FİNAL PROJESİ

K MEANS CLUSTRING
HIERARCHICAL CLUSTRING

Feyza Nur SAKA 1521221051
Bilgisayar Mühendisliği

Dr. Öğr. Üyesi Berna KİRAZ

Tıbbi Maliyet Kişisel Veri Kümeleri

(Medical Cost Personal Datasets)

- Bu veri kümesi 1338 satırdan oluşur

Kolonlar

1. **age:** Birincil yararlanıcı yaşı
2. **sex:** Cinsiyeti, kadın, erkek
3. **bmi:** Vücudun anlaşılmasını sağlayan vücut kitle indeksi
4. **children:** Sağlık sigortası kapsamındaki çocuk sayısı / bakmakla yükümlü olunanların sayısı
5. **smoker:** Yes, No
6. **region:** kuzeydoğu, güneydoğu, güneybatı, kuzeybatı (northeast, southeast, southwest, northwest)
7. **charges:** Sağlık sigortası tarafından faturalandırılan bireysel tıbbi masraflar

K MEANS CLUSTRING

- Unsupervised learning ve kümeleme algoritmasıdır.
- K-Means' teki **K değeri küme sayısını belirler** ve bu değeri parametre olarak alması gerekir.
- K değeri belirlendikten sonra algoritmada **rastgele K tane merkez noktası seçer.**
- Her **veri** ile rastgele belirlenen **merkez noktaları** arasındaki **uzaklığı** hesaplayarak veriyi **en yakın merkez** noktasına göre bir kümeye atar.
- Daha sonra **her küme için yeniden bir merkez noktası seçilir** ve **yeni merkez noktalarına göre kümeleme işlemi yapılır.**
- Bu durum sistem **kararlı** hale gelene kadar devam eder.

Algoritmada Kullanılan Kütüphaneler :

```
import pandas as pd
import matplotlib.pyplot as plt
```

Dosyadan veri okumak için pandas
Veri görselleştirmek için matplotlib

```
from sklearn import preprocessing
```

Preprocessing işlemi için sklearn

```
from sklearn.cluster import KMeans
```

Kümeleme yapmak için sklearn kütüphanesinin KMeans sınıfı

WCSS Metriği :

(Within Clusters Sum of Square)

$$WCSS = \sum_{i \in n} (X_i - Y_i)^2$$

Kümeler içi kareler toplamı

Birbirine benzeyenlerin, yakın olanların aynı kümede olmasını birbirine benzemeyenlerin uzak olmasını sağlayan bir metriktir.

Algoritmada Kullanılan Parametreler :

```
KMeans(n_clusters=k, init ='k-means++', max_iter=300, n_init=10,random_state=0 )
```

n_clusters : Oluşturulacak küme sayısı ve üretilecek sentroid sayısı.

init : İlk küme merkezlerini belirler

max_iter : Maksimum yineleme sayısı.

n_init : Küme merkezi başlangıç noktasının kaç farklı noktadan başlayabileceğini belirler.

random_state : Sonuçları tekrarlanabilir hale getirir ve hata ayıklama için yararlıdır.

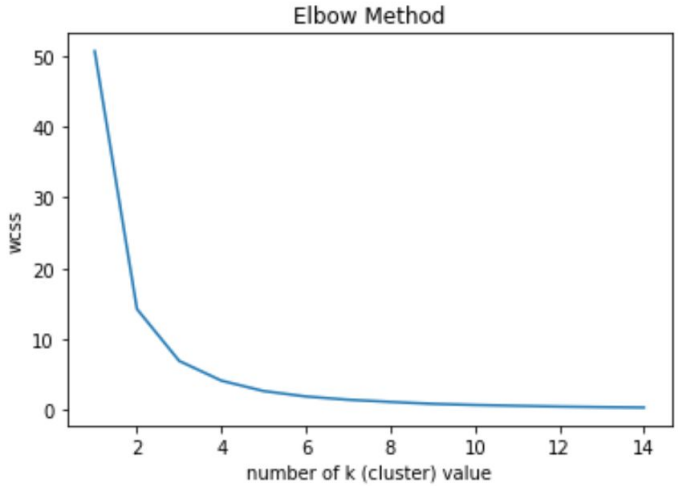
DENEYLER VE SONUÇLAR

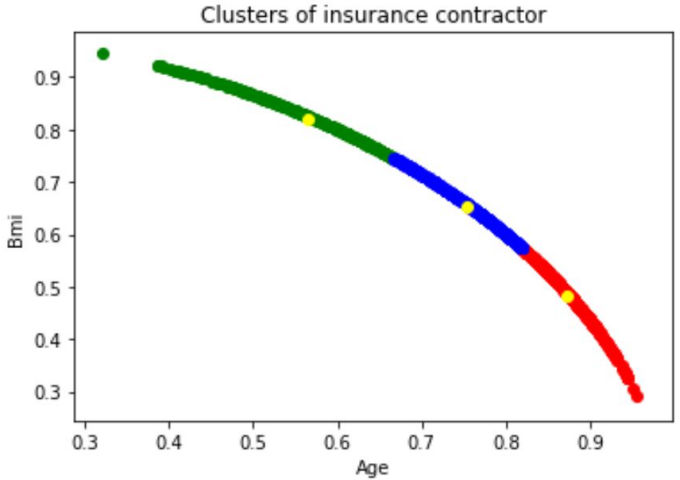
1) Age ve bmi kolonları

Parametreler default değerde iken :

```
k in range(1,15):
```

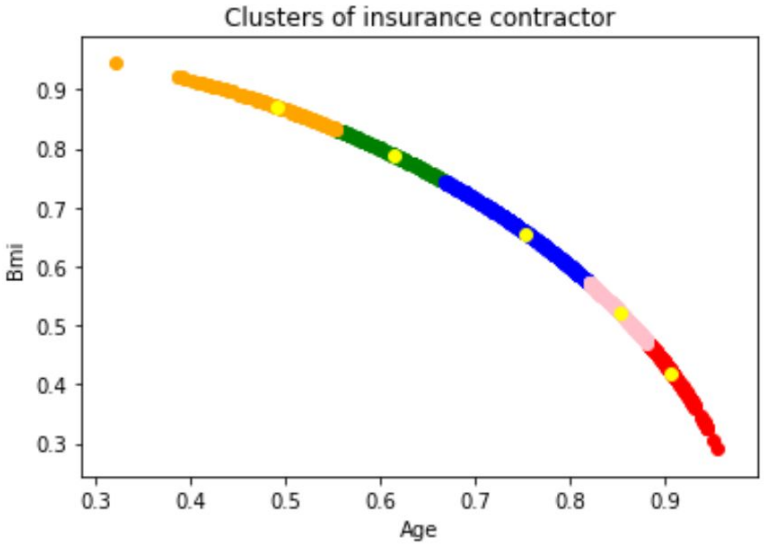
```
KMeans(n_clusters=k, init ='k-means++', max_iter=300, n_init=10,random_state=0 )
```





k = 5

Diğer parametreler sabitken k' yı arttırdığımda :

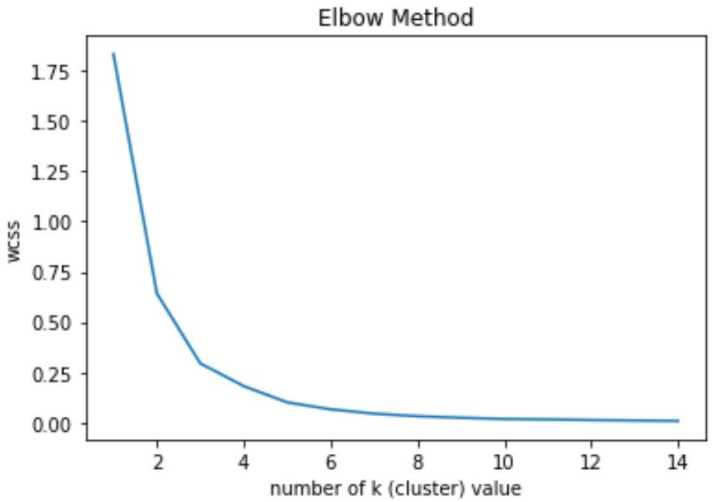


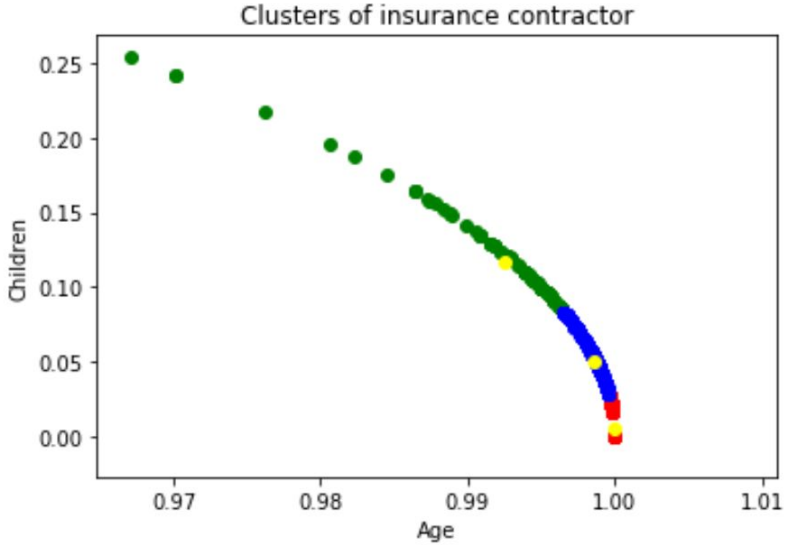
Diğer parametrelerde değişiklik yaptığımda grafik üzerinde bir fark göremedim.

2) Age ve children kolonları

Parametreler default değerde iken :

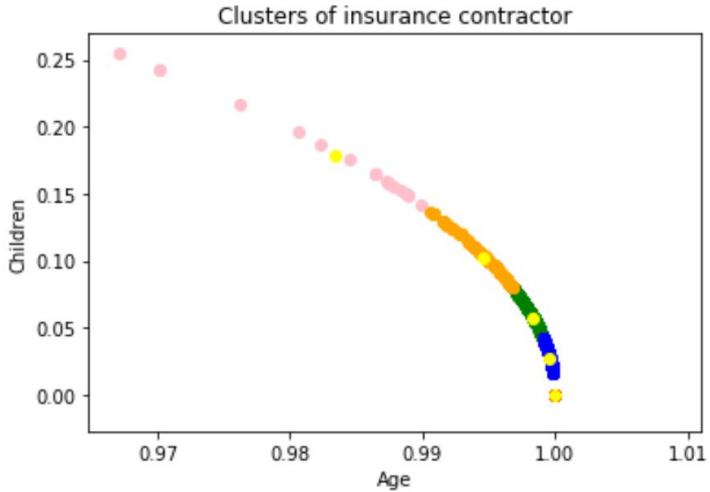
```
KMeans(n_clusters=3, init ='k-means++', max_iter=10, n_init=10,random_state=0 )
```





k = 5

Diğer parametreler sabitken k' yı arttırdığımda :

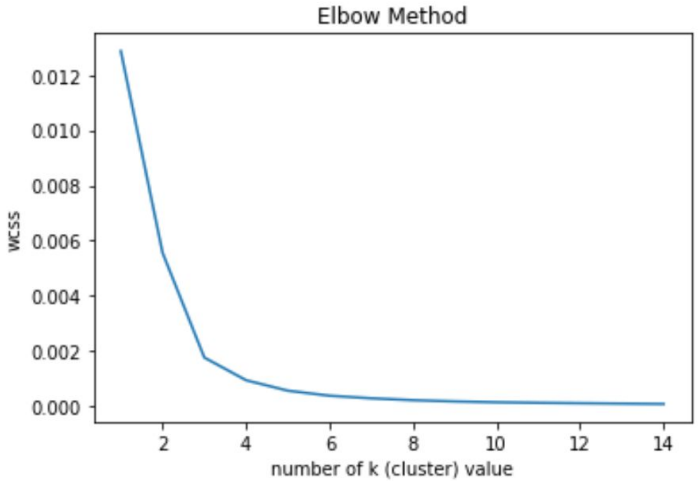


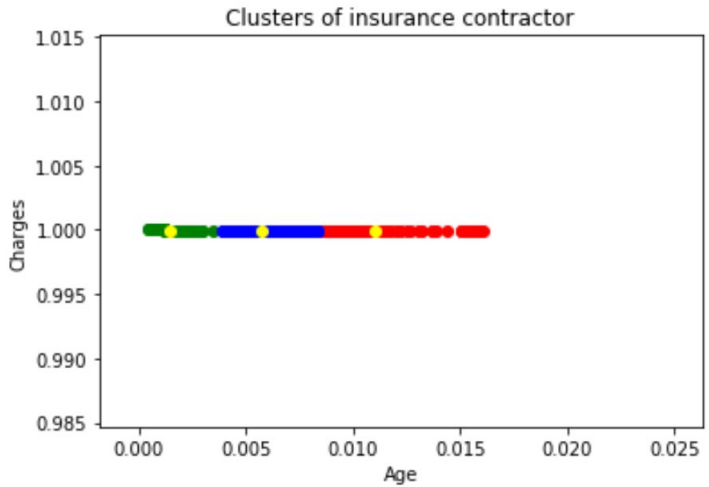
Diğer parametrelerde değişiklik yaptığımda grafik üzerinde bir fark göremedim.

3) Age ve charges kolonları

Parametreler default değerde iken :

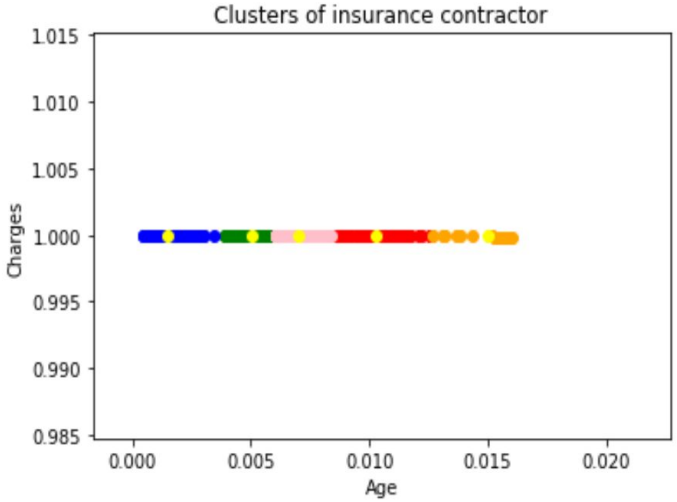
```
KMeans(n_clusters=3, init ='k-means++', max_iter=10, n_init=10,random_state=0 )
```





k = 5

Diğer parametreler sabitken k' yı arttırdığımda :



Diğer parametrelerde değişiklik yaptığımda grafik üzerinde bir fark göremedim.

HIERARCHICAL CLUSTRING

- Unsupervised learning ve kümeleme algoritmasıdır.
- Agglomerative (Parçadan bütüne) ve Divisive (Bütünden parçaya) olarak iki farklı varyasyonu vardır.
- Agglomerative (Yığınsal) hiyerarşik kümelemede mesafe hesaplamak için bir çok yol vardır. Dendrogram oluşturmada da kullanılırlar.

1. **Single Linkage :** İki küme arasındaki **en yakın** mesafeyi hesaplar.
2. **Complete Linkage:** İki küme arasındaki **en uzak** mesafeyi hesaplar.
3. **Average Linkage:** İki küme arasındaki **ortalama** mesafeyi hesaplar.

- Bunların dışında **ward**, **weighted**, **centroid** ve **median** yöntemleri vardır. **Seçilen yöntem sonucu etkiler.**

Dendrogram (Öbek Ağacı)

- Dendrogram, benzer veri kümeleri arasındaki ilişkileri veya hiyerarşik kümelenmeyi gösteren bir ağaç diyagramıdır.
- Kaç tane küme oluşturacağımız bilgisini verir.
- **En uzun bacaktan çizilen yatay çizgi küme sayısını verir.**

Algoritmada Kullanılan Kütüphaneler :

```
import pandas as pd
import matplotlib.pyplot as plt
```

Dosyadan veri okumak için pandas
Veri görselleştirmek için matplotlib

```
from sklearn import preprocessing
```

Preprocessing işlemi için sklearn

```
from scipy.cluster.hierarchy
import linkage, dendrogram
```

Dendrogram grafiği için

```
from sklearn.cluster
import AgglomerativeClustering
```

Kümeleme yapmak için
sklearn kütüphanesinin
AgglomerativeClustering sınıfı

Algoritmada Kullanılan Parametreler :

- **n_clusters** = Ayıracağımız küme sayısı
- **linkage ve affinity** = Mesafe ölçüm yöntemleri

```
AgglomerativeClustering(n_clusters = 3,affinity= "euclidean",linkage = "ward")
```

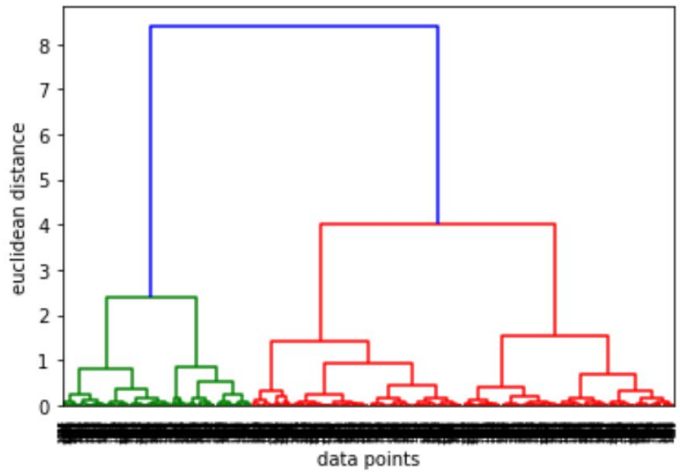
DENEYLER VE SONUÇLAR

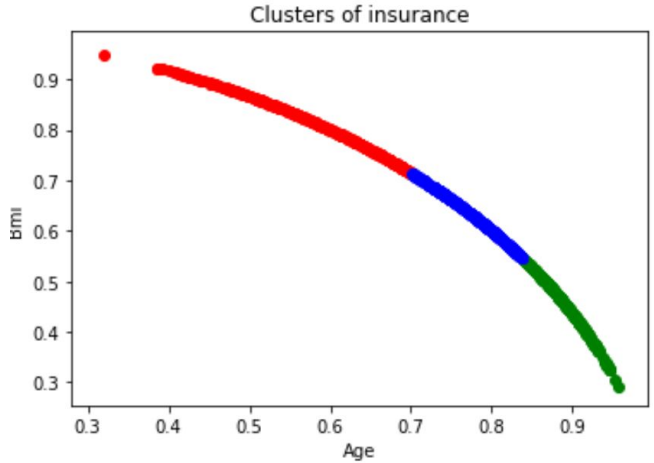
1) Age ve bmi kolonları

Parametreler default değerde iken :

- linkage = ward sadece affinity = euclidean ile kullanılır

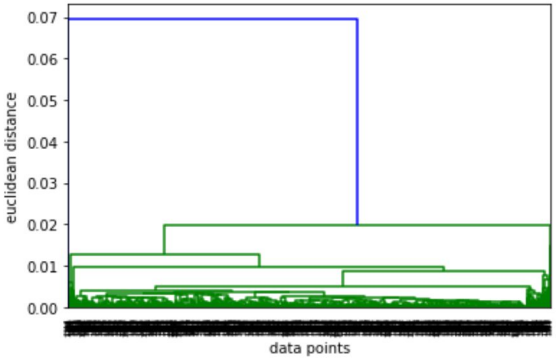
```
AgglomerativeClustering(n_clusters = 3,affinity= "euclidean",linkage = "ward")
```



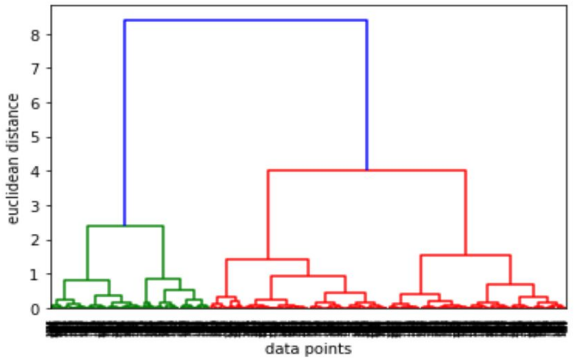


Dendrogram Çizimi

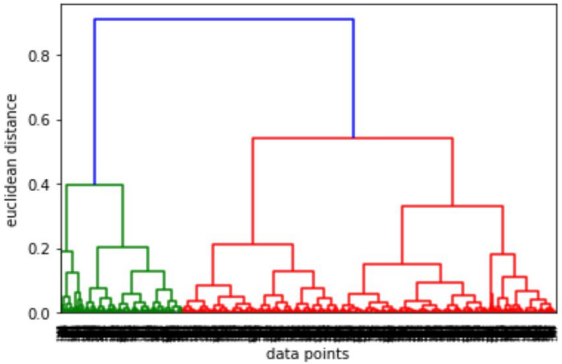
method : single



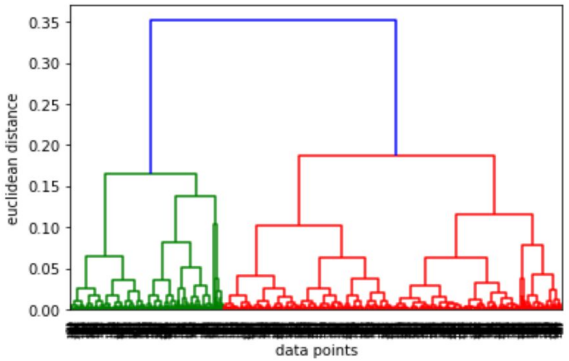
method : ward



method : complete

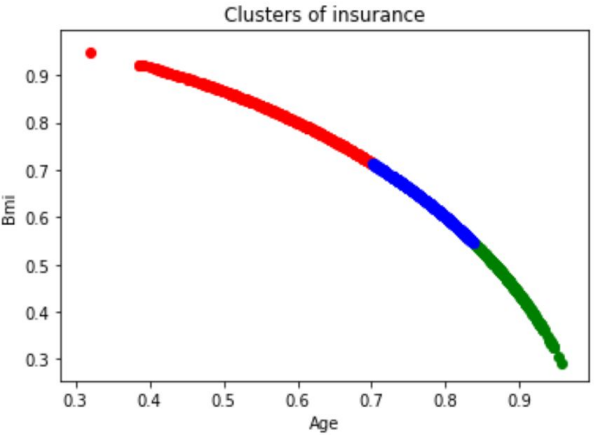


method : average



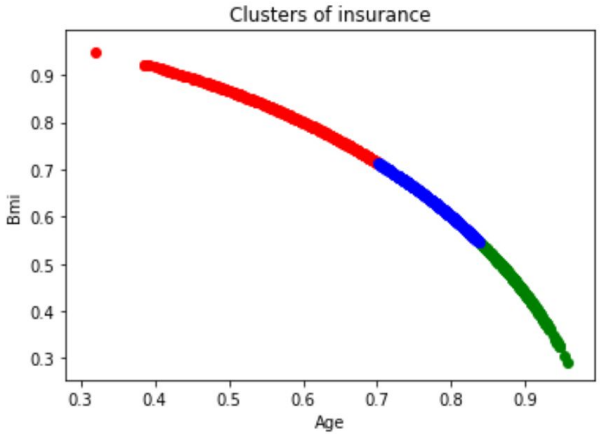
linkage = complete

affinity = manhattan



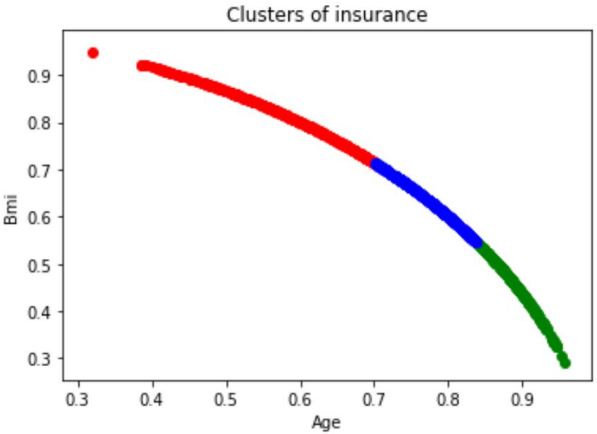
linkage = complete

affinity = euclidean



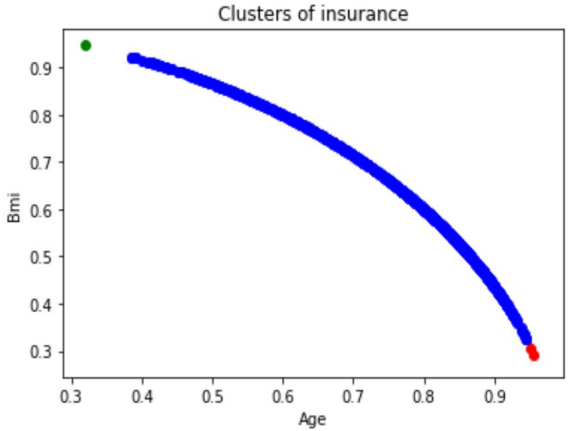
linkage = complete

affinity = cosine



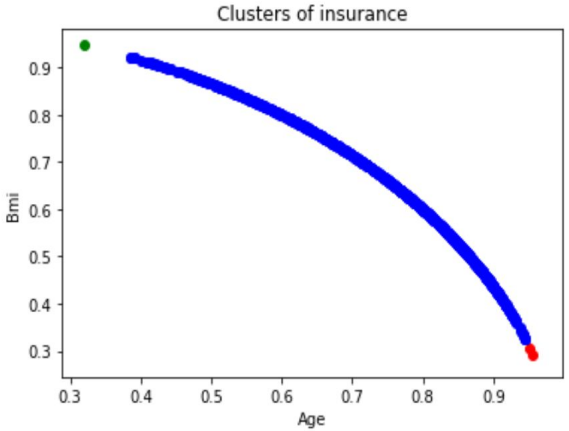
linkage = single

affinity = manhattan



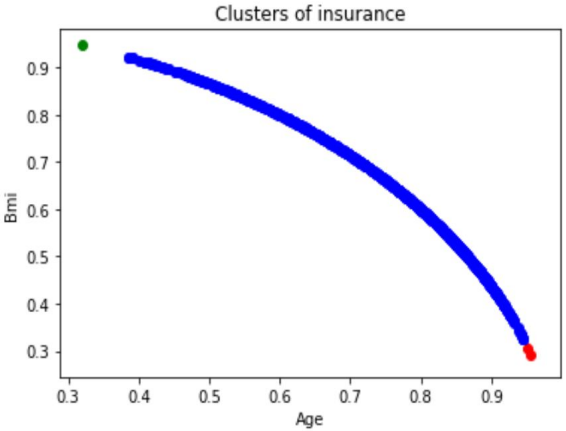
linkage = single

affinity = euclidean



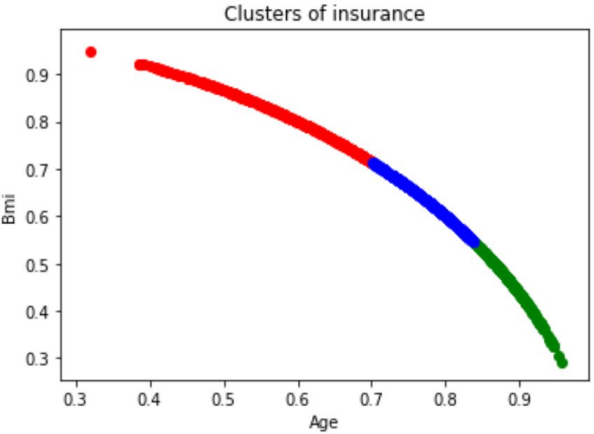
linkage = single

affinity = cosine



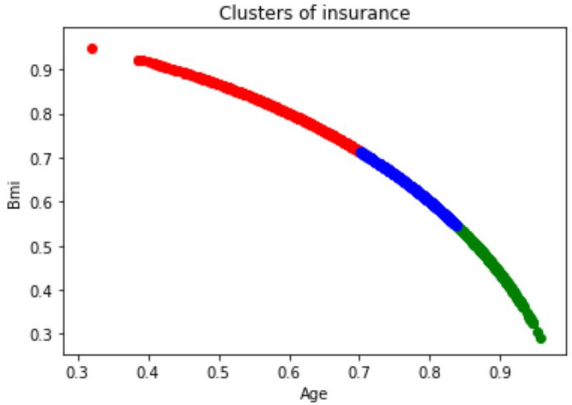
linkage = avarage

affinity = manhattan



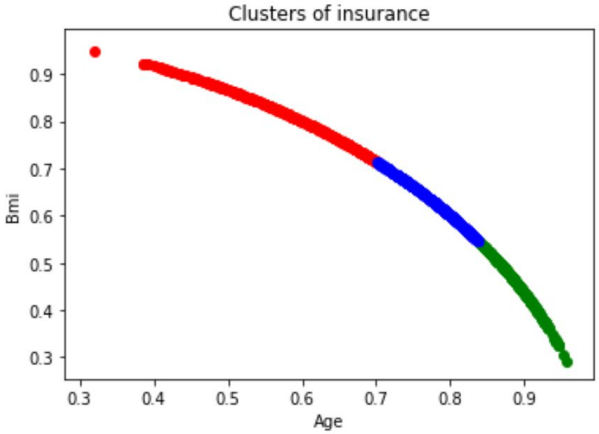
linkage = avarage

affinity = euclidean



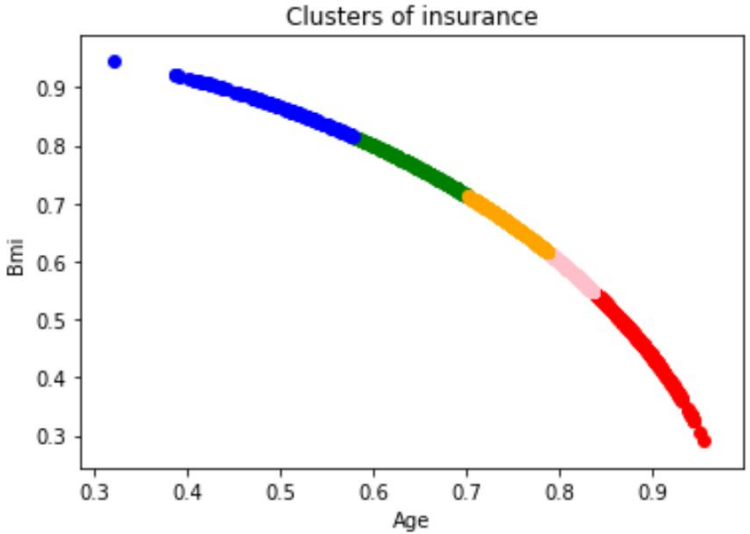
linkage = avarage

affinity = cosine



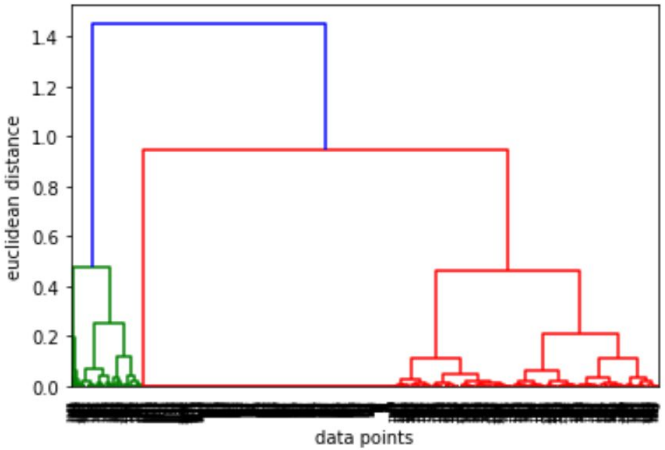
k = 5

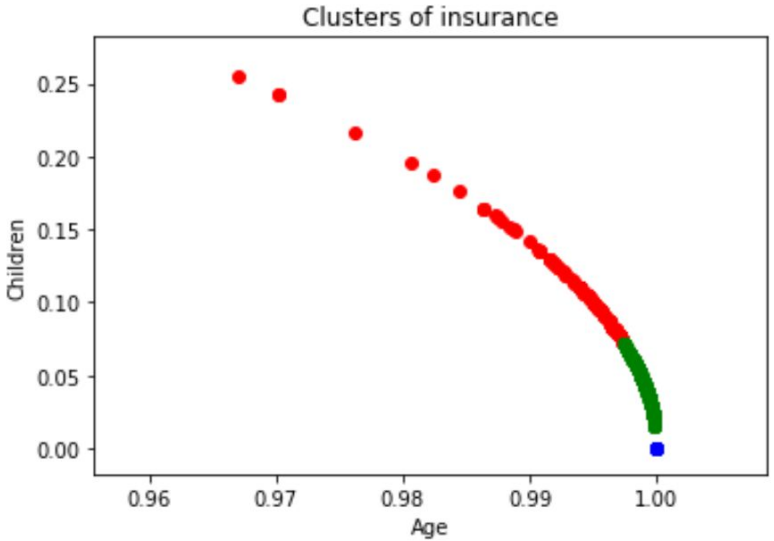
Diğer parametreler sabitken k' yı arttırdığımda :



2) Age ve children kolonları

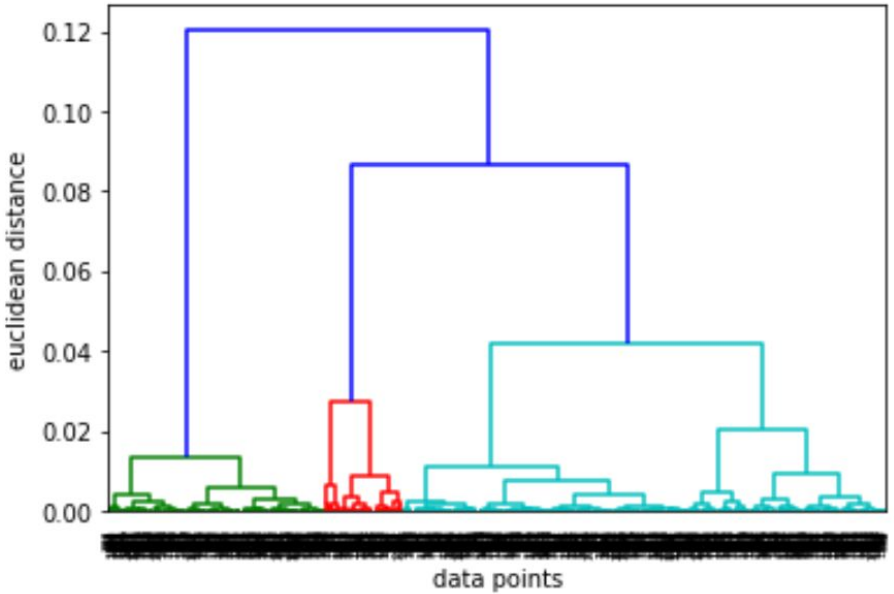
Parametreler default değerde iken :

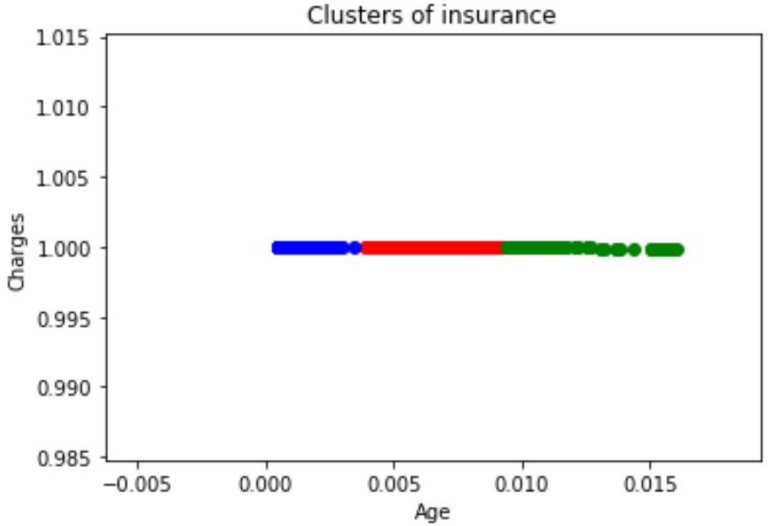




3) Age ve charges kolonları

Parametreler default değerde iken :





KARŞILAŞTIRMA

1. **Hierarchical clustring**
    - Veri seti çok büyükse dendrogram oluşturmak çok uzun sürüyor.
    - Veri tipine göre en iyi parametre ve method seçimleri yapılabilir.
    - K-means clustring' e göre daha çok seçenek var.
    - Küme sayısına daha kolay karar verilebilir dendrogram ile.
    - Treshold kullanılabilir.
2. **K-means clustring**
    - Dirsek metodu net olarak bir küme sayısı söylemiyor.
    - Kırılma olan yere göre biz karar veriyoruz.
    - Hierarchical clustring' e göre daha basit ve daha az method seçeneği var.

ÖĞRENİLENLER

1. K-means Clustring
2. Wcss Metriği (Kümeler içi kareler toplamı)
3. Hierarchical clustring
4. Agglomerative (Parçadan bütüne)
5. Linkage çeşitleri (complete, average, single)
6. Dendrogram (Öbek Ağacı)

KAYNAKÇA

https://medium.com/@ekrem.hatipoglu/machine-learning-clustering-kümeleme-k-means-algorithm-part-13-be33aeef4fc8

https://scikit-learn.org/stable/modules/generated/sklearn.cluster.AgglomerativeClustering.html

https://scikit-learn.org/stable/modules/generated/sklearn.cluster.AgglomerativeClustering.html

DİNLEDİĞİNİZ İÇİN TEŞEKKÜRLER