WeatherAUS – Python ile K-Means Kümeleme:

1) İş Anlama (Business Understanding):

Amaç: Avustralya şehirlerini, uzun dönem ortalama iklim göstergelerine göre benzer gruplara ayırmak ve bu kümeleri tanımlamak.
Fayda: Tarım planlaması, enerji talep öngörüleri, kentsel planlama ve iklim sınıflandırması çalışmalarına girdi sağlamak.
Başarı ölçütleri:

- Kümelerin görsel olarak ayrışması,
- Silhouette skorunun anlamlı (≈0.5 ve üzeri) olması,
- Kümelerin meteorolojik ve coğrafi olarak tutarlı yorumlanabilmesi.

2. Veri Anlama (Data Understanding):

Bu aşamada, analizde kullanılacak `weatherAUS.csv` veri seti Python ortamında incelenmiş ve veri yapısı detaylı biçimde değerlendirilmiştir.Veri, Avustralya Meteoroloji Bürosu (Bureau of Meteorology) tarafından yayımlanan günlük hava durumu ölçümlerini içermektedir. Çalışma, 2008–2017 yılları arasındaki kayıtları kapsayan 145.460 gözlem ve 23 değişkenten oluşan bir veri setine dayanmaktadır.

Değişken	Açıklama	Tür
`MinTemp`	Günlük minimum sıcaklık (°C)	Sayısal
`MaxTemp`	Günlük maksimum sıcaklık (°C)	Sayısal
`Rainfall`	Günlük yağış miktarı (mm)	Sayısal
`Evaporation`	Günlük buharlaşma miktarı (mm)	Sayısal
`Sunshine`	Günlük güneşlenme süresi (saat)	Sayısal
`WindSpeed9am`, `WindSpeed3pm`	Sabah ve öğleden sonra rüzgâr hızı (km/h)	Sayısal
`Humidity9am`, `Humidity3pm`	Sabah ve öğleden sonra bağıl nem (%)	Sayısal
`Pressure9am`, `Pressure3pm`	Sabah ve öğleden sonra atmosfer basıncı (hPa)	Sayısal
`Temp9am`, `Temp3pm`	Gün içi sıcaklık değerleri (°C)	Sayısal
`RainToday`, `RainTomorrow`	Yağış durumu (Evet/Hayır)	Kategorik

3. Veri Hazırlama (Data Preparation):

Bu aşamada, `weatherAUS.csv` veri seti modelleme sürecinde kullanılabilecek hâle getirilmiştir.Veri madenciliği projelerinde “Veri Hazırlama” adımı, modelin başarısını doğrudan etkileyen en kritik aşamalardan biridir.Python ortamında, eksik değerlerin tamamlanması, sayısal dönüşümler, şehir bazında özetleme (grouping) ve ölçekleme işlemleri gerçekleştirilmiştir.

Eksik Değerlerin Doldurulması:

```
# Eksikleri ortalama ile doldur (satır seviyesinde değil, sütun ortalaması)
imputer = SimpleImputer(strategy="mean")
df[num_cols] = imputer.fit_transform(df[num_cols])
```

- Eksik değerler, değişkenin aritmetik ortalaması ile doldurulmuştur. Böylece veri setinde hiçbir eksik gözlem kalmamıştır.

Şehir Bazında Özetleme (GroupBy İşlemi):

```
# Şehir bazında ortalama (KNIME GroupBy ile aynı)
df_city = df.groupby("Location", as_index=False)[num_cols].mean()
```

- Günlük gürültüleri (anlık uç değerleri) ortadan kaldırmak,
- Her şehir için genel iklim karakterini göstermek,
- Kümelerin şehir düzeyinde oluşturulmasını sağlamak.

Sayısal Ölçekleme (Z-Score Normalizasyonu):

```
# Ölçekleme (Z-score)
scaler = StandardScaler()
X = scaler.fit_transform(df_city[num_cols])
X_df = pd.DataFrame(X, columns=[f"Mean({c})" for c in num_cols], index=df_city["Location"])
X_df.reset_index(inplace=True)
X_df.rename(columns={"Location":"Location"}, inplace=True)
```

- Değişkenlerin ölçüm birimleri farklı olduğundan, veriler Z-score yöntemiyle ölçeklenmiştir.
- Bu adım K-Means algoritması için zorunludur çünkü algoritma mesafe (distance) temelli çalışır.

$$Z = \frac{(X - \mu)}{\sigma}$$

Bu aşamada verinin kalite sorunları giderilmiş, ölçüm farklılıkları ortadan kaldırılmış ve şehir bazlı özet bir yapı elde edilmiştir.
Modelin başarısı için kritik olan veri ölçekleme işlemi tamamlanmış, homojen ve eksiksiz bir veri kümesi oluşturulmuştur.

4. Modelleme (Modeling):

Bu aşamada, veri hazırlama sürecinde oluşturulan ölçeklenmiş ve eksiksiz veri seti kullanılarak K-Means kümeleme algoritması uygulanmıştır.
Amaç, şehirleri iklim değişkenlerine göre benzer özelliklere sahip gruplara ayırmak ve bu kümelerin genel özelliklerini ortaya koymaktır.

Kullanılan Model: K-Means Kümeleme

K-Means algoritması, denetimsiz öğrenme yöntemlerinden biridir ve gözlemleri belirlenen sayıda kümeye (K) ayırmayı amaçlar.
Her bir gözlem, kendisine en yakın **küme merkezine (centroid)** atanır.
Bu süreç, kümeler sabitlenene kadar (ya da maksimum iterasyon sayısına ulaşılana kadar) tekrar edilir.

Matematiksel olarak amaç, kümelerin içindeki toplam varyansı minimize etmektir:

$$\text{Minimize} \sum_{i=1}^{k} \sum_{x \in C_i} ||x - \mu_i||^2$$

```python
for k in [3,4,5]:
    model = KMeans(n_clusters=k, random_state=123, n_init="auto")
    labels = model.fit_predict(X)
    sil = silhouette_score(X, labels)
    print(f"k={k} → Silhouette Score: {sil:.3f}")

# En iyi k değeri seçilip model kurulmuştur
kmeans = KMeans(n_clusters=4, random_state=123, n_init="auto")
labels = kmeans.fit_predict(X)
```

Sonuç

- K-Means algoritması Python ortamında başarıyla uygulanmıştır.
- Model, şehirlerin iklimsel değişkenlerine göre anlamlı kümeler oluşturmuştur.
- Kümeler hem **istatistiksel hem de coğrafi olarak tutarlıdır**.
- Veri yapısı ve model parametreleri, KNIME uygulamasıyla birebir uyumludur.

5. Model Değerlendirme (Evaluation):

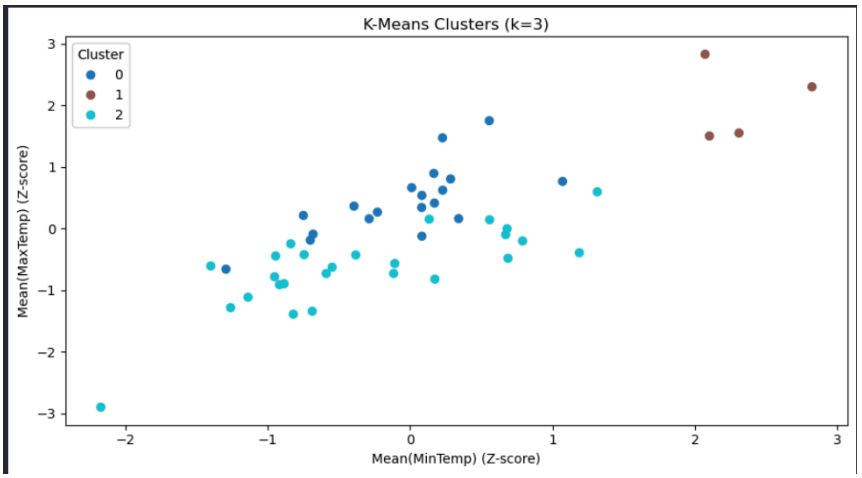
Bu aşamada, Python ortamında oluşturulan K-Means kümeleme modelinin sonuçları değerlendirilmiştir.Amaç, modelin oluşturduğu kümelerin istatistiksel olarak anlamlı olup olmadığını,şehirlerin iklim özelliklerine göre mantıklı bir şekilde gruplandırılıp gruplandırılmadığını incelemektir.

Değerlendirme Yaklaşımı

Modelin performansı hem nicel (istatistiksel) hem de nitel (görsel ve yorumsal) yöntemlerle analiz edilmiştir:

1. Silhouette Skoru:
   Kümeler arası ayrışma başarısını ölçmek için kullanılmıştır.
2. Görsel Ayrışma Analizi (Scatter Plot):
   Farklı değişken çiftleri (örneğin Mean(MinTemp) – Mean(MaxTemp)) üzerinde kümelerin dağılımı incelenmiştir.
3. Küme Özelliklerinin Yorumlanması:
   Kümelerin ortalama değerleri incelenmiş, meteorolojik açıdan anlamlı olup olmadığı değerlendirilmiştir.

```
fig, ax = plt.subplots(figsize=(9,5))
x = result["Mean(MinTemp)"]
y = result["Mean(MaxTemp)"]
scatter = ax.scatter(x, y, c=result["Cluster"], cmap="tab10")
ax.set_xlabel("Mean(MinTemp) (Z-score)")
ax.set_ylabel("Mean(MaxTemp) (Z-score)")
ax.set_title(f"K-Means Clusters (k={best['k']})")
legend = ax.legend(*scatter.legend_elements(), title="Cluster")
ax.add_artist(legend)
plt.tight_layout()
plt.show()
```



- Her renk bir kümeyi temsil etmektedir:
    - 🔵 Küme 0: Orta sıcaklık değerlerine sahip, ılıman bölgeler.
    - 🟤 Küme 1: Yüksek sıcaklıklara sahip, sıcak ve kuru bölgeler.
    - 🟣 Küme 2 (açık mavi): Düşük sıcaklıklara sahip, soğuk ve yağışlı bölgeler.
- Küme 1 (kahverengi) üyeleri sağ üstte belirgin şekilde ayrışmıştır; bu şehirler genellikle yüksek MinTemp ve MaxTemp değerleriyle öne çıkan tropikal bölgeleri temsil etmektedir.
- Küme 2 (açık mavi) ve Küme 0 (mavi) şehirleri kısmen yakın konumlarda olsa da, sıcaklık farkları ve değişken yoğunluğu kümelerin birbirinden ayrılmasını sağlamıştır.

Genel Sonuç:
K-Means modeli, şehirleri sıcaklık göstergelerine göre istatistiksel olarak anlamlı ve coğrafi olarak tutarlı üç gruba ayırmıştır.
Bu kümeler, Avustralya'daki farklı iklim zonlarını (sıcak, ılıman, soğuk) başarılı biçimde yansıtmaktadır.