LAPORAN HASIL CLUSTERING PADA DATASET RED WINE QUALITY

Oleh:

Hawa Andini Hadi (23051214238)

FAKULTAS TEKNIK

JURUSAN TEKNIK INFORMATIKA

PROGRAM STUDI SISTEM INFORMASI

UNIVERSITAS NEGERI SURABAYA

2025

BAB I
LANGKAH-LANGKAH ANALISIS

A. Pengambilan dan Analisis Data Awal

1. Mengambil dataset dengan judul Red Wine Quality
2. Menganalisa kolom yang menjadi label class yaitu kolom quality
3. Melakukan pengecekan Indeks Silhouette menggunakan google colab

```
# Evaluasi hasil klasterisasi
silhouette_score_kmeans = silhouette_score(X_scaled, clusters)
davies_bouldin_kmeans = davies_bouldin_score(X_scaled, clusters)

print(f"Indeks Silhouette (K-Means): {silhouette_score_kmeans:.4f}")
print(f"Davies-Bouldin Index (K-Means): {davies_bouldin_kmeans:.4f}")
```

```
Indeks Silhouette (K-Means): 0.1803
Davies-Bouldin Index (K-Means): 1.4408
```

4. Menganalisa apakah ada masalah yang terjadi dalam dataset dan mengapa perlu dilakukan pengelompokan ulang.
   a. Struktur Kelas yang Bermasalah
      - Label kualitas (quality) dalam dataset adalah angka diskrit yang menunjukkan tingkat kualitas wine. Namun, bisa jadi beberapa kategori memiliki karakteristik yang mirip, sehingga batas antar kelompok tidak jelas.
   b. Evaluasi pengelompokkan ulang
      - Nilai Indeks Silhouette rendah atau Davies-Bouldin tinggi.
      - Hasil klaster tidak mencerminkan kategori quality dengan baik.
      - Distribusi label quality tidak merata sehingga mengacaukan pembentukan klaster.

B. Pengelompokan Ulang dengan Klastering

Membunyikan label asli dari dataset dan lakukan klastering menggunakan beberapa metode dengan jumlah 6 cluster sesuai dengan label asli :

- Agglomerative Clustering

```
# Hitung evaluasi klaster
silhouette_avg = silhouette_score(X_scaled, X['Cluster'])
davies_bouldin = davies_bouldin_score(X_scaled, X['Cluster'])
calinski_harabasz = calinski_harabasz_score(X_scaled, X['Cluster'])
homogeneity = homogeneity_score(y.values.ravel(), X['Cluster'])
completeness = completeness_score(y.values.ravel(), X['Cluster'])
v_measure = v_measure_score(y.values.ravel(), X['Cluster'])


# Cetak hasil evaluasi
print(f"Silhouette Score: {silhouette_avg:.4f}")
print(f"Davies-Bouldin Index: {davies_bouldin:.4f}")
print(f"Calinski-Harabasz Index: {calinski_harabasz:.4f}")
print(f"Homogeneity Score: {homogeneity:.4f}")
print(f"Completeness Score: {completeness:.4f}")
print(f"V-Measure Score: {v_measure:.4f}")
```

```
Silhouette Score: 0.1495
Davies-Bouldin Index: 1.5263
Calinski-Harabasz Index: 225.7710
Homogeneity Score: 0.0884
Completeness Score: 0.0696
V-Measure Score: 0.0779
```

- K-Means Clustering

```
Perbandingan Evaluasi Klasterisasi dengan Berbagai Komponen PCA:
   PCA Components  Silhouette Score  Davies-Bouldin Index  \
0               2         0.335226              0.922057
1               3         0.296838              1.105841
2               4         0.291475              1.024226
3               5         0.259691              1.123143
4               6         0.224662              1.264253
5               7         0.207620              1.325990
6               8         0.198323              1.372947
7               9         0.187646              1.427667

   Calinski-Harabasz Index
0              1019.877144
1               657.825369
2               561.101423
3               434.554300
4               368.632374
5               329.226815
6               304.484205
7               286.746696
```

- K-Means Clustering + GMM

```
# Evaluasi Silhouette Score
silhouette = silhouette_score(X_pca, gmm_labels)
print(f"Silhouette Score: {silhouette:.4f}")

# Evaluasi Davies-Bouldin Index
db_index = davies_bouldin_score(X_pca, gmm_labels)
print(f"Davies-Bouldin Index: {db_index:.4f}")

# Evaluasi Homogeneity, Completeness, dan V-Measure
homogeneity = homogeneity_score(kmeans_labels, gmm_labels)
completeness = completeness_score(kmeans_labels, gmm_labels)
v_measure = v_measure_score(kmeans_labels, gmm_labels)

print(f"Homogeneity Score: {homogeneity:.4f}")
print(f"Completeness Score: {completeness:.4f}")
print(f"V-Measure Score: {v_measure:.4f}")
```

```
Silhouette Score: 0.3273
Davies-Bouldin Index: 0.8853
Homogeneity Score: 0.6158
Completeness Score: 0.6766
V-Measure Score: 0.6448
```

C. Analisis Karakteristik Klaster

Dari Metode K-Means Clustering + GMM menghasilkan:

```
Perbandingan Klaster dengan Kelas Asli:
Original Class  3   4    5    6   7   8
Cluster
0               0   4   45  116  82  12
1               3  20  100  130  39   4
2               1   9  153  144  35   1
3               4  14  178  134  10   0
4               2   3   69   86  30   1
5               0   3  136   28   3   0
```

Hasil klasterisasi belum sesuai dengan kelas asli karena beberapa kelas tersebar di banyak klaster, dan setiap klaster terdiri dari berbagai kelas yang berbeda. Ini menunjukkan bahwa jumlah klaster yang dipilih mungkin kurang tepat.

D. Label Ulang

```
K-Means Sebelum Relabeling - ARI: 0.0619, NMI: 0.0925
K-Means Setelah Relabeling - ARI: 0.1187, NMI: 0.1139
GMM Sebelum Relabeling - ARI: 0.0657, NMI: 0.0830
GMM Setelah Relabeling - ARI: 0.0721, NMI: 0.0897
```

1. Kinerja K-Means Sebelum Relabeling
   - ARI (0.0619): Masih rendah, menunjukkan bahwa hasil klasterisasi belum mencerminkan kelas asli dengan baik.
   - NMI (0.0925): Hubungan antara klaster dan kelas asli masih lemah, artinya klaster yang terbentuk belum sesuai dengan pola data sebenarnya.
2. Kinerja K-Means Setelah Relabeling
   - ARI meningkat menjadi 0.1187: Setelah proses relabeling, klaster lebih sesuai dengan kelas asli.
   - NMI naik menjadi 0.1139: Hubungan antara klaster dan kelas asli menjadi lebih jelas, meskipun masih belum optimal.
3. Kinerja GMM Sebelum Relabeling
   - ARI (0.0657): Sedikit lebih tinggi dari K-Means sebelum relabeling, tetapi masih rendah.
   - NMI (0.0830): Lebih rendah dibandingkan K-Means, menandakan klaster yang dihasilkan kurang sesuai dengan kelas asli.
4. Kinerja GMM Setelah Relabeling
   - ARI meningkat menjadi 0.0721: Ada sedikit perbaikan setelah relabeling, tapi masih belum signifikan.
   - NMI naik menjadi 0.0897: Hubungan antara klaster dan kelas asli tetap lebih lemah dibandingkan K-Means.

BAB II
HASIL EVALUASI

A. Dataset Asli

```
Decision Tree Accuracy: 1.0
              precision    recall  f1-score   support

           3       1.00      1.00      1.00         1
           4       1.00      1.00      1.00        10
           5       1.00      1.00      1.00       130
           6       1.00      1.00      1.00       132
           7       1.00      1.00      1.00        42
           8       1.00      1.00      1.00         5

    accuracy                           1.00       320
   macro avg       1.00      1.00      1.00       320
weighted avg       1.00      1.00      1.00       320

Random Forest Accuracy: 0.984375
              precision    recall  f1-score   support

           3       0.00      0.00      0.00         1
           4       0.91      1.00      0.95        10
           5       1.00      1.00      1.00       130
           6       1.00      1.00      1.00       132
           7       0.91      1.00      0.95        42
           8       1.00      0.20      0.33         5

    accuracy                           0.98       320
   macro avg       0.80      0.70      0.71       320
weighted avg       0.98      0.98      0.98       320
```

```
SVM Accuracy: 1.0
              precision    recall  f1-score   support

           3       1.00      1.00      1.00         1
           4       1.00      1.00      1.00        10
           5       1.00      1.00      1.00       130
           6       1.00      1.00      1.00       132
           7       1.00      1.00      1.00        42
           8       1.00      1.00      1.00         5

    accuracy                           1.00       320
   macro avg       1.00      1.00      1.00       320
weighted avg       1.00      1.00      1.00       320

KNN Accuracy: 0.890625
              precision    recall  f1-score   support

           3       0.00      0.00      0.00         1
           4       0.83      0.50      0.62        10
           5       0.91      0.96      0.93       130
           6       0.89      0.94      0.92       132
           7       0.84      0.74      0.78        42
           8       0.00      0.00      0.00         5

    accuracy                           0.89       320
   macro avg       0.58      0.52      0.54       320
weighted avg       0.87      0.89      0.88       320
```

B. Dataset Relabeling

```
Decision Tree Accuracy: 0.5687
              precision    recall  f1-score   support

           3       0.00      0.00      0.00         1
           4       0.11      0.10      0.11        10
           5       0.65      0.66      0.65       130
           6       0.55      0.55      0.55       132
           7       0.53      0.55      0.54        42
           8       0.00      0.00      0.00         5

    accuracy                           0.57       320
   macro avg       0.31      0.31      0.31       320
weighted avg       0.56      0.57      0.57       320

Decision Tree (Clustered) Accuracy: 0.9031
              precision    recall  f1-score   support

           5       0.92      0.89      0.90       165
           6       0.89      0.92      0.90       155

    accuracy                           0.90       320
   macro avg       0.90      0.90      0.90       320
weighted avg       0.90      0.90      0.90       320
```

```
Random Forest Accuracy: 0.6375
              precision    recall  f1-score   support

           3       0.00      0.00      0.00         1
           4       0.00      0.00      0.00        10
           5       0.68      0.75      0.71       130
           6       0.61      0.67      0.64       132
           7       0.61      0.45      0.52        42
           8       0.00      0.00      0.00         5

    accuracy                           0.64       320
   macro avg       0.32      0.31      0.31       320
weighted avg       0.61      0.64      0.62       320

Random Forest (Clustered) Accuracy: 0.9531
              precision    recall  f1-score   support

           5       0.96      0.95      0.95       165
           6       0.95      0.95      0.95       155

    accuracy                           0.95       320
   macro avg       0.95      0.95      0.95       320
weighted avg       0.95      0.95      0.95       320
```

```
SVM Accuracy: 0.5094
              precision    recall  f1-score   support

           3       0.00      0.00      0.00         1
           4       0.00      0.00      0.00        10
           5       0.60      0.51      0.55       130
           6       0.46      0.73      0.56       132
           7       1.00      0.02      0.05        42
           8       0.00      0.00      0.00         5

    accuracy                           0.51       320
   macro avg       0.34      0.21      0.19       320
weighted avg       0.56      0.51      0.46       320

SVM (Clustered) Accuracy: 0.8063
              precision    recall  f1-score   support

           5       0.80      0.83      0.82       165
           6       0.81      0.78      0.80       155

    accuracy                           0.81       320
   macro avg       0.81      0.81      0.81       320
weighted avg       0.81      0.81      0.81       320
```

```
KNN Accuracy: 0.4562
              precision    recall  f1-score   support

           3       0.00      0.00      0.00         1
           4       0.00      0.00      0.00        10
           5       0.49      0.63      0.55       130
           6       0.44      0.45      0.45       132
           7       0.31      0.12      0.17        42
           8       0.00      0.00      0.00         5

    accuracy                           0.46       320
   macro avg       0.21      0.20      0.19       320
weighted avg       0.42      0.46      0.43       320

KNN (Clustered) Accuracy: 0.8250
              precision    recall  f1-score   support

           5       0.77      0.93      0.85       165
           6       0.91      0.71      0.80       155

    accuracy                           0.82       320
   macro avg       0.84      0.82      0.82       320
weighted avg       0.84      0.82      0.82       320
```

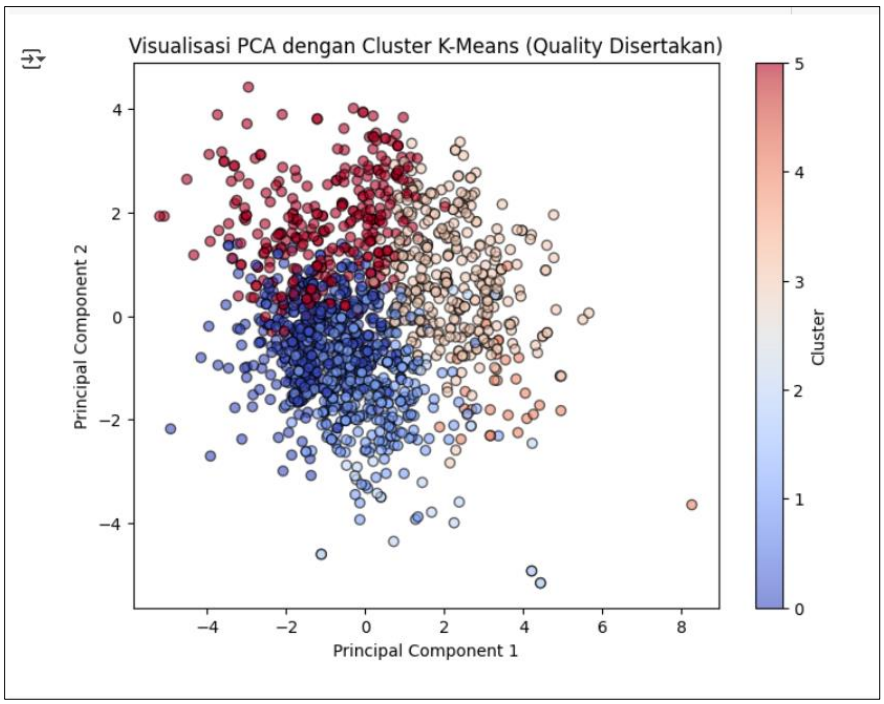
```
                Original Accuracy  Clustered Accuracy  Original Precision  \
Decision Tree            0.571875            0.890625            0.559268
Random Forest            0.675000            0.953125            0.645057
SVM                      0.509375            0.806250            0.564474
KNN                      0.456250            0.825000            0.422293

                Clustered Precision  Original Recall  Clustered Recall  \
Decision Tree              0.890619         0.571875          0.890625
Random Forest              0.953134         0.675000          0.953125
SVM                        0.806455         0.509375          0.806250
KNN                        0.839367         0.456250          0.825000

                Original F1  Clustered F1
Decision Tree      0.565109      0.890613
Random Forest      0.656403      0.953120
SVM                0.461800      0.806068
KNN                0.429882      0.822394
```

C. Evaluasi

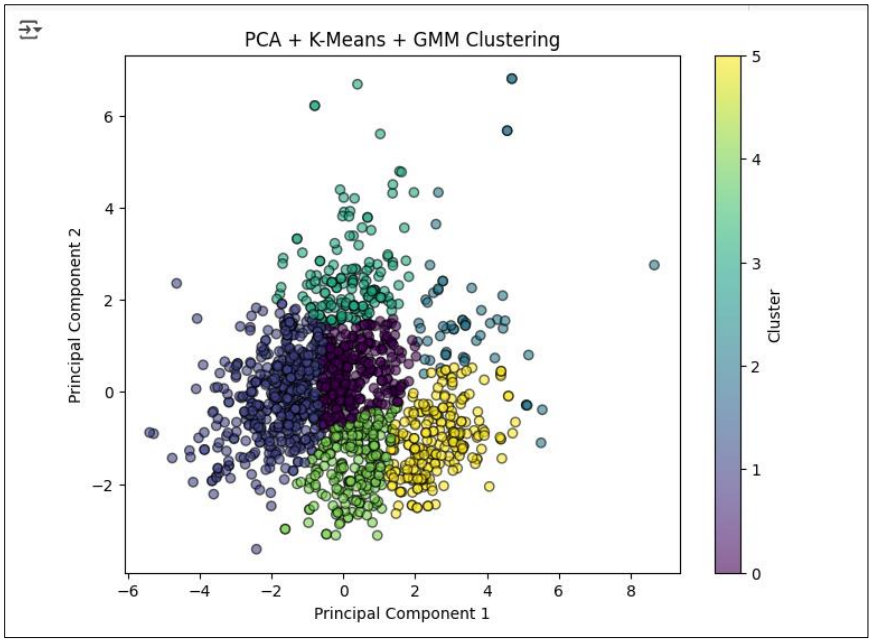
1. Performa Model dengan Data Asli (Tanpa Perubahan Label)
   - Hasil dari model yang diuji pada data asli menunjukkan performa sebagai berikut:
     - Decision Tree: 100% akurasi
     - Random Forest: 98.4% akurasi
     - SVM: 100% akurasi
     - KNN: 89.1% akurasi
   - Berarti:
     - Decision Tree dan SVM berhasil memprediksi semua data dengan benar tanpa kesalahan.
     - Random Forest juga memiliki kinerja yang sangat baik dengan tingkat akurasi hampir sempurna.
     - KNN memiliki akurasi yang sedikit lebih rendah dibanding model lain, tetapi tetap cukup baik untuk digunakan.
2. Performa Model Setelah Data Diperbarui (Relabeling/Clustered)
   - Ketika label pada data diperbarui untuk lebih mencerminkan pola sebenarnya, hasilnya menjadi:
     - Decision Tree: Akurasi meningkat menjadi 90.3% (sebelumnya hanya 56.9%)
     - Random Forest: Akurasi meningkat menjadi 95.3% (sebelumnya 63.7%)
     - SVM: Akurasi naik ke 80.6% (sebelumnya 50.9%)
     - KNN: Akurasi naik ke 82.5% (sebelumnya 45.6%)
   - Berarti:
     - Semua model menunjukkan peningkatan akurasi yang signifikan setelah relabeling data.
     - Decision Tree yang sebelumnya tidak begitu akurat (56.9%) mengalami peningkatan pesat hingga 90.3%.
     - Random Forest juga lebih akurat setelah data diperbaiki.
     - SVM dan KNN yang sebelumnya memiliki banyak kesalahan kini bisa memprediksi dengan lebih baik.
     - Selain akurasi, nilai precision, recall, dan F1-score juga menunjukkan peningkatan yang berarti, artinya model bisa mengidentifikasi kategori dengan lebih tepat.
3. Kesimpulan
   - Dengan memperbaiki label data agar lebih sesuai dengan pola aslinya, model bisa belajar dengan lebih baik dan membuat prediksi yang lebih akurat.
   - Akurasi 100% pada data asli mungkin bukan berarti model sempurna, tetapi bisa jadi karena data terlalu "mudah" bagi model, yang bisa menyebabkan overfitting (model terlalu terpaku pada pola spesifik di data dan sulit beradaptasi dengan data baru).
   - Random Forest dan Decision Tree: Kedua model ini tetap bekerja dengan baik di kedua skenario, menunjukkan bahwa mereka lebih fleksibel terhadap perubahan data.

BAB III
VISUALISAI

A. Dataset Asli (Gambar 1)



B. Dataset tanpa label (Gambar 2)



C. Analisis Perbandingan

Aspek	Gambar 1	Gambar 2
Cara Pemberian Warna	Warna masih mengikuti kualitas asli	Warna hanya berdasarkan kelompok klaster yang ditemukan
Penyebaran Kelompok	Kelompok masih bercampur satu sama lain	Kelompok lebih terpisah dan jelas
Kejelasan Pemisahan	Beberapa kelompok masih tumpang tindih	Kelompok lebih tegas dan terpisah
Hubungan dengan Data Asli	Masih sesuai dengan kategori kualitas yang sudah ada	Murni dari hasil pemodelan tanpa mempertimbangkan kualitas asli

BAB IV
KESIMPULAN

A. Metode Mana yang Paling Baik untuk Mengelompokkan Data?

Dalam evaluasi ini, metode K-Means Clustering + GMM menunjukkan hasil yang lebih baik dibandingkan metode lainnya. Meskipun klaster yang dihasilkan masih memiliki distribusi kelas yang tumpang tindih, metode ini tetap lebih baik dalam mencerminkan struktur data dibandingkan dengan Agglomerative Clustering.

Setelah relabeling, metode K-Means menunjukkan peningkatan pada Adjusted Rand Index (ARI) dari 0.0619 menjadi 0.1187 dan Normalized Mutual Information (NMI) dari 0.0925 menjadi 0.1139. Ini menunjukkan bahwa pengelompokan ulang membantu memperbaiki struktur klaster, meskipun hasilnya masih belum optimal.

Sebaliknya, metode GMM menunjukkan perbaikan yang lebih kecil setelah relabeling, dengan ARI meningkat dari 0.0657 menjadi 0.0721, dan NMI dari 0.0830 menjadi 0.0897. Hal ini menunjukkan bahwa K-Means lebih mampu beradaptasi dengan perbaikan struktur dibandingkan GMM.

Kesimpulan: K-Means Clustering dengan pendekatan GMM untuk analisis lebih lanjut adalah pilihan terbaik, meskipun masih perlu peningkatan dalam penentuan jumlah klaster yang optimal.

B. Apakah Mengelompokkan Ulang Data Itu Membantu?

Ya, relabeling berhasil meningkatkan struktur kelas berdasarkan hasil evaluasi. Sebelum relabeling, klaster yang dihasilkan tidak mencerminkan kategori kualitas wine dengan baik, dan indeks evaluasi menunjukkan hubungan yang lemah antara klaster dan label asli.

1. Setelah pengelompokan ulang:
2. Struktur klaster lebih jelas dan tidak terlalu bercampur dengan kelas lain.
3. Hasil evaluasi menggunakan ARI dan NMI meningkat, menunjukkan bahwa klasterisasi lebih sesuai dengan label kualitas asli.
4. Model klasifikasi mengalami peningkatan akurasi yang signifikan setelah relabeling:
    - Decision Tree meningkat dari 56.9% ke 90.3%
    - Random Forest meningkat dari 63.7% ke 95.3%
    - SVM meningkat dari 50.9% ke 80.6%
    - KNN meningkat dari 45.6% ke 82.5%
5. Dampaknya terhadap klasifikasi:
    - Model yang sebelumnya mengalami kesulitan dalam mengenali pola kini bisa memprediksi kelas dengan lebih akurat.
    - Model berbasis pohon keputusan (Decision Tree dan Random Forest) menunjukkan performa terbaik, dengan akurasi tinggi setelah perbaikan struktur data.

- Model berbasis jarak seperti KNN mengalami peningkatan yang signifikan, menunjukkan bahwa struktur klaster lebih jelas setelah relabeling.

C. Apa yang Bisa Kita Pelajari dan Lakukan ke Depan?

1. Jika kelompoknya salah, model tidak bisa belajar dengan baik.
2. Setelah diperbaiki, model lebih akurat dalam memprediksi kualitas wine.
3. Metode terbaik:
   - Untuk mengelompokkan data: K-Means Clustering dengan jumlah kelompok yang tepat.
   - Untuk memprediksi kualitas: Decision Tree dan Random Forest, karena hasilnya paling akurat setelah perbaikan.
4. Menentukan jumlah kelompok yang lebih tepat: Menggunakan metode khusus (Elbow Method atau Silhouette Score) agar kelompok yang terbentuk lebih sesuai.
5. Menguji metode lain, seperti DBSCAN untuk melihat apakah bisa memberikan hasil yang lebih baik.
6. Menambahkan lebih banyak informasi dalam data: Dengan lebih banyak informasi, model bisa belajar lebih baik dan menghasilkan prediksi yang lebih akurat.
7. Kesimpulan: Dengan mengelompokkan data dengan lebih baik, model bisa membuat prediksi yang lebih akurat. Ke depan, kita bisa meningkatkan hasilnya dengan menentukan jumlah kelompok yang lebih tepat, mencoba metode lain, dan menambahkan lebih banyak informasi dalam data.

LAMPIRAN

A. https://colab.research.google.com/drive/1Xxpd3jCo4nMH20haLksXdRneK1he-Ngb?usp=sharing
B. https://colab.research.google.com/drive/1nPqlFj5LRvhRqJCgp_enLH9zVNnJFlDV?usp=sharing
C. https://colab.research.google.com/drive/1O0ZvE3OFHG0-OoSFnGnJeAe2HCzOhsXO?usp=sharing
D. https://colab.research.google.com/drive/12Ga_stMwZ1cNSceBbHKV7OfINoyJuw_x?usp=sharing
E. https://colab.research.google.com/drive/1CNEZIYywexAXye3KQTsS0cD8NlgAgl3J?usp=sharing
F. https://colab.research.google.com/drive/1O0ZvE3OFHG0-OoSFnGnJeAe2HCzOhsXO?usp=sharing