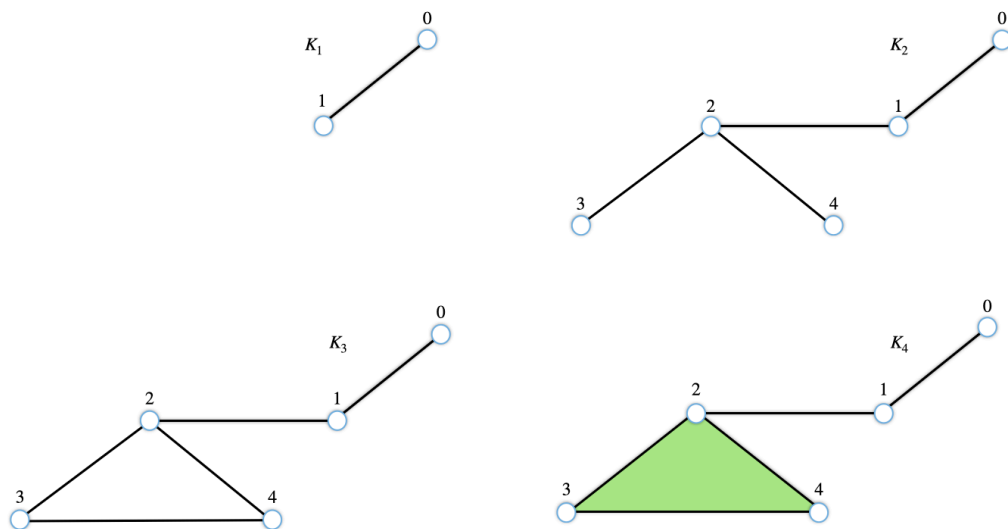


1 Filtración de complejos simpliciales



```
[73]: sc=SimplicialComplex([])
      sc.insert([(0,1)],1.0)
      sc.insert([(1,2),(2,3),(2,4)],2.0)
      sc.insert([(3,4)],3.0)
      sc.insert([(2,3,4)],4.0)
```

Devolvemos el conjunto de todas las caras del complejo simplicial

```
[74]: sc.face_set
```

```
[74]: {(0,),
      (0, 1),
      (1,),
      (1, 2),
      (2,),
      (2, 3),
      (2, 3, 4),
      (2, 4),
      (3,),
      (3, 4),
      (4,)}
```

El siguiente comando devuelve el parámetro (umbral) para el que aparece un símlice. Vemos cuando aparece el vértice 3.

```
[75]: sc.threshold((3,))
```

```
[75]: 2.0
```

Guardamos por separado cada uno de los complejos simpliciales de la filtración

```
[78]: K1=sc.filtration(1.0)
      K2=sc.filtration(2.0)
      K3=sc.filtration(3.0)
      K4=sc.filtration(4.0)
```

Calculamos el conjunto de caras de cada uno de ellos para ver que coinciden con los del dibujo

```
[80]: K1.face_set
```

```
[80]: {(0,), (0, 1), (1,)}
```

```
[81]: K2.face_set
```

```
[81]: {(0,), (0, 1), (1,), (1, 2), (2,), (2, 3), (2, 4), (3,), (4,)}
```

```
[82]: K3.face_set
```

```
[82]: {(0,), (0, 1), (1,), (1, 2), (2,), (2, 3), (2, 4), (3,), (3, 4), (4,)}
```

```
[83]: K4.face_set
```

```
[83]: {(0,),
      (0, 1),
      (1,),
      (1, 2),
      (2,),
      (2, 3),
      (2, 3, 4),
      (2, 4),
      (3,),
      (3, 4),
      (4,)}
```

Devolvemos la lista de todos los símlices del complejo simplicial ordenados según la filtración.

```
[84]: sc.filtrationorder
```

```
[84]: [(0,),
      (1,),
      (0, 1),
      (2,),
      (3,),
      (4,),
      (1, 2),
```

$(2, 3),$
 $(2, 4),$
 $(3, 4),$
 $(2, 3, 4)]$