MyVariant.info R Client

Adam Mark

June 1, 2015

Contents

1 Overview

MyVariant.Info is a simple-to-use REST web service to query/retrieve genetic variant annotation from an aggregation of variant annotation resources. *myvariant* is an easy-to-use R wrapper to access MyVariant.Info services and explore variant annotions.

2 Variant Annotation Service

2.1 Obtaining HGVS IDs from a VCF file.

- Use `getVcf` to read a VCF file into a data.frame. The vcf object can then be passed to `getSnps`, `getIns`, `getDels`, or `getAll` to extract the HGVS IDs for SNPs, deletions, insertions, or all of the above. Keep in mind the vcf object is simply created for the purpose of extracting HGVS IDs. There are no exported functions in the `myvariant` package that operate on the vcf object

other than the functions to extract the IDs. HGVS IDs are based on the GRCh38/hg19 reference genome. Support for hg38 is coming soon.

```
> file.path <- system.file("extdata", "dbsnp_mini.vcf", package="myvariant")
> vcf <- getVcf(file.path)
> head(vcf[c('CHROM', 'POS', 'rsID', 'REF', 'ALT')])
```

```
  CHROM   POS        rsID REF ALT
1  chr1 10019 rs376643643  TA   T
2  chr1 10055 rs373328635   T  TA
3  chr1 10108  rs62651026   C   T
4  chr1 10109 rs376007522   A   T
5  chr1 10139 rs368469931   A   T
6  chr1 10144 rs144773400  TA   T
```

- You can then use getSnps to extract HGVS IDs from the vcf object. The IDs will be appended as the 'query' column.

```
> snps <- getSnps(vcf)
> head(snps[c('query', 'type', 'pos')])
```

```
             query type              pos
3  chr1:g.10108C>T  snp chr1:10108-10108
4  chr1:g.10109A>T  snp chr1:10109-10109
5  chr1:g.10139A>T  snp chr1:10139-10139
8  chr1:g.10150C>T  snp chr1:10150-10150
9  chr1:g.10177A>C  snp chr1:10177-10177
11 chr1:g.10180T>C  snp chr1:10180-10180
```

```
> hgvs <- snps$query
```

2.2 getVariant

- Use getVariant, the wrapper for GET query of "/v1/variant/<hgvsid>" service, to return the variant object for the given HGVS id.

```
> variant <- getVariant("chr1:g.35367G>A")
> variant[[1]]$dbnsfp$genename
```

```
[1] "FAM138A"
```

```
> variant[[1]]$cadd$phred
```

```
[1] 1.493
```

2.3 getVariants

- Use getVariants, the wrapper for POST query of "/v1/variant" service, to return the list of variant objects for the given character vector of HGVS ids.

```
> getVariants(c("chr1:g.881627G>A", "chr1:g.887560A>C", "chr1:g.888639T>C",
+                "chr12:g.31477822G>A", "chr3:g.56771251A>C", "chr8:g.62416074G>A"),
+             fields="cadd.consequence")
```

```
DataFrame with 6 rows and 3 columns
                  _id               query cadd.consequence
          <character>         <character>      <character>
1    chr1:g.881627G>A    chr1:g.881627G>A       SYNONYMOUS
2    chr1:g.887560A>C    chr1:g.887560A>C               NA
3    chr1:g.888639T>C    chr1:g.888639T>C               NA
4 chr12:g.31477822G>A chr12:g.31477822G>A      STOP_GAINED
5  chr3:g.56771251A>C  chr3:g.56771251A>C   NON_SYNONYMOUS
6  chr8:g.62416074G>A  chr8:g.62416074G>A      SPLICE_SITE
```

3 Variant Query Service

3.1 queryVariant

- queryVariant is a wrapper for GET query of "/v1/query?q=<query>" service, to return the query result. This functions accepts wild card input terms and allows you to query for variants the contain a specific annotation. For example, the following query searches for the CADD phred score and consequence for all variants whose genename (dbNSFP) is MLL2.

```
> queryVariant(q="dbnsfp.genename:MLL2", fields=c("cadd.phred", "cadd.consequence"))
```

```
$hits
                   _id   _score
1  chr12:g.49418460A>T 9.948488
2  chr12:g.49418658C>A 9.948488
3  chr12:g.49420281G>T 9.948488
4  chr12:g.49420340T>A 9.948488
5  chr12:g.49420364T>A 9.948488
6  chr12:g.49420844C>A 9.948488
7  chr12:g.49420888G>C 9.948488
8  chr12:g.49421602A>T 9.948488
9  chr12:g.49421807C>A 9.948488
10 chr12:g.49422661T>A 9.948488

$max_score
[1] 9.948488

$took
[1] 25
```

```
$total
[1] 37953
```

- You can also use queryVariant to retrieve all annotations that map to a specific rsID.

```
> queryVariant(q="rs58991260", fields="dbsnp.flags")$hits
```

```
                   _id   _score                                          flags
1 chr1:g.218631822G>A 17.48191 ASP, G5, G5A, GNO, KGPhase1, KGPhase3, SLO
```

3.2 queryVariants

- queryVariants is a wrapper for POST query of "/v1/query?q=<query>" service, to return the query result. Query terms include any available field as long as scopes are defined. The following example reads the dbSNP rsIDs from a VCF and queries for all fields. The returned DataFrame can then be easily subsetted to include, for example, those that have not been documented in the Wellderly study.

```
> rsids <- vcf$rsID
> res <- queryVariants(q=rsids, scopes="dbsnp.rsid", fields="all")
```

```
Finished
Pass returnall=TRUE to return lists of duplicate or missing query terms.
```

```
> subset(res, !is.na(wellderly.vartype))$query
```

```
 [1] "rs367896724" "rs145427775" "rs147093981" "rs56289060"  "rs55998931"  "rs199606420"
 [7] "rs199606420" "rs58108140"  "rs62635284"  "rs62635286"  "rs200579949" "rs531730856"
[13] "rs180734498" "rs527952245" "rs201696125" "rs370886505" "rs546169444" "rs201055865"
[19] "rs369473859" "rs62635298"  "rs79585140"  "rs75454623"  "rs199856693" "rs201855936"
[25] "rs71252251"  "rs201045431" "rs71252250"  "rs200030104" "rs78601809"  "rs62636497"
[31] "rs62636497"  "rs62636497"  "rs62636497"  "rs201635489" "rs533630043" "rs2691315"
[37] "rs113442401" "rs572465511" "rs372319358" "rs200358166" "rs11489794"  "rs113141985"
[43] "rs62636498"  "rs148220436" "rs141130360" "rs150723783" "rs201459529" "rs199676946"
[49] "rs62636367"  "rs62636368"  "rs200205172" "rs199745162" "rs200658479" "rs201833382"
[55] "rs199740902" "rs200978805" "rs201535981" "rs192890528" "rs200046632" "rs374995955"
```

4 References

MyVariant.info