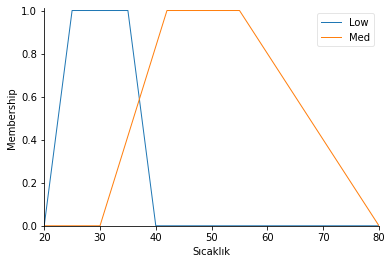
**Yarıyıl Sonu Puanı İçin Ödevler**

**Not:** Ödevler verildiği haftanın bir sonraki hafta derste bilgisayar üzerinde kontrol edilecektir. Ayrıca yazmış olduğunuz kodlarla ilgili iki adet soru sorulacaktır. Değerlendirme kriterleri aşağıdaki gibidir?

1. İstenen çıktıyı veren ve sorulara cevap veren öğrencilerin ödevleri 100 puan olarak değerlendirilir.
2. İstenen çıktıyı tam vermeyip sorulan sorulara cevap veren öğrencilerin ödevleri 50 puan olarak değerlendirilecektir.
3. Sorulan sorulara cevap verilmesi esastır. Ödevi istenen çıktıyı veren ancak kodla ilgili iki soruya cevap veremeyen öğrencilerin cevap veremediği soru başına 50 puan düşülecektir. İki soruya da cevap veremeyen öğrencilerin ödevleri 0 puandır.

**Ödev-5**

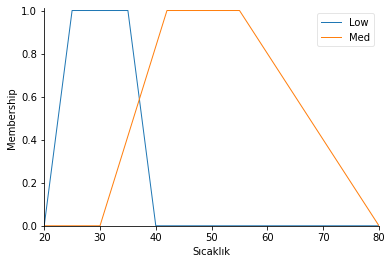
* + **Girişler: Sıcaklık**
  + **Evrensel Küme: np.arange(20,81,1)**
  + **Low:[20,25,35,40]**
  + **Medium:[30,42,55,80]**



Yukarıdaki bilgilere göre Low ve Med bulanık kümelerinin Birleşimini ve kesişimini tüm t-norm ve s-norm yöntemlerine göre hesaplayıp görselleştiren Python programını yazın.? Her öğrenci kendisi yapacaktır.

**Ödev-6**

* Çıkış: Sıcaklık
* Evrensel Küme: np.arange(20,81,1)
* Low:[20,25,35,40]
* Medium:[30,42,55,80]
* Kural 1: Öncül çıkışı:0.4 (Low->22)
  + - Çıkış Kümesi: Low olacak şekilde mamdani’ye göre çıkış kümesini elde edip görselleştirin.
* Kural 2: Öncül Çıkışı:0.75 (Med->39)
  + - Çıkış Kümesi:Med olacak şekilde mamdaniye göre çıkış kümesini elde edip görselleştirin.
    - Yukarıdaki benzer işlemleri Larsen’e göre de gerçekleştirip ayrı bir görsel olarak çizdirin.

****

**Not: Tüm ödevler bir sonraki hafta derste kontrol edilecektir. Ayrıca ödev cevabı ile ilgili bazı sorular sorulacaktır. Ödevi yapmakla birlikte sorulara da cevap vermek önem arz etmektedir.**

**Ödev 7**

Github’daki Ornek-1.py’deki kodları optimize edin. Tekrarlayan işlemler için fonksiyon yazarak kodları kısaltın?

**Ödev 8**

* Giriş Değişkeni: **OXI(Oksijen)**
* 3 kümeden oluşur:
* low\_flow: points: [0, 1.], [1., 1.], [1.5, 0]
* medium\_flow: points:[0.5, 0], [1.5, 1.], [2.5, 1], [3., 0]
* high\_flow: points=[[2., 0], [2.5, 1.], [3., 1.]
* Çıkış Değişkeni: POWER
* LOW\_POWER için k=0) #m(0.51)=1.0
* MEDIUM\_POWER için k=25 #m(0.51)=0.01
* HIGH\_FUN için k=OXI\*\*2 #m(0.51)=0

Kural1 = OXI==low\_flow İSE POWER=LOW\_POWER

Kural2 = OXI==medium\_flow İSE POWER=MEDIUM\_POWER

Kural3 = OXI!=low\_flow İSE POWER=HIGH\_FUN

OXI=0.51 için POWER değerini Sugeno yöntemine göre hesaplayın? Python kodunu yazın? Elde edile sonuçları karşılaştırın?