# test adjust pval

*John Stansfield*

*December 13, 2017*

```r
library(readr)
library(data.table)
library(HiCcompare)
library(dplyr)
library(pROC)
```

```r
make_changes <- function(hic.table, N = 300, FC = 2, quant = 0.25) {
  # introduce changes
  changes <- sample(1:nrow(hic.table), size = N)
  whichIF = ifelse(hic.table[changes, ]$M < 0, -1, 1)
  newIF1 = FC^whichIF * hic.table[changes,]$IF2
  newIF1 = (round(newIF1))
  newIF1[newIF1 < 0.01] <- 1
  hic.table[changes,]$IF1 = newIF1
  hic.table = hic.table[, M := log2(IF2/IF1)]

  # make truth vector
  truth <- rep(0, nrow(hic.table))
  truth[changes] <- 1
  hic.table[, truth := truth]

  # normalize
  hic.table <- hic_loess(hic.table, Plot = FALSE)
  hic.table <- hic_diff(hic.table, Plot = TRUE)


  # roc_result <- roc(response = result$truth, predictor = result$p.val)
  # return(roc_result)
  return(hic.table)
}
```

```r
# function to adjust p-values based on distance holm method
adjust_pval <- function(hic.table, theta = 1, alpha = 0.05) {
  # get distance percentages
  d_percent <- ((hic.table$D + 1) / max(hic.table$D + 1)) * 100
  # calculate threshold for checking if p-value is < thresh to be a rejection of the null hypothesis
  threshhold <- alpha / (d_percent * theta)
  P <- hic.table$p.val
  P[P > threshhold] <- 1
  hic.table[, thresh := threshhold]
  hic.table[, p.adj := P]
  MD.plot2(hic.table$adj.M, hic.table$D, hic.table$p.adj)
  return(hic.table)
}
```

# Test adjust_pval

N = 300 FC = 1.5

```
hic.table <- dplfc1_2[[1]]
backup.table <- hic.table
```

```
hic.table <- make_changes(hic.table, N = 300, FC = 1.5, quant = 0.25)
```
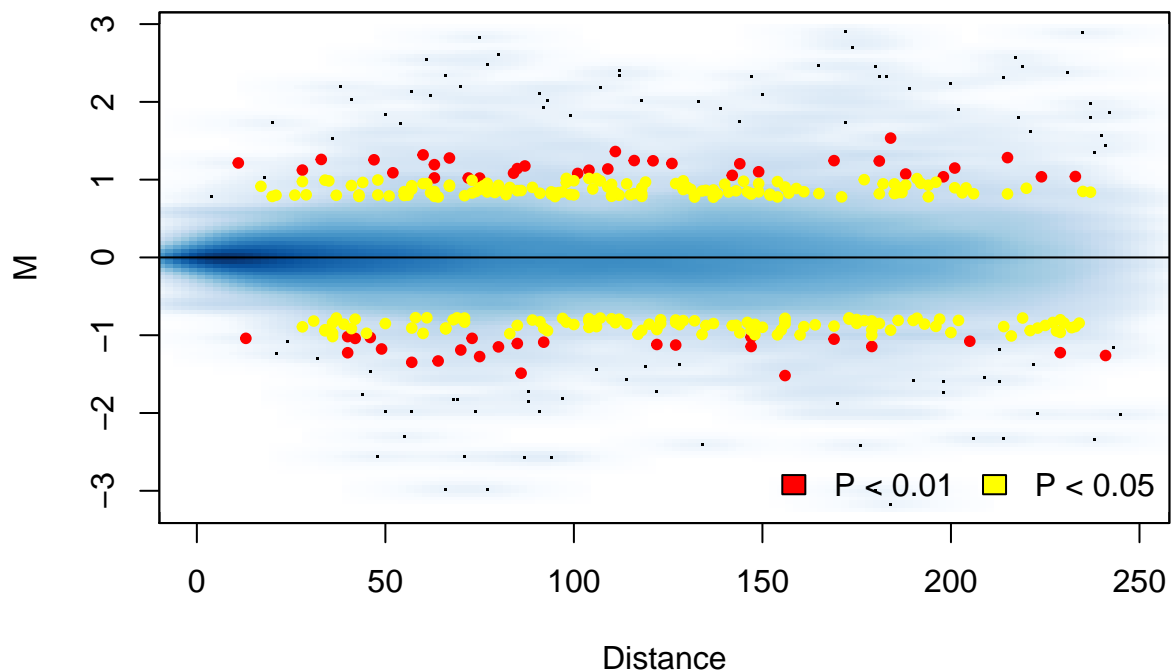
```
## Warning in `[<-.data.table`(`*tmp*`, changes, , value = structure(list(chr1
## = c("chr1", : Coerced 'double' RHS to 'integer' to match the column's
## type; may have truncated precision. Either change the target column to
## 'double' first (by creating a new 'double' vector length 25836 (nrows of
## entire table) and assign that; i.e. 'replace' column), or coerce RHS to
## 'integer' (e.g. 1L, NA_[real|integer]_, as.*, etc) to make your intent
## clear and for speed. Or, set the column type correctly up front when you
## create the table and stick to it, please.
```

```
## Span for loess: 0.899920444117736
```
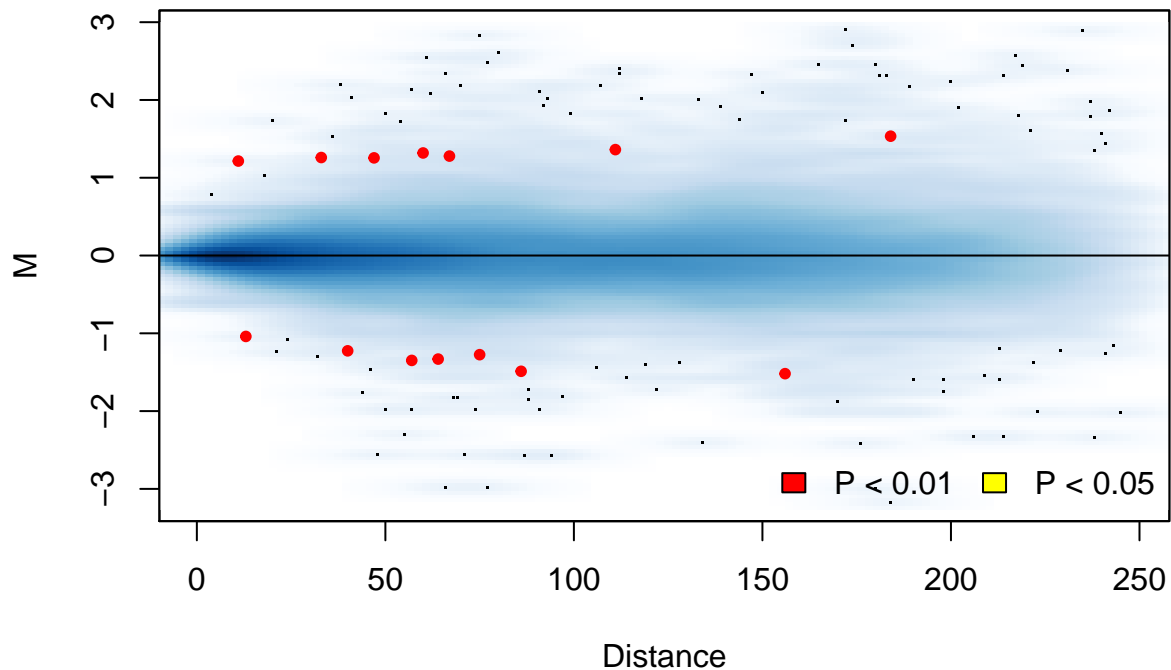
```
## GCV for loess: 6.0423611127932e-06
```

```
## AIC for loess: -0.856847627654749
```



```
hic.table <- adjust_pval(hic.table)
```

## MD Plot



**N = 300 FC = 2**

```
hic.table <- backup.table

hic.table <- make_changes(hic.table, N = 300, FC = 2, quant = 0.25)
```
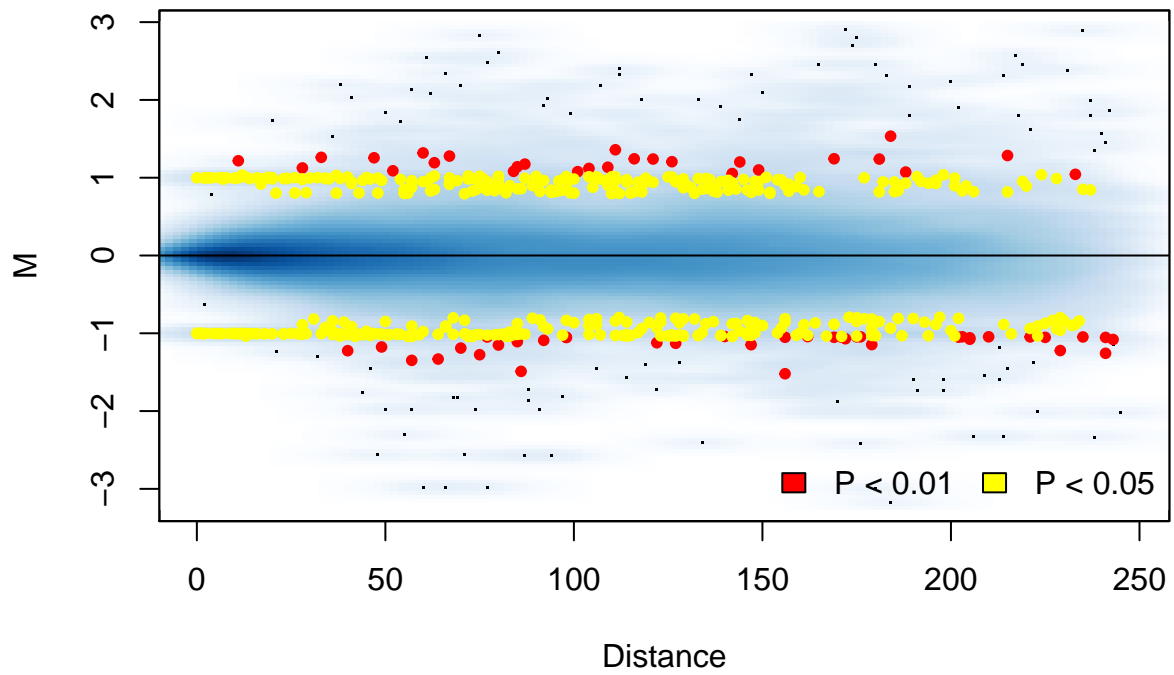
```
## Warning in `[<-.data.table`(`*tmp*`, changes, , value = structure(list(chr1
## = c("chr1", : Coerced 'double' RHS to 'integer' to match the column's
## type; may have truncated precision. Either change the target column to
## 'double' first (by creating a new 'double' vector length 25836 (nrows of
## entire table) and assign that; i.e. 'replace' column), or coerce RHS to
## 'integer' (e.g. 1L, NA_[real|integer]_, as.*, etc) to make your intent
## clear and for speed. Or, set the column type correctly up front when you
## create the table and stick to it, please.
```

```
## Span for loess: 0.898986621776549
```

```
## GCV for loess: 6.32549776540058e-06
```
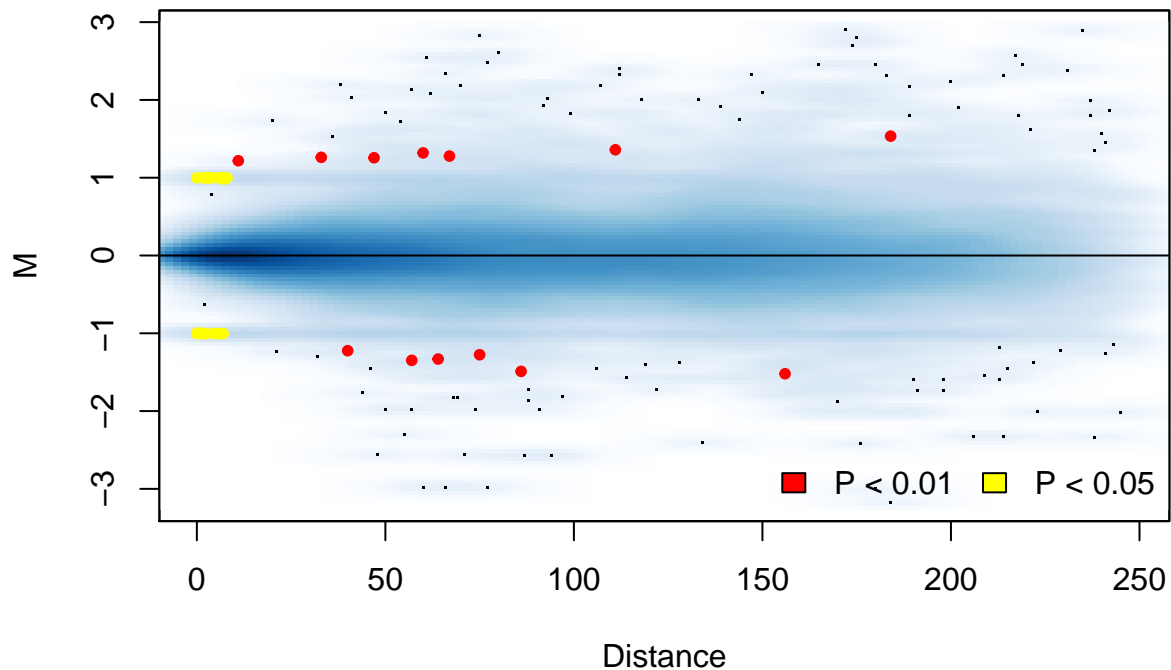
```
## AIC for loess: -0.811053549869331
```

## MD Plot



```
hic.table <- adjust_pval(hic.table)
```

**MD Plot**



**N = 300 FC = 3**

```
hic.table <- backup.table

hic.table <- make_changes(hic.table, N = 300, FC = 3, quant = 0.25)
```
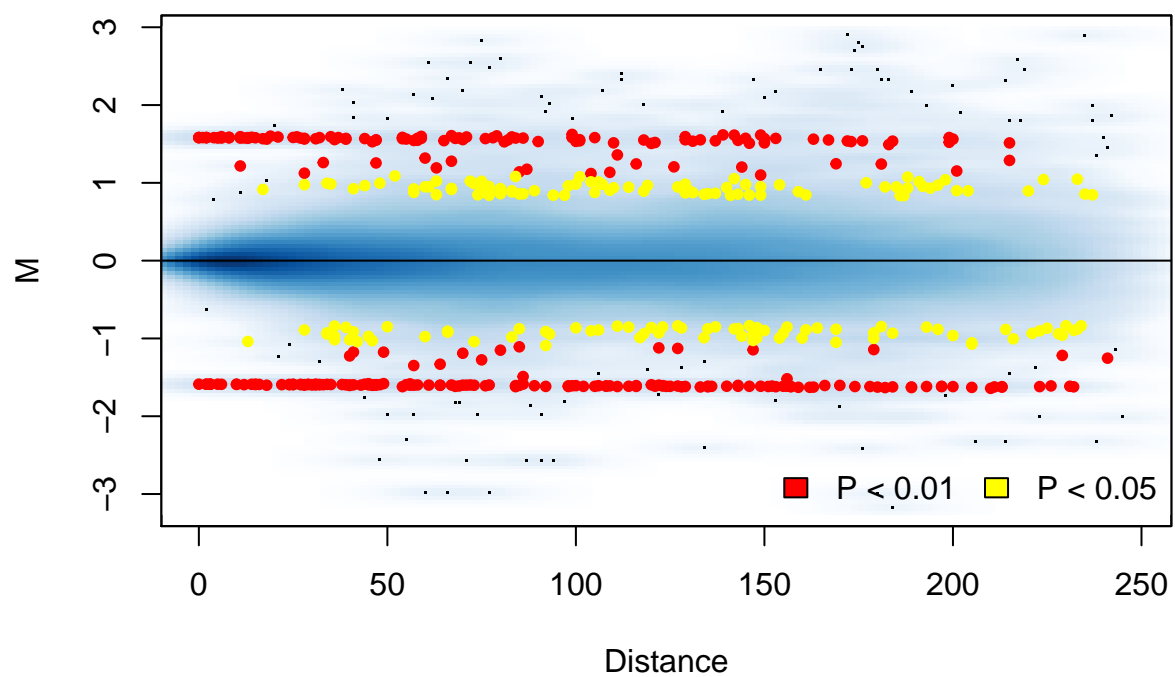
```
## Warning in `[<-.data.table`(`*tmp*`, changes, , value = structure(list(chr1
## = c("chr1", : Coerced 'double' RHS to 'integer' to match the column's
## type; may have truncated precision. Either change the target column to
## 'double' first (by creating a new 'double' vector length 25836 (nrows of
## entire table) and assign that; i.e. 'replace' column), or coerce RHS to
## 'integer' (e.g. 1L, NA_[real|integer]_, as.*, etc) to make your intent
## clear and for speed. Or, set the column type correctly up front when you
## create the table and stick to it, please.
```

```
## Span for loess: 0.899680913335134
```

```
## GCV for loess: 7.0296167553633e-06
```
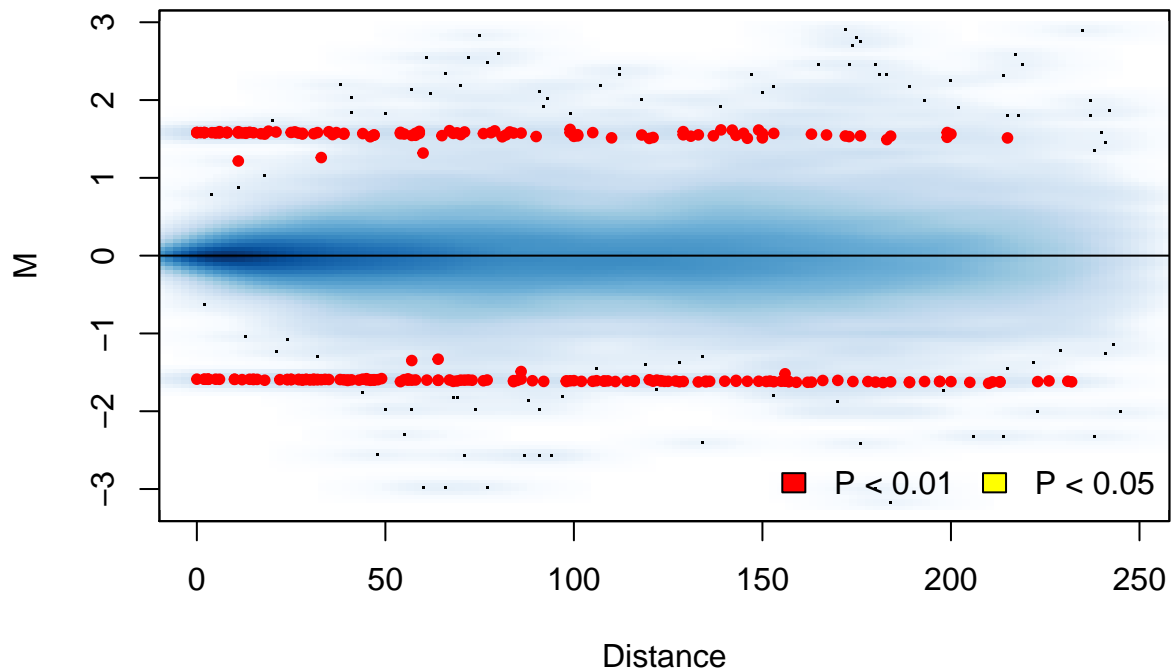
```
## AIC for loess: -0.705510236834998
```

**MD Plot**



```
hic.table <- adjust_pval(hic.table)
```

**MD Plot**



N = 300 FC = 4

```
hic.table <- backup.table

hic.table <- make_changes(hic.table, N = 300, FC = 4, quant = 0.25)
```
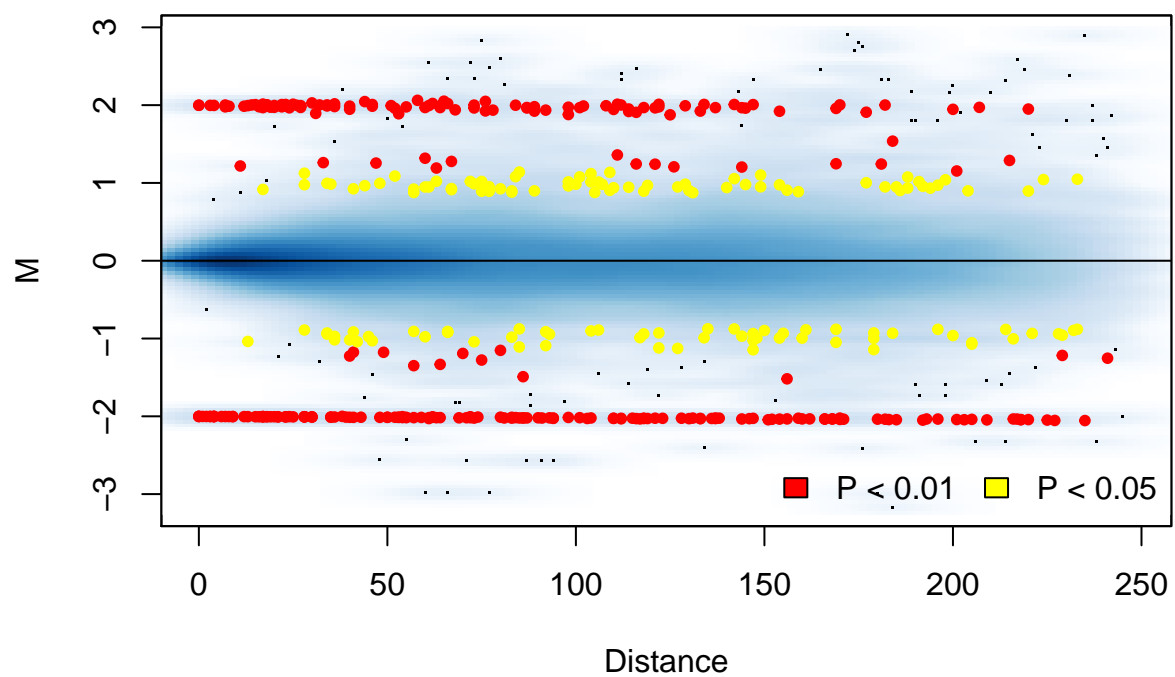
```
## Warning in `[<-.data.table`(`*tmp*`, changes, , value = structure(list(chr1
## = c("chr1", : Coerced 'double' RHS to 'integer' to match the column's
## type; may have truncated precision. Either change the target column to
## 'double' first (by creating a new 'double' vector length 25836 (nrows of
## entire table) and assign that; i.e. 'replace' column), or coerce RHS to
## 'integer' (e.g. 1L, NA_[real|integer]_, as.*, etc) to make your intent
## clear and for speed. Or, set the column type correctly up front when you
## create the table and stick to it, please.
```

```
## Span for loess: 0.899945506167573
```

```
## GCV for loess: 7.69563589611078e-06
```
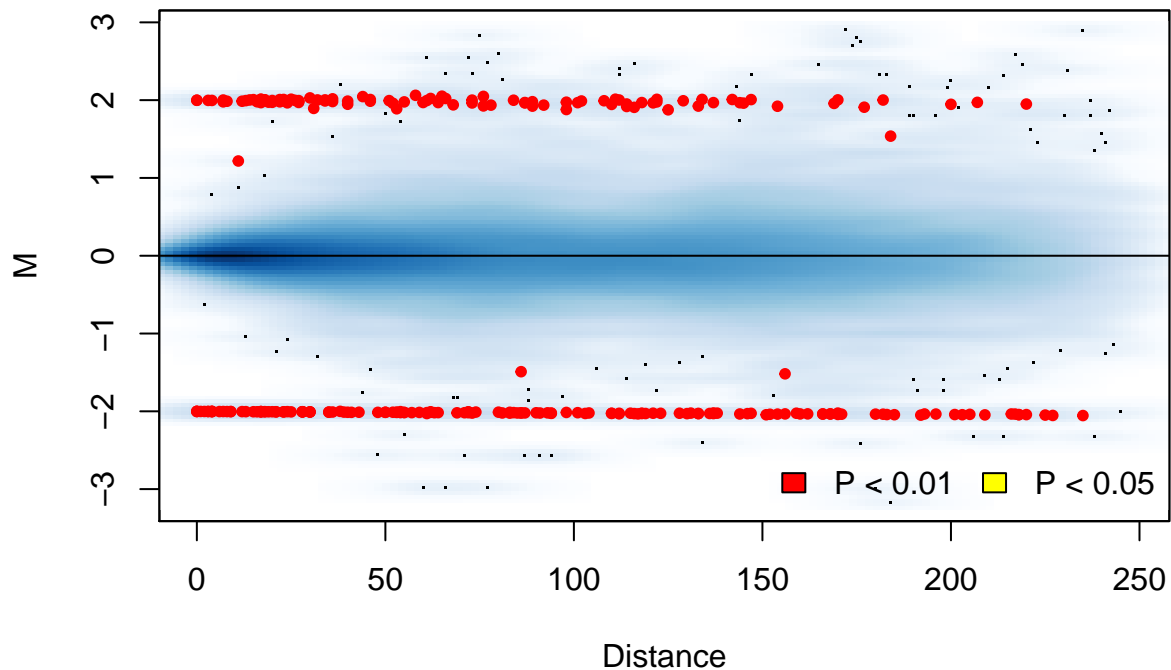
```
## AIC for loess: -0.614989079748196
```

**MD Plot**

```
hic.table <- adjust_pval(hic.table)
```

**MD Plot**



N = 50 FC = 3

```
hic.table <- backup.table

hic.table <- make_changes(hic.table, N = 50, FC = 4, quant = 0.25)
```
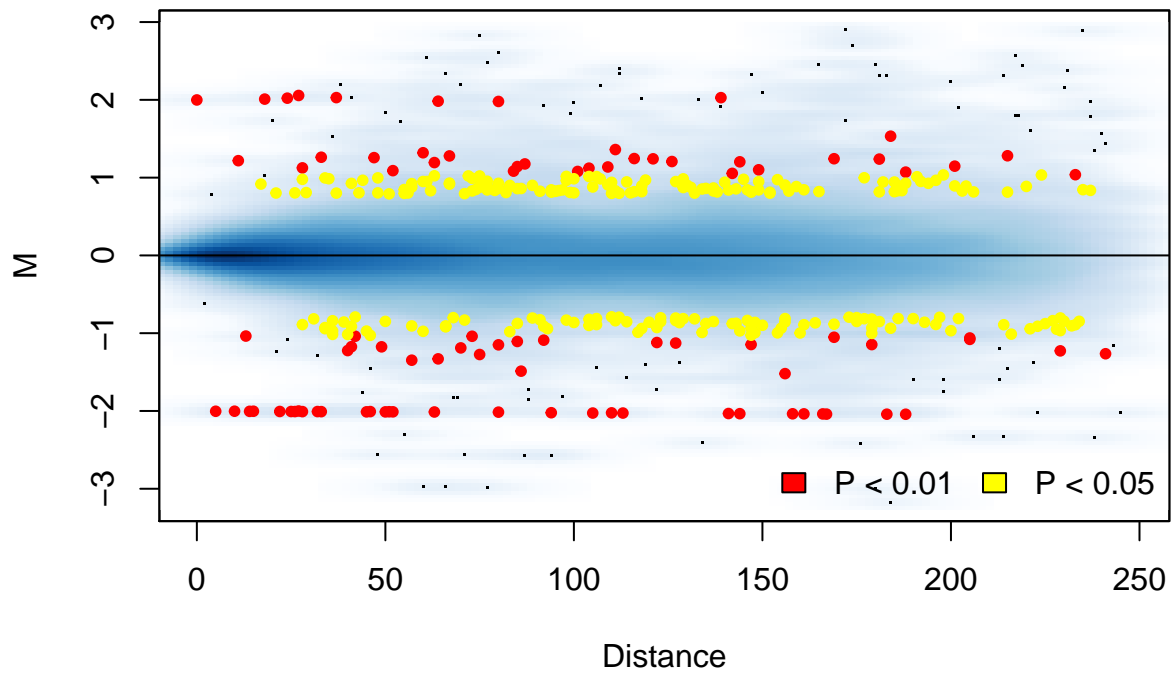
```
## Warning in `[<-.data.table`(`*tmp*`, changes, , value = structure(list(chr1
## = c("chr1", : Coerced 'double' RHS to 'integer' to match the column's
## type; may have truncated precision. Either change the target column to
## 'double' first (by creating a new 'double' vector length 25836 (nrows of
## entire table) and assign that; i.e. 'replace' column), or coerce RHS to
## 'integer' (e.g. 1L, NA_[real|integer]_, as.*, etc) to make your intent
## clear and for speed. Or, set the column type correctly up front when you
## create the table and stick to it, please.
```

```
## Span for loess: 0.899678144589143
```

```
## GCV for loess: 6.25118331385968e-06
```
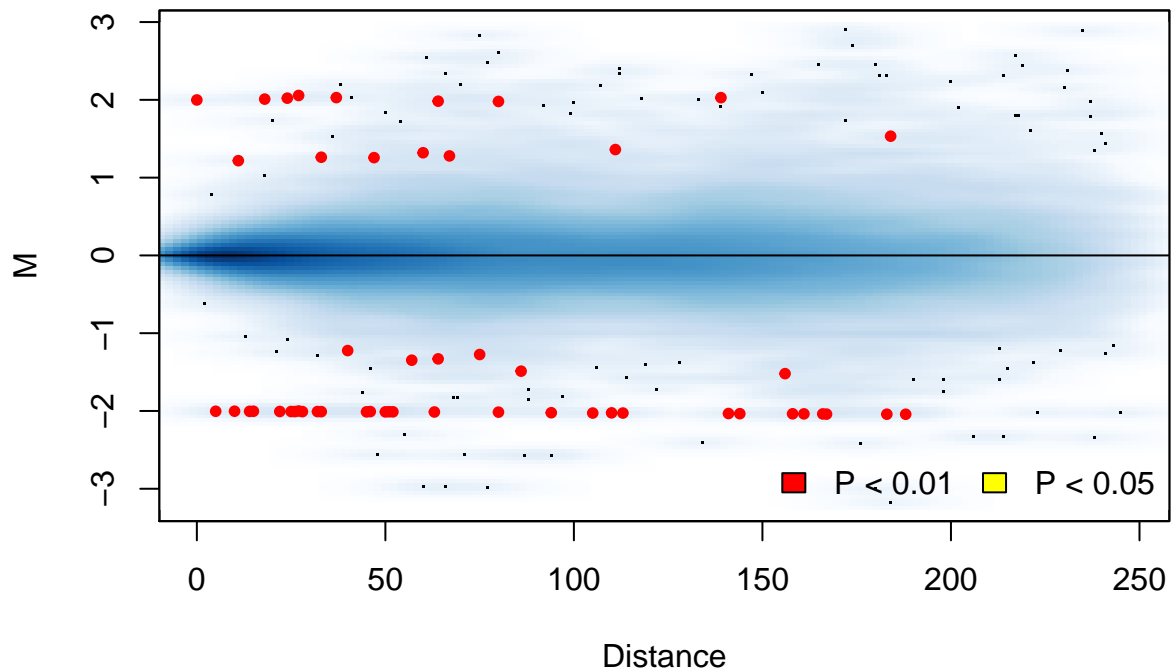
```
## AIC for loess: -0.82287164894865
```

**MD Plot**



```
hic.table <- adjust_pval(hic.table)
```

## MD Plot



N = 50 FC = 4

```
hic.table <- backup.table

hic.table <- make_changes(hic.table, N = 50, FC = 4, quant = 0.25)
```

```
## Warning in `[<-.data.table`(`*tmp*`, changes, , value = structure(list(chr1
## = c("chr1", : Coerced 'double' RHS to 'integer' to match the column's
## type; may have truncated precision. Either change the target column to
## 'double' first (by creating a new 'double' vector length 25836 (nrows of
## entire table) and assign that; i.e. 'replace' column), or coerce RHS to
## 'integer' (e.g. 1L, NA_[real|integer]_, as.*, etc) to make your intent
## clear and for speed. Or, set the column type correctly up front when you
## create the table and stick to it, please.
```
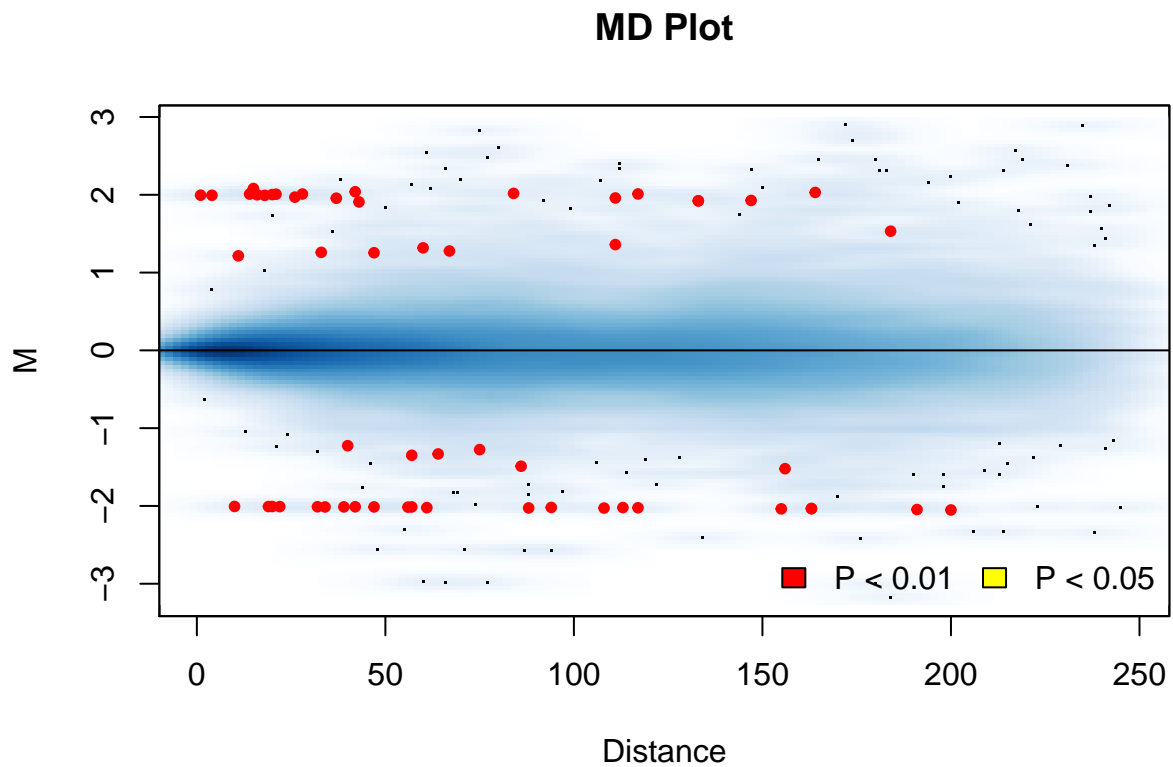
```
## Span for loess: 0.899929958808223
```

```
## GCV for loess: 6.25286846118057e-06
```

```
## AIC for loess: -0.822602165678567
```

**MD Plot**



```
hic.table <- adjust_pval(hic.table)
```

**MD Plot**
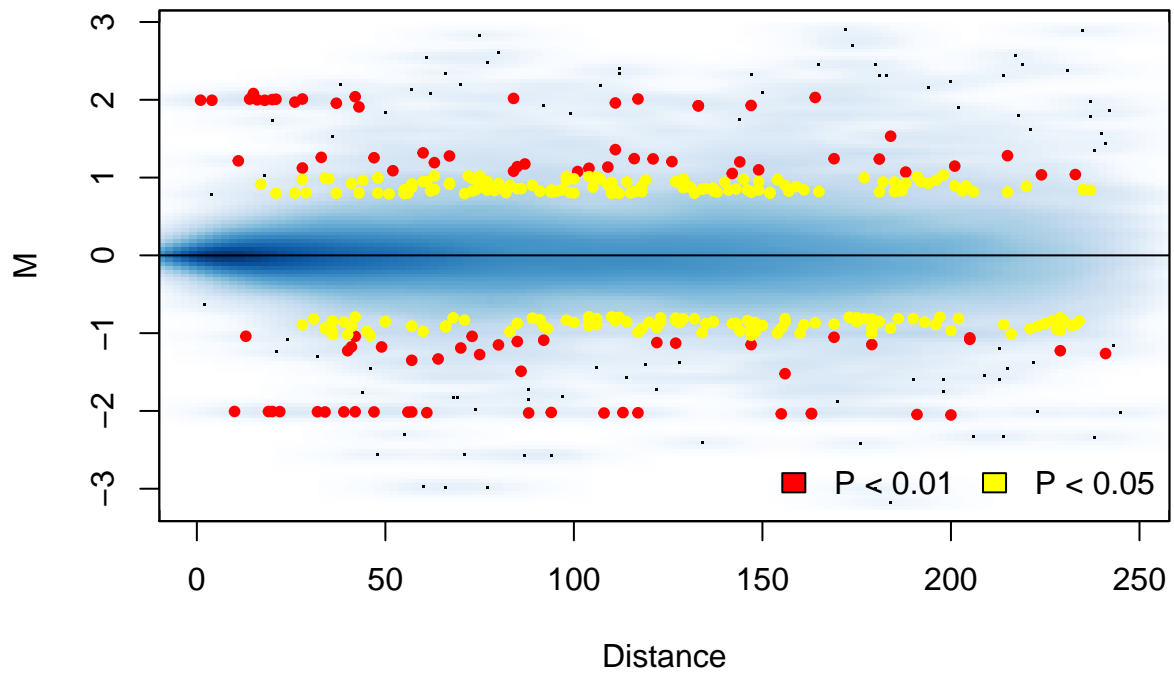


**No changes**

```
hic.table <- backup.table
hic.table <- hic_loess(hic.table, Plot = FALSE)

## Span for loess: 0.89992480205479

## GCV for loess: 5.9608868807595e-06

## AIC for loess: -0.870423201341588

hic.table <- hic_diff(hic.table, Plot = FALSE)
hic.table <- adjust_pval(hic.table)
```

# MD Plot



P < 0.01   P < 0.05

Distance

M