Package 'ACMGuru'

February 13, 2024

Title What the Package Does (One Line, Title Case)

Version 0.0.0.9000

Description What the package does (one paragraph).

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.2

R topics documented:

`apply_acmg_pm2` *Apply ACMG PM2 Criterion*

Description

Applies the ACMG PM2 criterion based on allele frequency in population databases.

Usage

```
apply_acmg_pm2(df, gnomad_max = 1e-06)
```

Arguments

df	A dataframe containing genetic data.
gnomad_max	The maximum allele frequency considered for PM2.

Value

A modified dataframe with the ACMG PM2 criterion applied.

---

apply_acmg_pm3 *Apply ACMG PM3 Criterion*

Description

Applies the ACMG PM3 criterion for recessive disorders detected in trans with a pathogenic variant.

Usage

```
apply_acmg_pm3(df)
```

Arguments

df	A dataframe containing genetic data.

Value

A modified dataframe with the ACMG PM3 criterion applied.

---

apply_acmg_pp3 *Apply ACMG PP3 Criterion*

Description

Applies the ACMG PP3 criterion based on multiple lines of computational evidence supporting a deleterious effect on the gene or gene product. This function assesses variants for evidence of pathogenicity based on computational predictions.

Usage

```
apply_acmg_pp3(df)
```

Arguments

df	A dataframe containing genetic data with necessary predictive columns.

Value

A modified dataframe with the ACMG PP3 criterion applied.

apply_acmg_ps1 *Apply ACMG PS1 Criterion*

Description

This function applies the ACMG PS1 criterion, identifying pathogenic variants based on clinical significance.

Usage

```
apply_acmg_ps1(df)
```

Arguments

df	A dataframe containing genetic data with a CLIN_SIG column.

Value

A modified dataframe with the ACMG PS1 criterion applied.

apply_acmg_ps5 *Apply ACMG PS5 Criterion*

Description

Applies the ACMG PS5 criterion based on strong pathogenic evidence.

Usage

```
apply_acmg_ps5(df)
```

Arguments

df	A dataframe containing genetic data.

Value

A modified dataframe with the ACMG PS5 criterion applied.

apply_acmg_pvs1 *Apply ACMG PVS1 Criterion*

Description

This function applies the ACMG PVS1 criterion to a given dataframe.

Usage

```
apply_acmg_pvs1(df)
```

Arguments

df	A dataframe containing genetic data.

Value

A modified dataframe with the ACMG PVS1 criterion applied.

apply_column_classes_to_processed_data *Apply Column Classes to Processed Data*

Description

Applies the column classes to each dataframe in a list based on a provided reference metadata CSV file.

Usage

```
apply_column_classes_to_processed_data(
  processed_data_list,
  metadataclass_file_path = NULL
)
```

Arguments

processed_data_list	A list of dataframes to be adjusted.
metadataclass_file_path	Optional. Path to the CSV file containing reference metadata for column classes. If not provided, uses the default file included in the package.

compare_column_classes_and_output_csv

Compare Column Classes and Output CSV

Description

Compares column classes across all dataframes in a list and outputs a CSV file with the most common class for each column where inconsistencies are found.

Usage

```
compare_column_classes_and_output_csv(
  processed_data_list,
  metadataclass_file_path = NULL
)
```

Arguments

processed_data_list	A list of dataframes to be analyzed.
metadataclass_file_path	Optional. Path where the output CSV file will be saved. If not provided, uses the default file included in the package.

filter_by_allele_frequency

Filter Data Based on Allele Frequency

Description

This function filters variants in the dataset based on a specified allele frequency threshold.

Usage

```
filter_by_allele_frequency(df, af_threshold)
```

Arguments

df	A dataframe containing genetic data, expected to have an `AF.x` column representing allele frequency.
af_threshold	A numeric value specifying the maximum allele frequency for included variants.

Value

A filtered dataframe with variants below the specified allele frequency threshold.

Examples

```
df <- data.frame(
  AF.x = c(0.01, 0.05, 0.2)
)
result <- filter_by_allele_frequency(df, 0.1)
```

plot_criteria_count_each_gene *Plotting Summaries for ACMG Criteria*

Description

Contains functions for generating various plots to summarize the counts and distributions of ACMG criteria across genes and variants.

Usage

```
plot_criteria_count_each_gene(df, file_suffix)
```

prepare_acmg_labels *Prepare ACMG Labels*

Description

Prepares ACMG labels by ensuring all expected ACMG label columns exist in the dataframe, assigning NA where they do not, identifying the highest priority ACMG classification for each row, and counting the number of non-NA ACMG labels for each row.

Usage

```
prepare_acmg_labels(df)
```

Arguments

df	A dataframe containing genetic data potentially including various ACMG label columns.

Value

A modified dataframe with additional columns for the highest priority ACMG classification (`ACMG_highest`) and the count of non-NA ACMG labels (`ACMG_count`) for each row.

Examples

```
df <- data.frame(
  ACMG_PVS1 = c(NA, "PVS1"),
  ACMG_PS1 = c("PS1", NA)
)
df_prepared <- prepare_acmg_labels(df)
```

process_genetic_data *Process Genetic Data According to ACMG Guidelines*

Description

This function processes genetic data by applying a series of ACMG criteria to identify variants based on specified thresholds and conditions.

Usage

```
process_genetic_data(input_path, samples_file_path, af_threshold)
```

Arguments

input_path	Path to the input files or directory containing the CSV files.
samples_file_path	Path to the samples file containing phenotype information.
af_threshold	Allele frequency threshold for filtering variants.

Value

A list of data frames, each representing processed genetic data for a file.

Examples

```
input_path <- system.file("extdata", package = "YourPackageName")
samples_file_path <- system.file("extdata", "samples.tsv", package = "YourPackageName")
af_threshold <- 0.01
processed_data_list <- process_genetic_data(input_path, samples_file_path, af_threshold)
```

read_data_file *Read Data Files and Merge with Sample Phenotype Information*

Description

This function reads data files from a specified path and merges them with sample phenotype information. It can handle a single file, all files within a directory, or a specific list of files.

Usage

```
read_data_file(input_path = NULL, samples_file_path, file_list = NULL)
```

Arguments

input_path	Optional. Path to the input directory containing the files or a single file path. Used only if file_list is NULL.
samples_file_path	Path to the samples file containing phenotype information.
file_list	Optional. A vector of specific file paths to be processed. Overrides input_path if provided.

Value

A list of data frames, each representing a merged data file.

`set_comp_het_flag` *Set Compound Heterozygous Flags*

Description

This function sets flags for compound heterozygous variants within the dataset. WARNING: This function does not check phase.

Usage

```
set_comp_het_flag(df)
```

Arguments

`df`	A dataframe containing genetic data with columns `sample`, `SYMBOL`, and `genotype`.

Value

A modified dataframe with a new `comp_het_flag` column.

Examples

```
df <- data.frame(
  sample = c("Sample1", "Sample1"),
  SYMBOL = c("Gene1", "Gene1"),
  genotype = c(1, 2)
)
result <- set_comp_het_flag(df)
```

Index