Example pipeline

1. Run STAR (build genome first)

```
STAR --genomeDir /path/to/star/genome --runThreadN 4 --readFilesIn
paired_reads_1.fastq paired_reads_2.fastq --outSAMtype BAM SortedByCoordinate --
outWigType wiggle --outWigNorm None --outSAMunmapped Within --outFileNamePrefix
aligned_reads
```

This command will generate the following files:

```
# All reads in the form of a BAM file #
* aligned_reads_STARAligned.sortedByCoord.out.bam
```

Contains both mapped and unmapped reads. Unmapped reads need to be extracted as part of the find_circ pipeline to detect non-canonical backsplicing

```
# Uniquely mapped reads only #
* aligned_reads_STARSignal.Unique.str1.out.wig  # (-) strand coverage
* aligned_reads_STARSignal.Unique.str2.out.wig  # (+) strand coverage

# Multi-mapped and uniquely mapped reads #
* aligned_reads_STARSignal.UniqueMultiple.str1.out.wig  # (-) strand coverage
* aligned_reads_STARSignal.UniqueMultiple.str2.out.wig  # (+) strand coverage

# Splice junction counts #
* aligned_reads_STARSJ_out.tab  # Canonical splice junction counts file needs to
                                # be converted to bed format appropriate for
                                # circleVis processing
```

2. Run star_sj_convert on canonical splice junction file

```
star_sj_convert aligned_reads_STARSJ_out.tab
```

This command will generate the following file:

```
* aligned_reads_STARSJ_out.tab.canonical.bed   # Final canonical splice junction
                                               # file
```

3. Run find_circ to generate backsplice junction calls

```
samtools view -hf 4 aligned_reads_STARAligned.sortedByCoord.out.bam | samtools
view -Sb - | ./unmapped2anchors.py aligned_unmapped_reads.bam >
aligned_unmapped_reads_anchors.fastq
```

```
mkdir -p circles_out
```

```
STAR --genomeDir /path/to/genome --readFilesIn aligned_unmapped_reads_anchors.fastq
|find_circ.py --genome=/path/to/fasta --prefix=canonical --name=my_sample --
stats=circles_out/my_sample_stats.txt > circles_out/my_sample_splice_sites.bed
```

4. Run find_circ_convert on find_circ junction file

```
find_circ_convert my_sample_splice_sites.bed
```

This command will generate the following file:

```
* my_sample_splice_sites.bed.circles.bed  # Final backsplice junction file
```

```
# Uniquely mapped coverage will be used for this example, but it is up to the user
to choose unique or unique + multi #
```

5. Using a GTF file with the same chromosome names as the genome used for your alignement (i.e. if chromosome 1 is labeled 'chr1' in the STAR genome, it must also be labeled 'chr1' in the GTF file you use. It cannot be 'chr1' in one file and '1' in the other.), run circbuild.

```
circbuild --gtf Homo_sapiens.GRCh38.92.gtf --wigneg
aligned_reads_STARSignal.Unique.str1.out.wig  --wigpos
aligned_reads_STARSignal.Unique.str2.out.wig --splicejunction
aligned_reads_STARSJ_out.tab.canonical.bed  --circlejunction
my_sample_splice_sites.bed.circles.bed --output my_sample_name
```

This command will generate the following file:

* my_sample_name.db

6. Run circplot using your gene of interest

```
circplot --database my_sample_name.db --gene EGFR
```

This command will generate the following files:

#Open either one of them with any web browser

* EGFR_ENST00000275493.svg
* EGFR_ENST00000275493.html