Create Gene Coordinate Table from CCDS table

Load data downloaded from FTP site: ftp://ftp.ncbi.nih.gov/pub/CCDS

Check if the first entry is the longest

You can also embed plots, for example:

```
check_largest <- function(x){if(x[1]==max(x)) {return(1)} else {return(0)}}
check_res <- group_by(public,gene) %>%
  summarise(check = check_largest(cds_to))
length(which(check_res$check != 1)) ## not always first
```

```
## [1] 916
```

```
#uniq <- group_by(public,gene) %>% summarise_all(funs(first))
```

```
uniq <- group_by(public,gene) %>%
  summarise(cds_start = min(cds_from),cds_end = max(cds_to),
            `#chromosome`=paste(unique(`#chromosome`),collapse='_'))
unique(uniq$`#chromosome`) ## there is "X_Y"
```

```
##  [1] "19"  "10"  "12"  "1"   "22"  "3"   "4"   "9"   "15"  "2"   "11"
## [12] "17"  "20"  "8"   "16"  "6"   "7"   "X"   "13"  "14"  "21"  "18"
## [23] "5"   "X_Y" "Y"
```

```
filter(uniq,`#chromosome`=='X_Y')
```

```
## # A tibble: 18 x 4
##       gene cds_start   cds_end `#chromosome`
##      <chr>     <dbl>     <dbl>         <chr>
##  1 AKAP17A   1593462   1601593           X_Y
##  2    ASMT   1615199   1643013           X_Y
##  3   ASMTL   1403268   1452839           X_Y
##  4    CD99   2691360   2740803           X_Y
##  5   CRLF2   1190896   1212633           X_Y
##  6  CSF2RA   1282703   1309588           X_Y
##  7   DHRSX   2221040   2500924           X_Y
##  8  GTPBP6    305073    318786           X_Y
##  9   IL3RA   1341765   1382464           X_Y
## 10    IL9R  57184263 156010408           X_Y
## 11   P2RY8   1465478   1466557           X_Y
## 12  PLCXD1    284187    299334           X_Y
## 13 PPP2R3B    334366    386690           X_Y
## 14    SHOX    630897    658828           X_Y
## 15 SLC25A6   1386601   1392008           X_Y
## 16   SPRY3  56960391 155774737           X_Y
## 17   VAMP7  57075986 155942138           X_Y
## 18   ZBED1   2488634   2490718           X_Y
```

```
XYgenes <- filter(uniq,`#chromosome`=='X_Y') $gene
```

Apparently, some genes are only on X, but entry is wrong; and some on both X and Y.

```
## apparently some genes are only on X, but entry is wrong
filter(public,gene=='ZBED1')
```

```
## # A tibble: 2 x 11
##   `#chromosome` nc_accession  gene gene_id     ccds_id ccds_status
##           <chr>        <chr> <chr>   <int>       <chr>       <chr>
## 1             X NC_000023.11 ZBED1    9189 CCDS14118.1      Public
## 2             Y NC_000024.10 ZBED1    9189 CCDS14118.1      Public
## # ... with 5 more variables: cds_strand <chr>, cds_from <int>,
## #   cds_to <int>, cds_locations <chr>, match_type <chr>
```

```
## and some on both X and Y.
filter(public,gene=='SPRY3')
```

```
## # A tibble: 2 x 11
##   `#chromosome` nc_accession  gene gene_id     ccds_id ccds_status
##           <chr>        <chr> <chr>   <int>       <chr>       <chr>
## 1             X NC_000023.11 SPRY3   10251 CCDS14769.4      Public
## 2             Y NC_000024.10 SPRY3   10251 CCDS14769.4      Public
## # ... with 5 more variables: cds_strand <chr>, cds_from <int>,
## #   cds_to <int>, cds_locations <chr>, match_type <chr>
```

For 3 genes, need to create entry on both X and Y.

```
uniq <- group_by(public,gene,`#chromosome`) %>%
  summarise(cds_start = min(cds_from),cds_end = max(cds_to))
```

Manually remove the genes which are using X coordinates but have chromosome Y entry. Then write table.

```
retain <- c('IL9R','SPRY3','VAMP7')
XYgenes <- XYgenes [!XYgenes %in% retain]
uniq <- filter(uniq, !(gene %in% XYgenes & `#chromosome` == 'Y'))

final <- uniq[,c('gene','#chromosome','cds_start','cds_end')]
head(final)
```

```
## # A tibble: 6 x 4
## # Groups:   gene [6]
##      gene `#chromosome` cds_start  cds_end
##     <chr>         <chr>     <dbl>    <dbl>
## 1    A1BG            19  58347021 58353436
## 2    A1CF            10  50806728 50859939
## 3     A2M            12   9067822  9115848
## 4   A2ML1            12   8822651  8875010
## 5 A3GALT2             1  33306765 33321097
## 6  A4GALT            22  42692889 42693950
```

```
#write.table(final,'table.tsv',quote = F, sep='\t',row.names = F)
```