Statistical Significance of DMLs by genomic regions

Cemile Balkas

2025-07-03

1 Introduction

This report presents the results of Fisher's exact tests on differentially methylated loci (DMLs) across genomic features compared to all tested CpG sites.

Fisher's exact tests were used to compare hyper- and hypomethylated DMLs across features like transcription start sites, exons, and intergenic regions. The analysis includes both exclusive overlaps (each CpG counted once) and non-exclusive overlaps (allowing overlaps with multiple features).

The aim is to see if DMLs are randomly distributed or if they tend to cluster in certain regions.

1.1 Load Data

```
df_dml_feature_counts <- read.csv("dml_feature_overlap_output/overlap_counts_exclusive.csv")
total_cpg <- sum(df_dml_feature_counts$All_CpG_Count)
df_dml_feature_counts$Percent_of_Total_CpGs <- round(100 * df_dml_feature_counts$All_CpG_Count / total_
kable(df_dml_feature_counts, caption = "Exclusive counts per genomic feature")
```

Table 1: Exclusive counts per genomic feature

Feature	All_CpG_Count	HyperDMLs	HypoDMLs	Percent_of_Total_CpGs
tss	492557	13708	11190	1.81
downstream_1kb	268352	2846	12471	0.99
5utrs	341338	11626	9960	1.26
3utrs	653184	5657	22290	2.41
exons	1096878	17683	36828	4.04
introns	11534889	282730	605257	42.49
intergenic	12758595	593117	1064491	47.00

```
df_overlap_inclusive <- read.csv("dml_feature_overlap_output/overlap_counts.csv")
df_overlap_inclusive$Percent_of_Total_CpGs <- round(100 * df_overlap_inclusive$All_CpG_Count / total_cpg
kable(df_overlap_inclusive, caption = "Non-exclusive counts per genomic feature")
```

Table 2: Non-exclusive counts per genomic feature

Feature	All_CpG_Count	HyperDMLs	HypoDMLs	Percent_of_Total_CpGs
tss	492557	13708	11190	1.81
downstream_1kb	285537	3038	13062	1.05
5utrs	548166	17469	14011	2.02
3utrs	690312	5995	23960	2.54
exons	2395519	42456	76664	8.82

Feature	All_CpG_Count	HyperDMLs	HypoDMLs	Percent_of_Total_CpGs
introns	11635556	284544	608747	42.86
intergenic	12759559	593141	1064521	47.00

1.2 Summarise totals

```
# Get total background counts
total_cpg <- sum(df_dml_feature_counts$All_CpG_Count)
total_hyper <- sum(df_dml_feature_counts$HyperDMLs)
total_hypo <- sum(df_dml_feature_counts$HypoDMLs)

# Show totals
total_cpg
```

```
## [1] 27145793
```

```
total_hyper
```

```
## [1] 927367
```

```
total_hypo
```

```
## [1] 1762487
```

1.3 Fisher's Exact test with box analogy

To determine whether differentially methylated loci (DMLs) are enriched in specific genomic features (eg TSS, exons), the distribution of DMLs is compared to all tested CpG sites using Fisher's Exact Test.

This can be explained by this analogy:

- Imagine a box containing 1,000 colored balls, each representing a tested CpG site.
  - 200 balls are red, representing CpGs in a particular genomic feature (eg TSS).
  - 800 balls are white representing CpGs not in that feature.
- If a total of 300 balls are randomly drawn from the box to represent DMLs.
  - Among these there are 90 red balls — ie 90 DMLs fall in TSS.
  - The question: Is observing 90 red balls (DMLs in TSS) more than expected by chance?

If DMLs were randomly distributed, the expected number of red DMLs would be:

300 * (200 / 1000) = 60

90 > 60 but is this statistically significant?

Fisher's Exact Test calculates the probability of seeing 90 or more red balls in a random draw of 300, given the background proportions using the hypergeometric distribution.

For Fisher's test to be valid:

- CpGs must fall into mutually exclusive groups: each site is either a DML or not, and either in the feature or not.
- All CpGs must come from the same tested background set, not the whole genome.

In this analysis, Fisher's test is applied first to exclusive overlaps, where each CpG is assigned to only one feature, and then to hyper dmls in non-exclusive overlaps.

```
box_analogy_CpGs <- data.frame(
  CpG_in_tss = c(90, 110),
  CpG_not_in_tss = c(210, 590),
```

```
  row.names = c("DML", "Non-DML")
)

box_analogy_CpGs
```

```
##         CpG_in_tss CpG_not_in_tss
## DML             90            210
## Non-DML        110            590
```

```
fisher.test(box_analogy_CpGs)
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  box_analogy_CpGs
## p-value = 4.764e-07
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  1.645304 3.202829
## sample estimates:
## odds ratio
##   2.296597
```

1.4 Fisher's test example: Hyper-DMLs in TSS

Total CpGs tested (total balls in the box): 27,145,793 CpGs in TSS (red balls): 492,557

Hyper-DMLs (balls drawn from the box): 927,367 Hyper-DMLs in TSS (red balls drawn): 13,708

927367 * (492557 / 27145793) = 16,827

But only 13,708 observed : fewer than expected... statistically significant?

```
# Totals
total_cpgs <- 27145793
total_hyper_dmls <- 927367
tss_cpgs <- 492557
tss_hyper_dmls <- 13708

# Calculate values
hyperdml_in_tss <- tss_hyper_dmls
hyperdml_not_in_tss <- total_hyper_dmls - hyperdml_in_tss

non_dml_total <- total_cpgs - total_hyper_dmls
non_dml_in_tss <- tss_cpgs - hyperdml_in_tss
non_dml_not_in_tss <- non_dml_total - non_dml_in_tss

# Create table
hyperdml_CpGs <- data.frame(
  CpG_in_tss = c(hyperdml_in_tss, non_dml_in_tss),
  CpG_not_in_tss = c(hyperdml_not_in_tss, non_dml_not_in_tss),
  row.names = c("HyperDML", "Non-DML")
)

hyperdml_CpGs
```

```
##          CpG_in_tss CpG_not_in_tss
```

```
## HyperDML      13708         913659
## Non-DML      478849       25739577
```

```
fisher.test(hyperdml_CpGs)
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  hyperdml_CpGs
## p-value < 2.2e-16
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.7927592 0.8204087
## sample estimates:
## odds ratio
##  0.8064758
```

1.5 Run Fisher's Test for Hyper DMLs

Are hypermethylated DMLs more common in this feature than we would expect by chance?

We test if the proportion of DMLs in this feature is significantly different from what you'd expect based on the overall CpG background...

```
options(scipen = 0) # turn off scientific notation

# Run Fisher's test for all features for HyperDMLs
fisher_results_hyper <- data.frame()

for (i in 1:nrow(df_dml_feature_counts)) {
  # table logic
  #               In_feature    Not_in_feature      Total
  # DML               a               b           total_hyper_dmls
  # Not_DML           c               d                .
  # Total     total_cpg_in_feature    .            total_cpg

  total_cpg <- sum(df_dml_feature_counts$All_CpG_Count)
  total_cpg_in_feature <- df_dml_feature_counts$All_CpG_Count[i]
  total_hyper_dmls <- sum(df_dml_feature_counts$HyperDMLs)
  total_hyper_dmls_in_feature <- df_dml_feature_counts$HyperDMLs[i]

  a <- total_hyper_dmls_in_feature
  b <- total_hyper_dmls - a
  c <- total_cpg_in_feature - a
  d <- (total_cpg - total_cpg_in_feature) - b

  contingency <- data.frame(
    In_Feature = c(a, c),
    Not_In_Feature = c(b, d),
    row.names = c("DML", "Not_DML")
  )

  test <- fisher.test(contingency)

  fisher_pval <- test$p.value
  odds_ratio <- as.numeric(test$estimate)
```

```
  fisher_results_hyper <- rbind(fisher_results_hyper, data.frame(
    Feature = df_dml_feature_counts$Feature[i],
    CpG_count = total_cpg_in_feature,
    Hyper_dmls = total_hyper_dmls_in_feature,
    p_value = fisher_pval,
    OddsRatio = odds_ratio
  ))
}

kable(fisher_results_hyper, caption = "Fisher's test results for hypermethylated DMLs (exclusive feature
```

Table 3: Fisher's test results for hypermethylated DMLs (exclusive features)

Feature	CpG_count	Hyper_dmls	p_value	OddsRatio
tss	492557	13708	0.0000000	0.8064758
downstream_1kb	268352	2846	0.0000000	0.3009054
5utrs	341338	11626	0.7435551	0.9968582
3utrs	653184	5657	0.0000000	0.2423706
exons	1096878	17683	0.0000000	0.4528115
introns	11534889	282730	0.0000000	0.5833581
intergenic	12758595	593117	0.0000000	2.0497974

1.6 Run Fisher's Test for Hypo DMLs

```
fisher_results_hypo <- data.frame()

for (i in 1:nrow(df_dml_feature_counts)) {
  # table logic
  #               In_feature    Not_in_feature      Total
  # DML               a               b           total_hypo_dmls
  # Not_DML           c               d                .
  # Total     total_cpg_in_feature    .            total_cpg

  total_cpg <- sum(df_dml_feature_counts$All_CpG_Count)
  total_cpg_in_feature <- df_dml_feature_counts$All_CpG_Count[i]
  total_hypo_dmls <- sum(df_dml_feature_counts$HypoDMLs)
  total_hypo_dmls_in_feature <- df_dml_feature_counts$HypoDMLs[i]

  a <- total_hypo_dmls_in_feature
  b <- total_hypo_dmls - a
  c <- total_cpg_in_feature - a
  d <- (total_cpg - total_cpg_in_feature) - b

  contingency <- data.frame(
    In_Feature = c(a, c),
    Not_In_Feature = c(b, d),
    row.names = c("DML", "Not_DML")
  )

  test <- fisher.test(contingency)
```

```
  fisher_pval <- test$p.value
  odds_ratio <- as.numeric(test$estimate)

  fisher_results_hypo <- rbind(fisher_results_hypo, data.frame(
    Feature = df_dml_feature_counts$Feature[i],
    CpG_count = total_cpg_in_feature,
    Hypo_dmls = total_hypo_dmls_in_feature,
    p_value = fisher_pval,
    OddsRatio = odds_ratio
  ))
}

kable(fisher_results_hypo, caption = "Fisher's test results for hypomethylated DMLs (exclusive features)
```

Table 4: Fisher's test results for hypomethylated DMLs (exclusive features)

Feature	CpG_count	Hypo_dmls	p_value	OddsRatio
tss	492557	11190	0	0.3305424
downstream_1kb	268352	12471	0	0.6997752
5utrs	341338	9960	0	0.4296473
3utrs	653184	22290	0	0.5025644
exons	1096878	36828	0	0.4896871
introns	11534889	605257	0	0.6916617
intergenic	12758595	1064491	0	1.7852292

```
fe_hyper <- c()
for (i in 1:nrow(fisher_results_hyper)) {
  fold_enrichment = (fisher_results_hyper$Hyper_dmls[i] / total_hyper_dmls) / (fisher_results_hyper$CpG_
  print(fold_enrichment)
  fe_hyper <- c(fe_hyper, fold_enrichment)
}
```

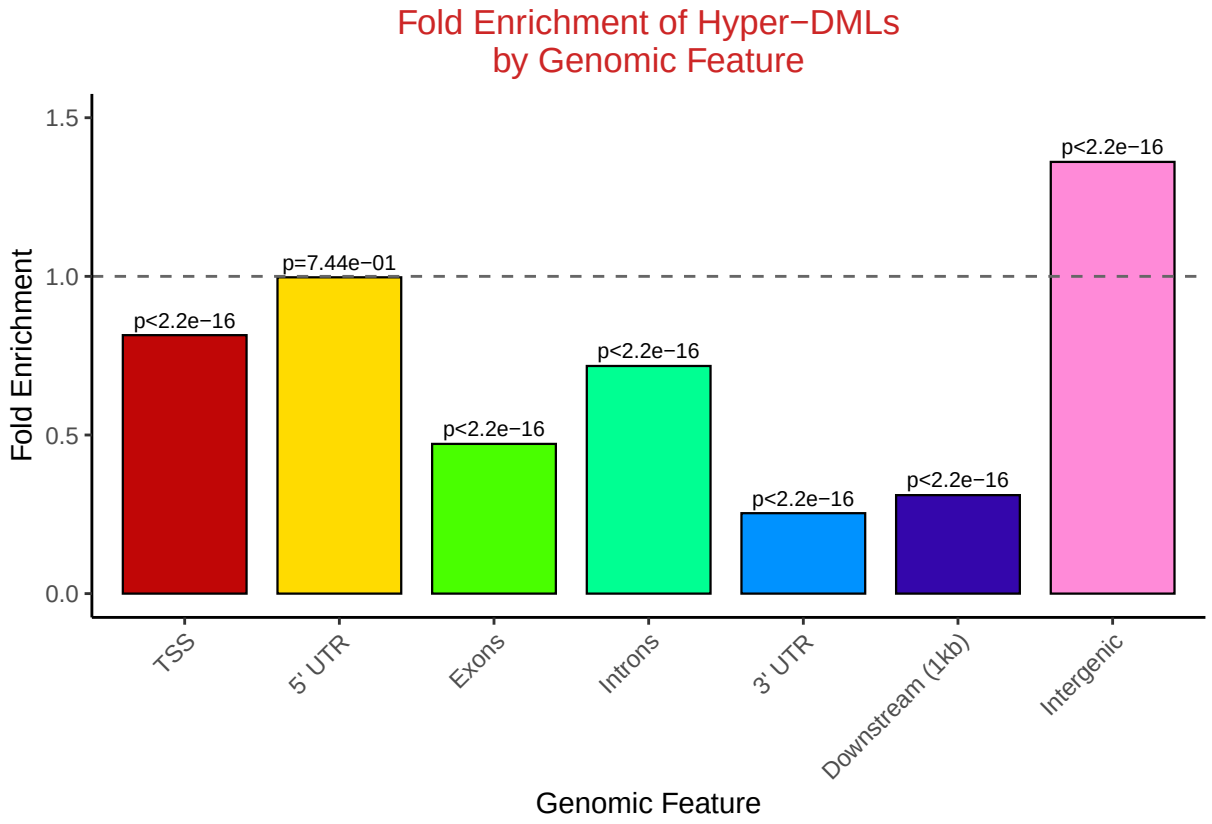
```
## [1] 0.8146452
## [1] 0.3104423
## [1] 0.9970033
## [1] 0.2535138
## [1] 0.4718984
## [1] 0.7174793
## [1] 1.360782
```

```
fisher_results_hyper$FoldEnrichment <- fe_hyper

fe_hypo <- c()
for(i in 1:nrow(fisher_results_hypo)) {
  fold_enrichment = (fisher_results_hypo$Hypo_dmls[i] / total_hypo_dmls) / (fisher_results_hypo$CpG_cou
  fe_hypo <- c(fe_hypo, fold_enrichment)
}

fisher_results_hypo$FoldEnrichment <- fe_hypo
```

The Fisher's exact tests compare the proportion of hyper- and hypomethylated DMLs within each genomic feature to the background of all tested CpG sites.

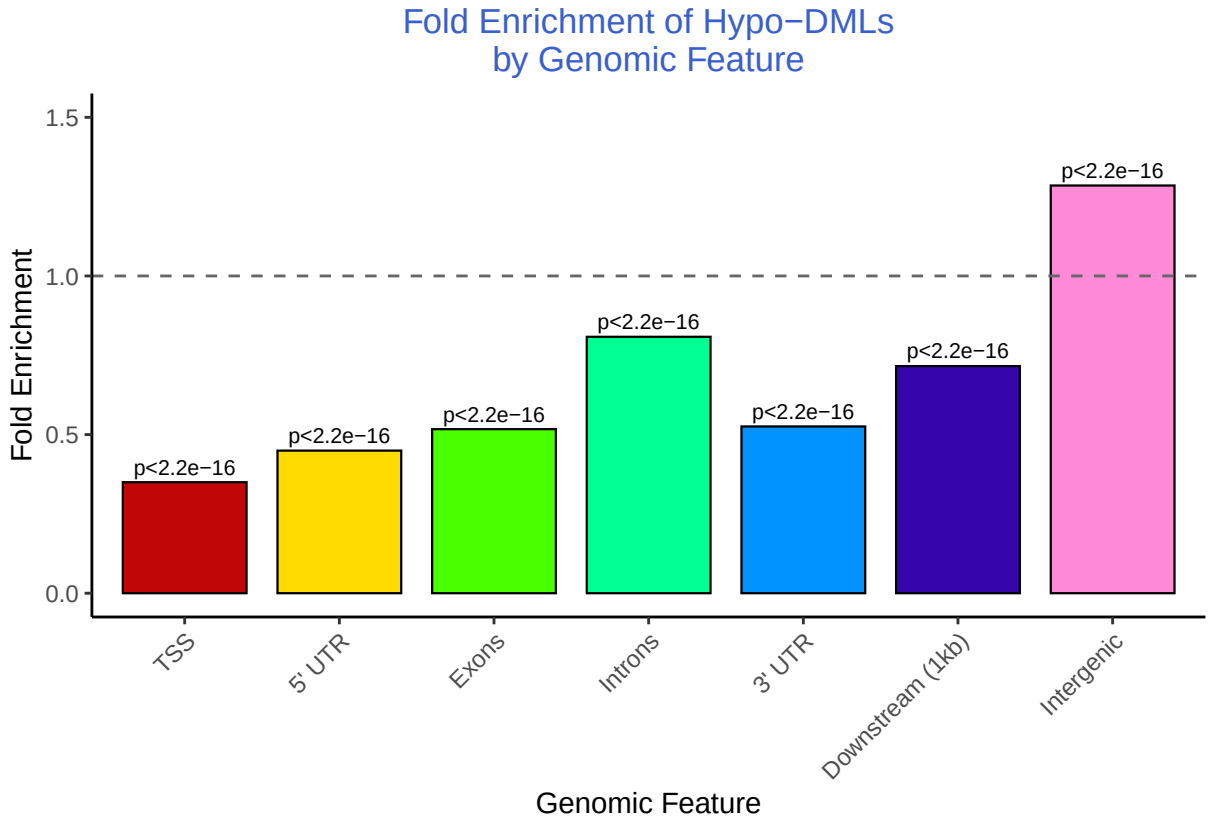
```r
# Add DML type column if not already present
fisher_results_hyper$DML_Type <- "hyper"
fisher_results_hypo$DML_Type <- "hypo"

# Replace all p-values < 2.2e-16 with display value
fisher_results_hyper$p_display <- ifelse(fisher_results_hyper$p_value < 2.2e-16, "p<2.2e-16",
                                         paste0("p=", formatC(fisher_results_hyper$p_value, format = "e

fisher_results_hypo$p_display <- ifelse(fisher_results_hypo$p_value < 2.2e-16, "p<2.2e-16",
                                         paste0("p=", formatC(fisher_results_hyper$p_value, format = "e

# Rename bar column names
label_map <- c(
  tss = "TSS",
  `5utrs` = "5' UTR",
  exons = "Exons",
  introns = "Introns",
  `3utrs` = "3' UTR",
  downstream_1kb = "Downstream (1kb)",
  intergenic = "Intergenic"
)

# Apply to both dataframes
fisher_results_hyper$FeatureLabel <- label_map[as.character(fisher_results_hyper$Feature)]
fisher_results_hypo$FeatureLabel <- label_map[as.character(fisher_results_hypo$Feature)]

# Set factor order for feature
feature_order <- c("TSS", "5' UTR", "Exons", "Introns", "3' UTR", "Downstream (1kb)", "Intergenic")
fisher_results_hyper$FeatureLabel <- factor(fisher_results_hyper$FeatureLabel, levels = feature_order)
fisher_results_hypo$FeatureLabel  <- factor(fisher_results_hypo$FeatureLabel, levels = feature_order)

custom_colors <- c(
  "tss" = "#c00606",
  "5utrs" = "#ffdb00",
  "exons" = "#49ff00",
  "introns" = "#00ff92",
  "3utrs" = "#0092ff",
  "downstream_1kb" = "#3406ab",
  "intergenic" = "#ff8ad8"
)

ggplot(fisher_results_hyper, aes(x = FeatureLabel, y = FoldEnrichment, fill = Feature)) +
  geom_bar(stat = "identity", width = 0.8, lwd = 0.4, color = "black") +
  geom_text(aes(label = p_display), vjust = -0.5, size = 2.75) +
  coord_cartesian(ylim = c(0, 1.5)) +
  geom_hline(yintercept = 1, linetype = "dashed", color = "gray40") +
  scale_fill_manual(values = custom_colors) +
  labs(
    title = "Fold Enrichment of Hyper-DMLs\nby Genomic Feature",
    x = "Genomic Feature",
    y = "Fold Enrichment"
  ) +
  theme_classic(base_size = 11) +
```

```
  theme(
    plot.title = element_text(hjust = 0.5, colour = "#CD2626"),
    axis.text.x = element_text(angle = 45, hjust = 1),
    plot.margin = margin(t = 10, r = 10, b = 10, l = 10),
    legend.position = "none"
  )
```




```
# ggsave("analysis_results/Fold_enrichment_hyper_dmls_in_features.png", width = 5, height = 6, units =

ggplot(fisher_results_hypo, aes(x = FeatureLabel, y = FoldEnrichment, fill = Feature)) +
  geom_bar(stat = "identity", width = 0.8, lwd = 0.4, color = "black") +
  geom_text(aes(label = p_display), vjust = -0.5, size = 2.75) +
  coord_cartesian(ylim = c(0, 1.5)) +
  geom_hline(yintercept = 1, linetype = "dashed", color = "gray40") +
  scale_fill_manual(values = custom_colors) +
  labs(
    title = "Fold Enrichment of Hypo-DMLs\nby Genomic Feature",
    x = "Genomic Feature",
    y = "Fold Enrichment"
  ) +
  theme_classic(base_size = 11) +
  theme(
    plot.title = element_text(hjust = 0.5, color = "#3A5FCD"),
    axis.text.x = element_text(angle = 45, hjust = 1),
    plot.margin = margin(t = 10, r = 10, b = 10, l = 10),
    legend.position = "none"
```

```
)
```




```
# ggsave("analysis_results/Fold_enrichment_hypo_dmls_in_features.png", width = 5, height = 6, units = "
```

1.6.1 Intergenic regions

Show strong enrichment for both hyper- and hypomethylated DMLs: - Odds Ratios: Hyper = 2.05, Hypo = 1.79 - This suggests that DMLs are overrepresented in intergenic regions, relative to the CpG background.

1.6.2 Exons, Introns and 3'UTRS

Show consistent depletion of DMLs: - Exons: Hyper = 0.45, Hypo = 0.49 - Introns: Hyper = 0.58, Hypo = 0.69 - 3' UTRs: Hyper = 0.24, Hypo = 0.50 These odds ratios suggest that DMLs are less likely to occur within gene bodies and 3' regulatory elements than expected by chance.

1.6.3 Transcription Start Sites

Show mild depletion for hyper-DMLs (OR = 0.81) and a stronger depletion for hypo-DMLs (OR = 0.33), both statistically significant.

This indicates that DMLs are less likely to be found in TSS regions, especially hypomethylated ones, suggesting that TSSs are relatively protected from methylation changes.

1.6.4 5' UTRs

Show mixed results:

- Hyper-DMLs: OR = 1.00, p = 0.74

  - no significant difference from background
- Hypo-DMLs: OR = 0.43
  - significant depletion

This indicates that hypermethylation in 5' UTRs occurs at background levels, while hypomethylation is significantly underrepresented.

```
# Round Odds Ratio and Fold Enrichment to 2 decimals
fisher_results_hyper$`Odds Ratio` <- round(fisher_results_hyper$OddsRatio, 2)
fisher_results_hyper$`Fold Enrichment` <- round(fisher_results_hyper$FoldEnrichment, 2)

fisher_results_hypo$`Odds Ratio` <- round(fisher_results_hypo$OddsRatio, 2)
fisher_results_hypo$`Fold Enrichment` <- round(fisher_results_hypo$FoldEnrichment, 2)

# Update p-values again just to ensure format
fisher_results_hyper$p_display <- ifelse(
  fisher_results_hyper$p_value < 2.2e-16,
  "<2.2 × 10^-16",
  formatC(fisher_results_hyper$p_value, format = "e", digits = 2)
)

fisher_results_hypo$p_display <- ifelse(
  fisher_results_hypo$p_value < 2.2e-16,
  "<2.2 × 10^-16",
  formatC(fisher_results_hypo$p_value, format = "e", digits = 2)
)

# Build clean tables
hyper_table <- fisher_results_hyper[, c("FeatureLabel", "CpG_count", "Hyper_dmls", "p_display", "Odds Ra
colnames(hyper_table) <- c("Feature", "CpG Count", "Hyper DMLs", "P-value", "Odds Ratio", "Fold Enrichme

hypo_table <- fisher_results_hypo[, c("FeatureLabel", "CpG_count", "Hypo_dmls", "p_display", "Odds Ratio
colnames(hypo_table) <- c("Feature", "CpG Count", "Hypo DMLs", "P-value", "Odds Ratio", "Fold Enrichmen

# Define the desired order
feature_order <- c("TSS", "5' UTR", "Exons", "Introns", "3' UTR", "Downstream (1kb)", "Intergenic")

# Apply order to both tables
hyper_table$Feature <- factor(hyper_table$Feature, levels = feature_order)
hypo_table$Feature  <- factor(hypo_table$Feature,  levels = feature_order)

# Sort the data frames by the ordered factor
hyper_table <- hyper_table[order(hyper_table$Feature), ]
hypo_table  <- hypo_table[order(hypo_table$Feature), ]


# View tables
kable(hyper_table, caption = "Fisher's Test Results for Hypermethylated DMLs")
```

Table 5: Fisher's Test Results for Hypermethylated DMLs

	Feature	CpG Count	Hyper DMLs	P-value	Odds Ratio	Fold Enrichment
1	TSS	492557	13708	<2.2 × 10^-16	0.81	0.81
3	5' UTR	341338	11626	7.44e-01	1.00	1.00

	Feature	CpG Count	Hyper DMLs	P-value	Odds Ratio	Fold Enrichment
5	Exons	1096878	17683	<2.2 × 10^-16	0.45	0.47
6	Introns	11534889	282730	<2.2 × 10^-16	0.58	0.72
4	3' UTR	653184	5657	<2.2 × 10^-16	0.24	0.25
2	Downstream (1kb)	268352	2846	<2.2 × 10^-16	0.30	0.31
7	Intergenic	12758595	593117	<2.2 × 10^-16	2.05	1.36

```
kable(hypo_table, caption = "Fisher's Test Results for Hypomethylated DMLs")
```

Table 6: Fisher's Test Results for Hypomethylated DMLs

	Feature	CpG Count	Hypo DMLs	P-value	Odds Ratio	Fold Enrichment
1	TSS	492557	11190	<2.2 × 10^-16	0.33	0.35
3	5' UTR	341338	9960	<2.2 × 10^-16	0.43	0.45
5	Exons	1096878	36828	<2.2 × 10^-16	0.49	0.52
6	Introns	11534889	605257	<2.2 × 10^-16	0.69	0.81
4	3' UTR	653184	22290	<2.2 × 10^-16	0.50	0.53
2	Downstream (1kb)	268352	12471	<2.2 × 10^-16	0.70	0.72
7	Intergenic	12758595	1064491	<2.2 × 10^-16	1.79	1.29

1.7 Tests for overlaps wihtout prececence

```
options(scipen = 0) # turn off scientific notation

# Run Fisher's test for all features for HyperDMLs
fisher_results_inclusive_hyper <- data.frame()

for (i in 1:nrow(df_overlap_inclusive)) {
  # table logic
  #               In_feature    Not_in_feature      Total
  # DML               a               b           total_hyper
  # Not_DML           c               d                .
  # Total       total_in_feature      .             total_cpg

  a <- df_overlap_inclusive$HyperDMLs[i]
  total_in_feature <- df_overlap_inclusive$All_CpG_Count[i]
  c <- total_in_feature - a

  total_hyper <- sum(df_overlap_inclusive$HyperDMLs)
  total_cpg <- sum(df_overlap_inclusive$All_CpG_Count)

  b <- total_hyper - a
  d <- (total_cpg - total_in_feature) - b

  contingency <- matrix(c(a, b, c, d),
    nrow = 2,
    dimnames = list(
      Region = c("In_Feature", "Not_In_Feature"),
      Status = c("DML", "Not_DML")
```

```
    )
  )

  test <- fisher.test(contingency)

  fisher_pval <- test$p.value
  odds_ratio <- as.numeric(test$estimate)

  fisher_results_inclusive_hyper <- rbind(fisher_results_inclusive_hyper, data.frame(
    Feature = df_dml_feature_counts$Feature[i],
    p_value = fisher_pval,
    OddsRatio = odds_ratio
  ))
}

kable(fisher_results_inclusive_hyper, caption = "Fisher's test results for hypermethylated DMLs (non-ex
```

Table 7: Fisher's test results for hypermethylated DMLs (non-exclusive features)

Feature	p_value	OddsRatio
tss	0	0.8276208
downstream_1kb	0	0.3096455
5utrs	0	0.9536381
3utrs	0	0.2493399
exons	0	0.5011264
introns	0	0.6118821
intergenic	0	2.0818213

Fisher's exact tests were performed using both exclusive and inclusive genomic feature overlaps. Odds ratios were very similar across both approaches, meaning a consistent pattern of DML enrichment or depletion.

While p-values differed slightly most likely due to overlapping feature counts but the overall conclusions remain the same

1.8 Conclusions

To assess whether differentially methylated loci (DMLs) preferentially occur in specific genomic regions, I performed Fisher's exact tests comparing the distribution of hyper- and hypomethylated DMLs across annotated features relative to the background of all tested CpG sites.

The results showed extremely small p-values (0) for most genomic features, indicating highly significant differences between the observed distribution of DMLs and the background. Based on the odds ratios, DMLs were strongly enriched in intergenic regions, while significantly depleted in the majority of the rest of the regions.

These findings suggest that differential methylation is not randomly distributed throughout the genome, but tends to occur more often in certain genomic features.

Differentially methylated regions (DMRs) were identified using the same statistical thresholds and parameters as the DML analysis, i.e. the same minimum methylation difference, smoothing, and significance cut-offs.

Although permutation-based methods could be used in principle to assess the significance of DMRs, they are computationally intensive and were not applied here due to time constraints. However, since our Fisher's

exact tests demonstrated that DMLs detected under the same parameters are highly statistically significant across multiple genomic features, we can infer that the DMRs identified are also likely to be satisfactory.