, 457, 413, and 635. Note that genes listed in the output correspond to the column number of the matrices.

3.2 Colon Cancer Data

The colon cancer data set consists of the 2000 genes with the highest minimal intensity across the 62 tissue samples out of the original 6,500+ genes. The data set is filtered using the procedures described at the author's web site. The 62 samples consist of 40 colon tumor tissue samples and 22 normal colon tissue samples (Alon *et al.*, 1999). Li *et al.* (2001) identified 5 samples (N34, N36, T30, T33, and T36) which were likely to have been contaminated. As a result, these five samples are excluded from any future analysis; our error rate would be higher if they were included.

Since we are working with a small data set (57 samples), we will be implementing cross-validation techniques. With the lack of a 'true' independent test set, we randomly create a training data set with 38 samples (25 tumor and 13 normal) and an independent data set with 19 samples (12 tumor and 7 normal). Since this is a random creation of the data set, it would be of interest to see what model is selected based upon a different random split of the data. Note that the choice of the sizes of the training and independent test set is somewhat arbitrary, but consistent results were found using a training and test set of sizes 29 (19 tumor and 10 normal) and 28 (18 tumor and 10 normal), respectively. The colon data set of the *MiPP* package contains only 200 genes as an example. For the colon data with no independent test set, *MiPP* can be run as follows.

```
data(colon)
x <- mipp.preproc(colon)
y <- factor(c("T", "N", "T", "N", "T", "N", "T", "N", "T", "N",
  "T", "N", "T", "N", "T", "N", "T", "N", "T", "N",
  "T", "N", "T", "N", "T", "T", "T", "T", "T", "T",
  "T", "T", "T", "T", "T", "T", "T", "T", "N", "T",
  "T", "N", "N", "T", "T", "T", "T", "N", "T", "N",
  "N", "T", "T", "N", "N", "T", "T", "T", "T", "N",
  "T", "N"))

#Deleting contaminated chips
x <- x[,-c(51,55,45,49,56)]
y <- y[ -c(51,55,45,49,56)]
```

```
out <- mipp(x=x, y=y, n.fold=5, p.test=1/3, n.split=20, n.split.eval=100,
            percent.cut = 0.1 , rule="lda")
```

This divides the whole data into two groups for training (two-third) and testing (one-third) ($p.test = 1/3$) and performs the forward gene selection as done with the acute leukemia data. Splitting of the data set into training and independent data sets and then selecting a model for a given split are repeated 20 times ($n.split=20$). This generates the following output.

out\$model

| | | | | | | | | | | |
|----|---|----|-----|--------|-------|--------|--------|-------|--------|----|
| 1 | 1 | 1 | 29 | 0.0263 | 34.75 | 0.9144 | 0.1579 | 13.38 | 0.7042 | |
| 2 | 1 | 2 | 36 | 0.0263 | 35.95 | 0.9461 | 0.0000 | 18.35 | 0.9659 | |
| 3 | 1 | 3 | 30 | 0.0000 | 37.89 | 0.9972 | 0.0000 | 18.88 | 0.9939 | ** |
| 4 | 1 | 4 | 177 | 0.0000 | 38.00 | 0.9999 | 0.0000 | 18.98 | 0.9990 | |
| 5 | 1 | 5 | 18 | 0.0000 | 38.00 | 1.0000 | 0.0000 | 18.96 | 0.9979 | |
| 6 | 1 | 6 | 185 | 0.0000 | 38.00 | 1.0000 | 0.0000 | 19.00 | 1.0000 | |
| 7 | 1 | 7 | 49 | 0.0000 | 38.00 | 1.0000 | 0.0000 | 19.00 | 1.0000 | |
| 8 | 1 | 8 | 163 | 0.0000 | 38.00 | 1.0000 | 0.0000 | 19.00 | 1.0000 | |
| 9 | 1 | 9 | 91 | 0.0000 | 38.00 | 1.0000 | 0.0000 | 19.00 | 1.0000 | |
| 10 | 1 | 10 | 78 | 0.0000 | 38.00 | 1.0000 | 0.0000 | 19.00 | 0.9999 | |
| 11 | 1 | 11 | 148 | 0.0000 | 38.00 | 1.0000 | 0.0000 | 19.00 | 1.0000 | |
| 12 | 1 | 12 | 95 | 0.0000 | 38.00 | 1.0000 | 0.0000 | 19.00 | 1.0000 | * |
| | | | | | | | | | | |
| 13 | 2 | 1 | 163 | 0.0789 | 30.22 | 0.7952 | 0.1053 | 14.39 | 0.7574 | |
| 14 | 2 | 2 | 177 | 0.0000 | 37.12 | 0.9767 | 0.0000 | 18.65 | 0.9818 | |
| 15 | 2 | 3 | 29 | 0.0000 | 37.76 | 0.9936 | 0.0000 | 18.98 | 0.9988 | ** |
| 16 | 2 | 4 | 36 | 0.0000 | 37.97 | 0.9991 | 0.0000 | 18.99 | 0.9996 | |
| 17 | 2 | 5 | 28 | 0.0000 | 37.96 | 0.9991 | 0.0000 | 18.99 | 0.9997 | |
| 18 | 2 | 6 | 185 | 0.0000 | 37.98 | 0.9995 | 0.0000 | 18.99 | 0.9997 | * |
| 19 | 2 | 7 | 182 | 0.0000 | 37.98 | 0.9995 | 0.0000 | 18.98 | 0.9987 | |
| 20 | 2 | 8 | 65 | 0.0000 | 37.99 | 0.9998 | 0.0000 | 18.99 | 0.9993 | |
| 21 | 2 | 9 | 84 | 0.0000 | 37.99 | 0.9999 | 0.0000 | 18.99 | 0.9994 | |
| 22 | 2 | 10 | 18 | 0.0000 | 37.99 | 0.9999 | 0.0000 | 18.99 | 0.9995 | |
| 23 | 2 | 11 | 102 | 0.0000 | 38.00 | 0.9999 | 0.0000 | 18.99 | 0.9993 | |
| 24 | 2 | 12 | 76 | 0.0000 | 38.00 | 0.9999 | 0.0000 | 18.99 | 0.9996 | |
| | | | | | | | | | | |
| . | | | | | | | | | | |
| . | | | | | | | | | | |
| . | | | | | | | | | | |

| | | | | | | | | | | |
|-----|----|----|-----|--------|-------|--------|--------|-------|--------|----|
| 206 | 20 | 1 | 30 | 0.0263 | 35.31 | 0.9291 | 0.1579 | 12.39 | 0.6522 | |
| 207 | 20 | 2 | 78 | 0.0263 | 36.35 | 0.9566 | 0.0526 | 16.54 | 0.8704 | |
| 208 | 20 | 3 | 36 | 0.0000 | 38.00 | 0.9999 | 0.0000 | 18.86 | 0.9924 | ** |
| 209 | 20 | 4 | 51 | 0.0000 | 38.00 | 1.0000 | 0.0526 | 17.05 | 0.8974 | |
| 210 | 20 | 5 | 18 | 0.0000 | 38.00 | 1.0000 | 0.0526 | 16.93 | 0.8911 | |
| 211 | 20 | 6 | 177 | 0.0000 | 38.00 | 1.0000 | 0.1053 | 15.32 | 0.8062 | |
| 212 | 20 | 7 | 28 | 0.0000 | 38.00 | 1.0000 | 0.1053 | 15.03 | 0.7911 | |
| 213 | 20 | 8 | 102 | 0.0000 | 38.00 | 1.0000 | 0.1579 | 13.38 | 0.7040 | |
| 214 | 20 | 9 | 182 | 0.0000 | 38.00 | 1.0000 | 0.1053 | 14.81 | 0.7794 | |
| 215 | 20 | 10 | 148 | 0.0000 | 38.00 | 1.0000 | 0.1053 | 14.98 | 0.7883 | |
| 216 | 20 | 11 | 84 | 0.0000 | 38.00 | 1.0000 | 0.1579 | 13.35 | 0.7027 | |
| 217 | 20 | 12 | 141 | 0.0000 | 38.00 | 1.0000 | 0.1053 | 14.86 | 0.7821 | |

For each split, the parsimonious model identified (denoted as **) is evaluated by an independent 100 splits (n.split.eval=100) generating the following output.

out\$model.eval

| | Split | G1 | G2 | G3 | G4 | G5 | G6 | G7 | G8 | G9 | G10 | G11 | G12 | mean | ER | mean | MiPP | mean | sMiPP |
|-----|-------|-----|-----|-----|-----|-----|-----|----|----|-----|-----|-----|-----|----------|-------|------|------|-----------|-------|
| S1 | 1 | 29 | 36 | 30 | NA | NA | NA | NA | NA | NA | NA | NA | NA | 0.022631 | 18.03 | | | 0.9487025 | |
| S2 | 2 | 163 | 177 | 29 | NA | NA | NA | NA | NA | NA | NA | NA | NA | 0.004210 | 18.70 | | | 0.9840302 | |
| S3 | 3 | 30 | 36 | 185 | NA | NA | NA | NA | NA | NA | NA | NA | NA | 0.000000 | 18.91 | | | 0.9951495 | |
| S4 | 4 | 29 | 102 | 177 | 36 | 84 | NA | NA | NA | NA | NA | NA | NA | 0.010526 | 18.51 | | | 0.9743218 | |
| S5 | 5 | 91 | 177 | 29 | 36 | NA | NA | NA | NA | NA | NA | NA | NA | 0.006842 | 18.62 | | | 0.9798232 | |
| S6 | 6 | 49 | 78 | 185 | 177 | 29 | NA | NA | NA | NA | NA | NA | NA | 0.015789 | 18.33 | | | 0.9646823 | |
| S7 | 7 | 29 | 78 | 91 | 30 | 51 | 102 | 36 | 18 | 76 | 141 | NA | NA | 0.006842 | 18.73 | | | 0.9857517 | |
| S8 | 8 | 30 | 36 | 185 | NA | NA | NA | NA | NA | NA | NA | NA | NA | 0.000000 | 18.91 | | | 0.9951495 | |
| S9 | 9 | 30 | 36 | 78 | NA | NA | NA | NA | NA | NA | NA | NA | NA | 0.000000 | 18.88 | | | 0.9938337 | |
| S10 | 10 | 29 | 185 | 177 | NA | NA | NA | NA | NA | NA | NA | NA | NA | 0.006315 | 18.67 | | | 0.9826242 | |
| S11 | 11 | 29 | 177 | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | 0.014736 | 18.08 | | | 0.9517509 | |
| S12 | 12 | 30 | 36 | 177 | 185 | NA | NA | NA | NA | NA | NA | NA | NA | 0.000000 | 18.97 | | | 0.9986411 | |
| S13 | 13 | 30 | 36 | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | 0.000526 | 18.76 | | | 0.9874807 | |
| S14 | 14 | 49 | 185 | 141 | 65 | 102 | 18 | 30 | 36 | 163 | 95 | 91 | 29 | 0.008947 | 18.62 | | | 0.9799060 | |
| S15 | 15 | 163 | 177 | 185 | 29 | NA | NA | NA | NA | NA | NA | NA | NA | 0.005263 | 18.73 | | | 0.9860384 | |
| S16 | 16 | 49 | 91 | 185 | 78 | NA | NA | NA | NA | NA | NA | NA | NA | 0.013684 | 18.32 | | | 0.9643648 | |
| S17 | 17 | 163 | 29 | 49 | 36 | 148 | 91 | 84 | 30 | 18 | 185 | 141 | 182 | 0.027894 | 17.85 | | | 0.9396381 | |
| S18 | 18 | 29 | 177 | 163 | NA | NA | NA | NA | NA | NA | NA | NA | NA | 0.004210 | 18.69 | | | 0.9840302 | |
| S19 | 19 | 29 | 36 | 185 | 91 | NA | NA | NA | NA | NA | NA | NA | NA | 0.020526 | 18.15 | | | 0.9551247 | |
| S20 | 20 | 30 | 78 | 36 | NA | NA | NA | NA | NA | NA | NA | NA | NA | 0.000000 | 18.88 | | | 0.9938337 | |

| | 5% ER | 50% ER | 95% ER | 5% sMiPP | 50% sMiPP | 95% sMiPP |
|-----|------------|--------|--------|-----------|-----------|-----------|
| S1 | 0.10526316 | 0 | 0 | 0.8083084 | 0.9898493 | 0.9969918 |
| S2 | 0.05263158 | 0 | 0 | 0.9083072 | 0.9949838 | 0.9996731 |
| S3 | 0.00000000 | 0 | 0 | 0.9858440 | 0.9965600 | 0.9990983 |
| S4 | 0.05263158 | 0 | 0 | 0.8829518 | 0.9965193 | 0.9996263 |
| S5 | 0.05263158 | 0 | 0 | 0.8964819 | 0.9946817 | 0.9993379 |
| S6 | 0.05263158 | 0 | 0 | 0.8906567 | 0.9925318 | 0.9998255 |
| S7 | 0.05263158 | 0 | 0 | 0.9074560 | 0.9992884 | 0.9999863 |
| S8 | 0.00000000 | 0 | 0 | 0.9858440 | 0.9965600 | 0.9990983 |
| S9 | 0.00000000 | 0 | 0 | 0.9856433 | 0.9947618 | 0.9987210 |
| S10 | 0.05263158 | 0 | 0 | 0.8974827 | 0.9959758 | 0.9989191 |
| S11 | 0.05526316 | 0 | 0 | 0.8655241 | 0.9726538 | 0.9902048 |
| S12 | 0.00000000 | 0 | 0 | 0.9968295 | 0.9991665 | 0.9998100 |
| S13 | 0.00000000 | 0 | 0 | 0.9744228 | 0.9896188 | 0.9954914 |
| S14 | 0.05526316 | 0 | 0 | 0.8777725 | 0.9992962 | 0.9999910 |
| S15 | 0.05263158 | 0 | 0 | 0.9071621 | 0.9976359 | 0.9997154 |
| S16 | 0.05263158 | 0 | 0 | 0.8896781 | 0.9860472 | 0.9983609 |
| S17 | 0.10526316 | 0 | 0 | 0.7931386 | 0.9778348 | 0.9995296 |
| S18 | 0.05263158 | 0 | 0 | 0.9083072 | 0.9949838 | 0.9996731 |
| S19 | 0.05263158 | 0 | 0 | 0.8801255 | 0.9952021 | 0.9989584 |
| S20 | 0.00000000 | 0 | 0 | 0.9856433 | 0.9947618 | 0.9987210 |

3.3 Sequential selection

Good classifying (masked) genes may be masked by other better classifying (masking) genes, so the masked genes may be discovered if the masking genes are not present. Therefore, it is worth selecting gene models after removing genes selected in the previous runs. This sequential selection can be performed by manually removing the genes selected in the previous runs. For users' convenience, the *MiPP* package enables one to perform such a sequential selection process automatically many times.

For the acute leukemia data with an independent test set, the sequential analysis can be performed by the following arguments in the `mipp.seq` function:

```
out <- mipp.seq(x=x.train, y=y.train, x.test=x.test, y.test=y.test, n.seq=3)
```

The argument `n.seq=3` means that the sequential selection is performed 3 times after removing all the genes in the selected gene models. For the colon cancer data with no independent test set, the sequential analysis can be performed by the following arguments:

```
out <- mipp.seq(x=x, y=y, n.seq=3, cutoff.sMiPP=0.7)
```

By the argument *cutoff.sMiPP=0.7*, the gene models with 5% sMiPP > 0.7 are selected, so all the genes in the selected models are removed for the next run. All the arguments in the `mipp.seq` function can also be used in the `mipp.seq` function.

Reference

Soukup M, Cho H, and Lee JK (2005). Robust classification modeling on microarray data using misclassification penalized posterior, *Bioinformatics*, 21 (Suppl): i423-i430.

Soukup M and Lee JK (2004). Developing optimal prediction models for cancer classification using gene expression data, *Journal of Bioinformatics and Computational Biology*, 1(4) 681-694.