Simulating pathways, mutual exclusivity, et al.

Cristea, Kuipers, and Beerenwinkel (2017) mutal exclusivity and modules

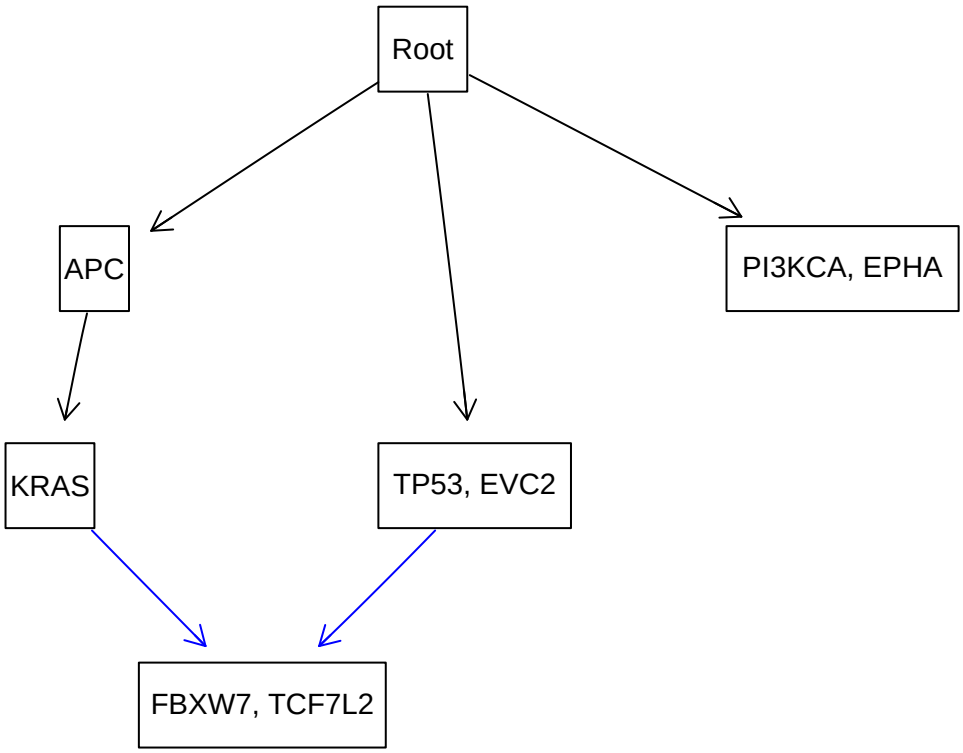
In Cristea, Kuipers, and Beerenwinkel (2017), the authors introduce a generative probabilistic graphical model of cancer progression called *pathTiMEx*. It is both, a waiting time model for independent mutually exclusive pathways, and a waiting time model for cancer progression among single genes.

From *cross-sectional* data derived from different type of tumors, authors are able to generate a cancer progression model including mutual exclusivity between groups, progression among pathways and modules of genes. The colorectal cancer model inferred depicted in Figure 3.A is used as an example of model to map.

The colorectal cancer dataset used to built that model is obtained from Wood et al. (2007). The poset restrictions proposed can be coded using the **OncoSimulR package**, concretely, the `allFitnessEffects` function. Model will be represented as an Diaciclic Direct Graph (DAG).

```
# First, it is necessary to load OncoSimulR package
library(OncoSimulR)

# Restriction table (extended version of the poset)
colcancer <- data.frame(
                 parent = c(rep("Root",3), "A", "B", "C"), # Parent nodes
                 child = c("A", "B", "D", "C", "E", "E"), # Child nodes
                 s = c(rep(0.5, 3), rep(0.05, 3)),

                 sh = -0.5,

                 typeDep = c(rep("MN", 4), rep("SM", 2)) # Type of dependency
                  )

# Fitness specification of the poset
colcancer_efec <- allFitnessEffects(
                  colcancer, # Poset

                  geneToModule = c( # Specification of the modules
                               "Root" = "Root",
                               "A" = "APC",
                               "B" = "TP53, EVC2",
                               "C" = "KRAS",
                               "D" = "PI3KCA, EPHA",
                               "E" = "FBXW7, TCF7L2"),

                  drvNames = c( # Specification of drivers
                               "APC", "TP53", "EVC2", "KRAS",
                               "PI3KCA", "EPHA", "FBXW7", "TCF7L2")
                  )
```

```
# DAG representation
plot(colcancer_efec, expandModules = TRUE, autofit = TRUE)
```

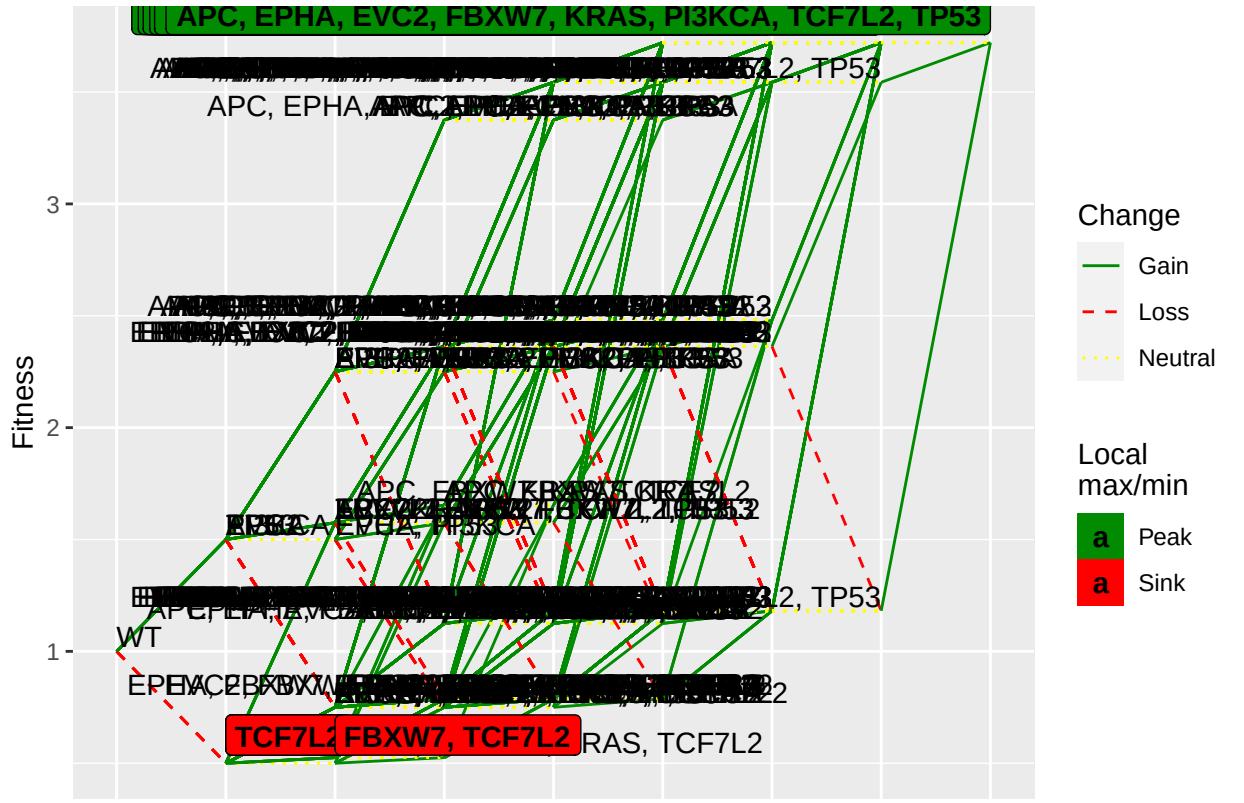


Parameter `s` refers to a numeric vector with the fitness effect that applies if the relationship is satisfied. On the other hand, parameter `sh` refers to a numeric vector with the fitness effect that applies if the relationship is not satisfied. Authors don't specify its value. However, the model proposed include a waiting time rate parameter λ, and a mutual exclusivity intensity parameter μ. λ could give us an idea about the value we should give to each module. On the other hand, `sh` is given a constant value for all possible situations.

Although authors don't specify the sort of dependency, in this poset, a semimonotonic dependency is defined between modules B and C with E, while a monotonic dependency is defined for the others.

The `evalAllGenotypes` function returns a table with all the genotypes from a fitnessEffects description. This table can be used to plot a fitness landscape of the model.

```
colcancer_efec_FL <- evalAllGenotypes(colcancer_efec, max = 110000)
# Output is not shown due to size of the table.

# Plot of fitness landscape
plotFitnessLandscape(colcancer_efec_FL)
```



The fitness landscape is quite busy with a lot of genotypes overlapping. However, it shows lables refering to a specific genotype and its corresponding fitness. As the number of genes mutated increases, the fitness does it as well.

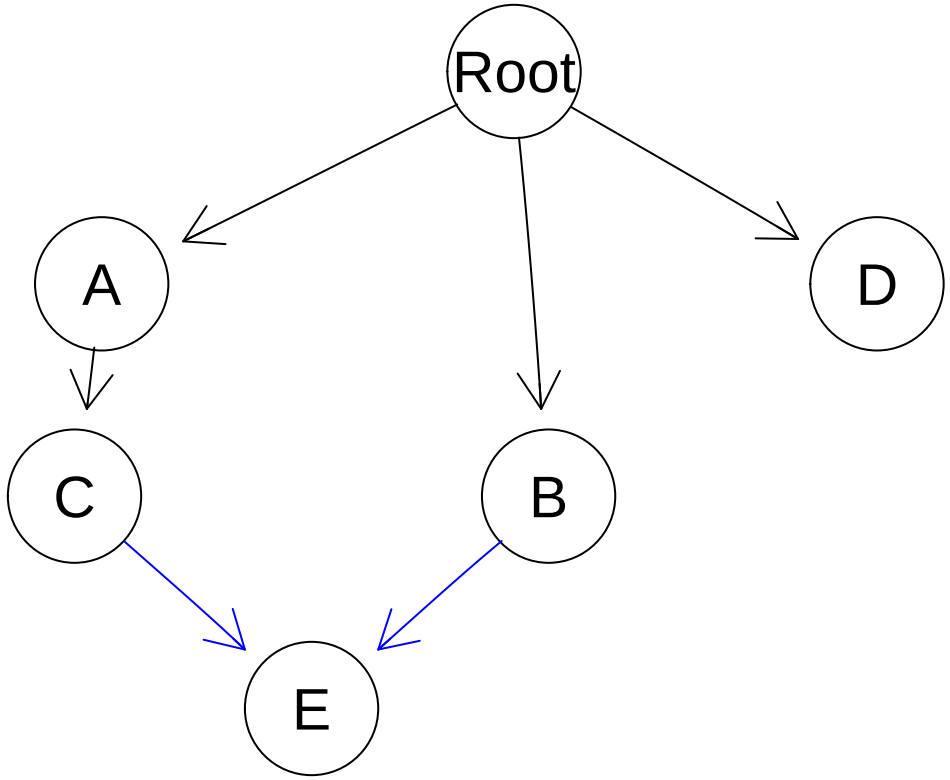
Simplification of Cristea, Kuipers, and Beerenwinkel (2017)

In order to properly visualize a fitness landscape, a simplified model of Cristea, Kuipers, and Beerenwinkel (2017) is constructed. This model doesn't use modules, just individual genes. This approach will lead to clear fitness landscape and to properly identify processes taking place.

Authors say that there is a phenomena of mutual exclusivity between certain genes of a specific pathway. Mutual exclusivity means that the presence of one gene of a specific pathway will be enough to fitness contribution, since the presence of the other gene will not have a huge impact. However, there is a specific case of mutual exclusivity called `synthetic lethality`. In this case, the presence of both genes will lead to a decrease in fitness.

Therefore, the same model as in the previous case will be coded, but without specifying modules. Each module will be consider as an specific gene.

```
# Fitness specification of the simplified poset
Scolcancer <- allFitnessEffects(colcancer)

plot(Scolcancer)
```

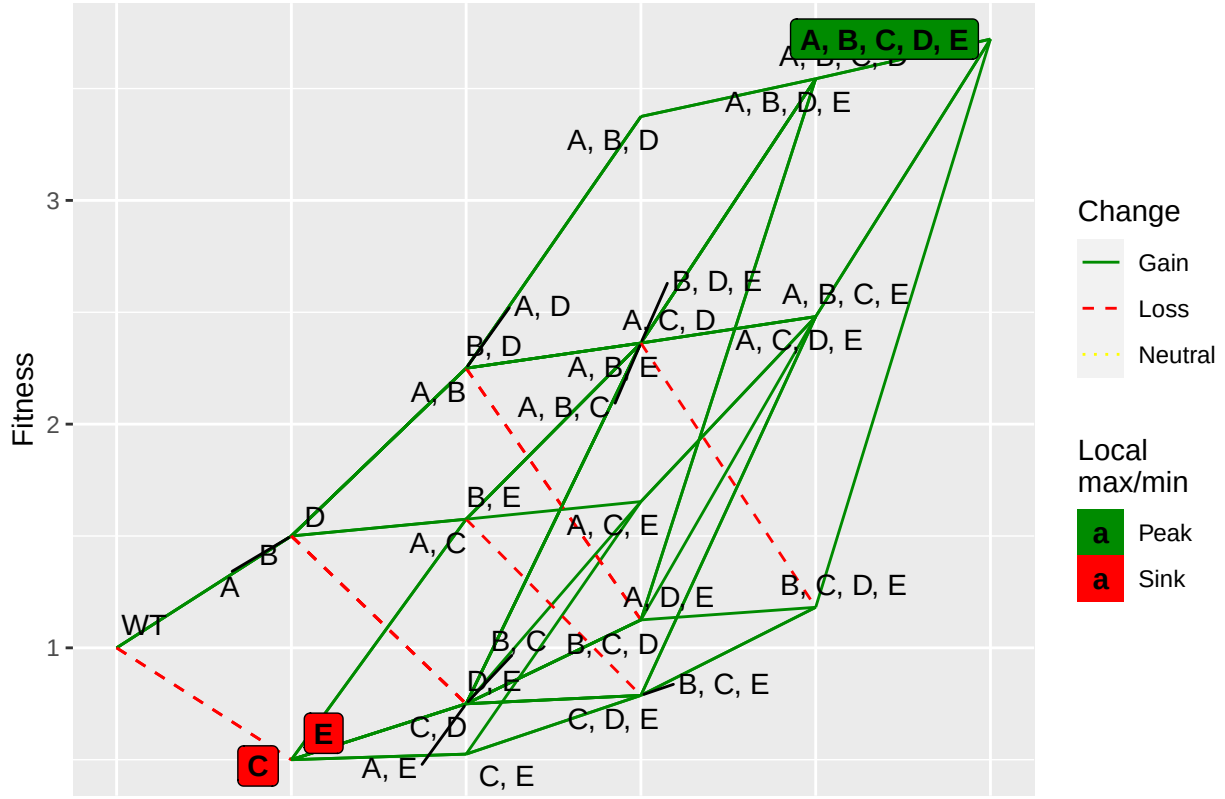



```
# Obtain all genotypes from the fitnessEffect
(Scolcancer_ge <- evalAllGenotypes(Scolcancer))
```

```
##         Genotype  Fitness
## 1              A 1.500000
## 2              B 1.500000
## 3              C 0.500000
## 4              D 1.500000
## 5              E 0.500000
## 6           A, B 2.250000
## 7           A, C 1.575000
## 8           A, D 2.250000
## 9           A, E 0.750000
## 10          B, C 0.750000
## 11          B, D 2.250000
## 12          B, E 1.575000
## 13          C, D 0.750000
## 14          C, E 0.525000
## 15          D, E 0.750000
## 16       A, B, C 2.362500
## 17       A, B, D 3.375000
## 18       A, B, E 2.362500
## 19       A, C, D 2.362500
## 20       A, C, E 1.653750
## 21       A, D, E 1.125000
## 22       B, C, D 1.125000
```

```
## 23       B, C, E 0.787500
## 24       B, D, E 2.362500
## 25       C, D, E 0.787500
## 26    A, B, C, D 3.543750
## 27    A, B, C, E 2.480625
## 28    A, B, D, E 3.543750
## 29    A, C, D, E 2.480625
## 30    B, C, D, E 1.181250
## 31 A, B, C, D, E 3.720938
```

```
# Plot the fitness landscape.
plotFitnessLandscape(Scolcancer_ge,
                     use_ggrepel = TRUE)
```



Order effects

To explore what does really mean order effects, we are going to create a simple model derived from the restriction model inferred by Cristea, Kuipers, and Beerenwinkel (2017).

This simplified model contains 3 genes: `APC`, `TP53` and `KRAS`. Genes considered as `superdrivers` by Gerstung et al. (2011), meaning that are the main responsible for cancer progression since they provide a higher fitness gain than the other genes in the model. This conclusion is obtained from its article where they used the same colorectal cancer dataset as Cristea, Kuipers, and Beerenwinkel (2017). Thus, it can be extrapolated to our case.

The relationships between those genes was previously depicted in section 1. In this case, we will set `APC` as the parent of `KRAS`. Both, `APC` and `TP53` have as parent `Root`. Based on the waiting time rate parameter λ, the fitness values of each possible order is given.

λ is higher for `APC`, which means that it seems to appear before in the cancer progression. λ for `KRAS` is the lower between the three, meaning that it mutates the last. `TP53` mutation occurs between `APC` and `KRAS`.

Order effect is visualize using `evalAllGenotypes` function.

```
cc <- data.frame(parent = c(rep("Root", 2), "A"),
                 child = c("A", "C", "B"),
                 typeDep = "MN")

cc_order <- allFitnessEffects(
  orderEffects = c("A > B > C" = 0.5, "B > A > C" = 0.2,
                   "B > C > A" = 0.1,
                   "B > C" = 0.2,
                   "C > B" = 0.1,
                   "B > A" = 0.1,
                   "A > B" = 0.3),

  geneToModule =
    c("A" = "APC",
      "B" = "KRAS",
      "C" = "TP53") )

(cc_order_geno <- evalAllGenotypes(cc_order, order = TRUE))
```

```
##             Genotype Fitness
## 1                APC   1.000
## 2               KRAS   1.000
## 3               TP53   1.000
## 4          APC > KRAS   1.300
## 5          APC > TP53   1.000
## 6          KRAS > APC   1.100
## 7         KRAS > TP53   1.200
## 8          TP53 > APC   1.000
## 9         TP53 > KRAS   1.100
## 10 APC > KRAS > TP53   2.340
## 11 APC > TP53 > KRAS   1.430
## 12 KRAS > APC > TP53   1.584
## 13 KRAS > TP53 > APC   1.452
## 14 TP53 > APC > KRAS   1.430
## 15 TP53 > KRAS > APC   1.210
```

We obtain a table with the different possible genotypes as well as the order of appearance. However, this approach doesn't allow to generate neither a DAG neither a fitness landscape. Thus, is not possible to visualize the evolution of the genotypes with time.

```
#DAG
plot(cc_order)
```

```
## Error in `*tmp*`[[i]]: subíndice fuera de  los límites
```

```r
# Fitness landscape
plotFitnessLandscape(cc_order_geno)
```

```
## Error in to_Fitness_Matrix(x, max_num_genotypes = max_num_genotypes): We cannot deal with order effe
```

Assuming a model where there is not an order effect, a mutation in gene "B" followed by a mutation in gene "A" will reach the same fitness as if the mutation in gene "A" occurs first. However, in the model just generated, the order of the mutation impacts the final fitness reached by the tumoral clone. Since, mutation some genes before can lead to an evolutionary advantage.

In a non order effect model, the final fitness value is the same for all the clones, while this is unlikely to happen in real life. Clones carrying a certain mutation from the beginning would survive easily than those reaching the same genotype in a different order.

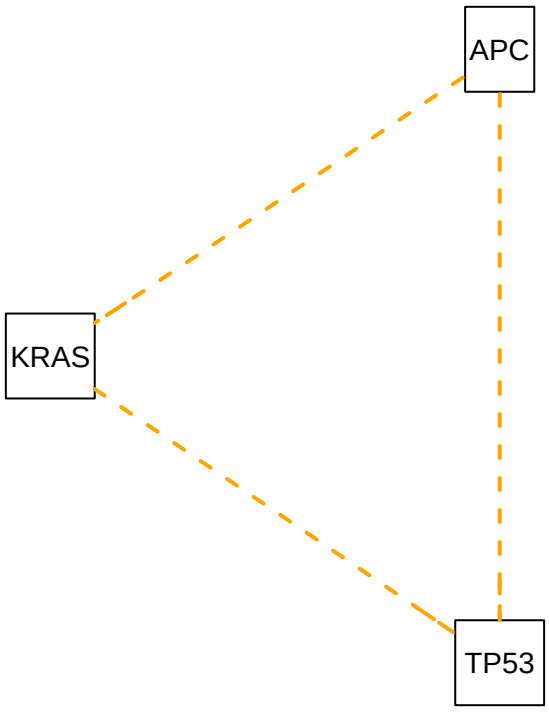
This is one limitation of `OncoSimulR package`, it doesn't allow to visualize those scenarios (yet).

Epistasis

Epistasis assume that there is a dependence between genotypes. The effect of a mutation depends on the genetic background in which it happens (Poelwijk et al. 2007). Now, we will cope with dependences between genes using epistasis.

For that, we will use the same model described in section (Order effects). As explained before, it is supposed to be a certain cancer progression restriction and therefore, the fitness values given to each different genotype is based in that criteria.

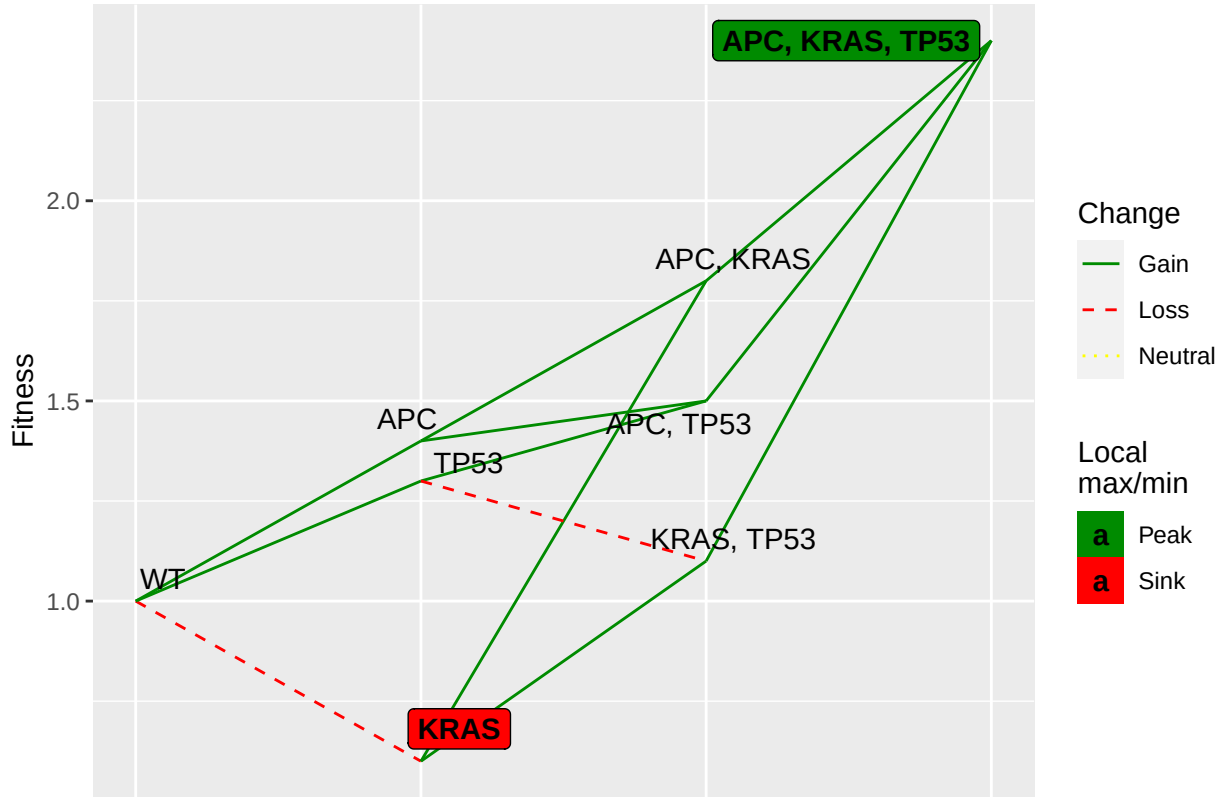
```r
# Fitness object defined using epistasias
cc_epi <- allFitnessEffects(epistasis =
                              c("A: -B: -C" = 0.4,
                                "-A: B: -C" = -0.4,
                                "-A: -B: C" = 0.3,
                                "A: B: -C" = 0.8,
                                "A : B: C" = 1.4,
                                "-A: B: C" = 0.1,
                                "A : -B: C" = 0.5
                                ),
                            geneToModule =
                              c("A" = "APC",
                                "B" = "KRAS",
                                "C" = "TP53")
                              )

# DAG (epistasis)
plot(cc_epi, expandModules = TRUE, autofit = TRUE)
```




```
# Genotypes derived from fitness defined with epistasia relationships
(cc_epi_geno <- evalAllGenotypes(cc_epi ))
```

```
##          Genotype Fitness
## 1             APC     1.4
## 2            KRAS     0.6
## 3            TP53     1.3
## 4       APC, KRAS     1.8
## 5       APC, TP53     1.5
## 6      KRAS, TP53     1.1
## 7 APC, KRAS, TP53     2.4
```

```
## Fitness landscape from this relationships
plotFitnessLandscape(cc_epi_geno, use_ggrepel = TRUE)
```



Using this approach, it is possible to visualize the DAG. In this case, there are discontinues yellow lines connecting each gene. This lines indicate a dependence between them. Fitness landscape is also plotted.

With this model, we promote the clones of tumoral cells beginning with a first mutation in `APC`. Other clones not starting with that mutation (`KRAS` or `TP53`) have a lower fitness value. On the other hand, all genotypes end with the same fitness.

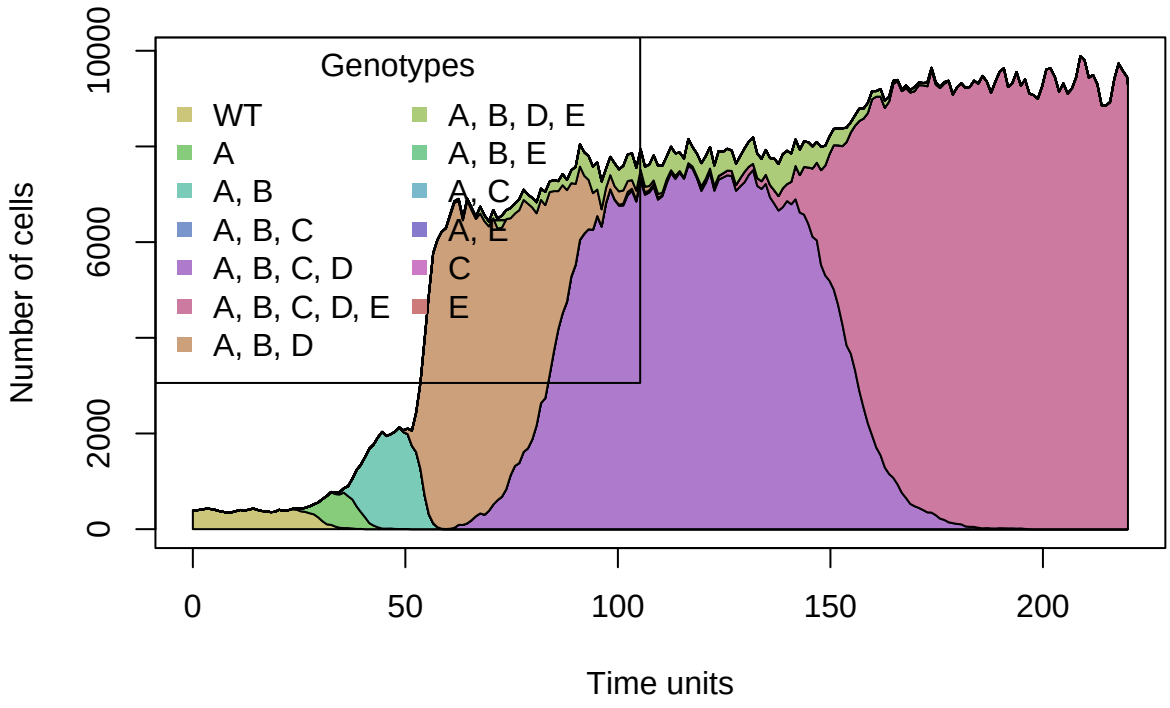
Simulating data from simplified model of Cristea, Kuipers, and Beerenwinkel (2017)

Restrictions set in DAG were used as a guide line to built the fitness landscape. This fitness landscape shows each possible genotype as well as its fitness. This landscape can be used to simulate the cancer progression.

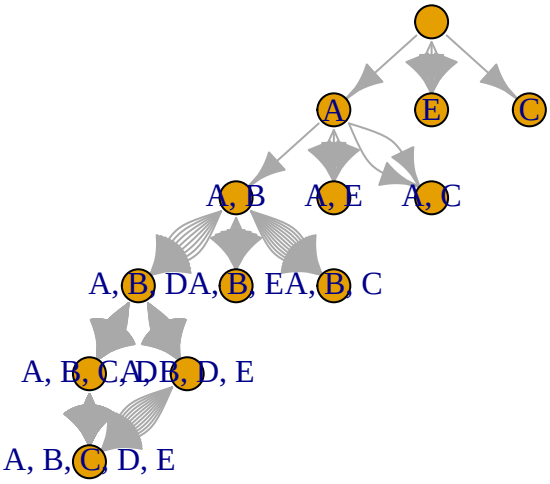
`OncoSimulIndiv` is used to simulate colorectal tumor progression.

```
set.seed(257) # Fix the seed for reproducibility

Simul <- oncoSimulIndiv(Scolcancer, ## A fitnessEffects object
                        model = "McFL", ## Model used
                        mu = 1e-4, ## Mutation rate
                        sampleEvery = 0.02, ## How often the whole population is sampled
                        keepEvery = 1,
                        initSize = 400, ## Initial population time
                        finalTime = 220,
                        keepPhylog = TRUE, ## Allow to see parent-child relationships
                        onlyCancer = FALSE
)
```

```
# Plot of simulation
plot(Simul, ## OncoSimulIndv model
     show = "genotypes",
     type = "stacked"

)
```




```
# Parent-child relationship derived from simulation
plotClonePhylog(Simul,
                #fixOverlap = TRUE,
                N = 0, ## Specify clones that exist
                keepEvents = TRUE ## Arrows showing how many times each clones appeared


)
```



References

Cristea, Simona, Jack Kuipers, and Niko Beerenwinkel. 2017. "PathTiMEx: Joint Inference of Mutually Exclusive Cancer Pathways and Their Progression Dynamics." *Journal of Computational Biology* 24 (6): 603–15. https://doi.org/10.1089/cmb.2016.0171.

Gerstung, Moritz, Nicholas Eriksson, Jimmy Lin, Bert Vogelstein, and Niko Beerenwinkel. 2011. "The temporal order of genetic and pathway alterations in tumorigenesis." *PLoS ONE* 6 (10). https://doi.org/10.1371/journal.pone.0027136.

Poelwijk, Frank J., Daniel J. Kiviet, Daniel M. Weinreich, and Sander J. Tans. 2007. "Empirical fitness landscapes reveal accessible evolutionary paths." *Nature* 445 (7126): 383–86. https://doi.org/10.1038/nature05451.

Wood, Laura D., D. Williams Parsons, Siân Jones, Jimmy Lin, Tobias Sjöblom, Rebecca J. Leary, Dong Shen, et al. 2007. "The genomic landscapes of human breast and colorectal cancers." *Science*. https://doi.org/10.1126/science.1145720.