

DAFTAR RIWAYAT HIDUP



Saya Anggiq Karisma Aji Restu, juga dikenal sebagai Anggiq. Saya lahir pada tanggal 3 Juli 2000 di Banyuwangi, Jawa Timur. Saya adalah anak pertama dari pasangan Agus Triyono dan Natalisa Dwi Kristiani. belajar di SDN 004 Bontang dari 2007 hingga 2013, SMPN 4 Bontang dari 2013 hingga 2016, dan SMKN 3 Jurusan Geologi Pertambangan dari 2016 hingga 2019. Sebelumnya Pada tahun 2019–2020, penulis pernah magang di BPN(Badan Pertanahan Nasional) Kota Bontang selama 3 bulan dan pernah bekerja posisi maintenance di salah satu perusahaan BUMN yaitu Pertamina dan Gas Kota Bontang selama 6 bulan. Pada tahun 2020, penulis menjadi mahasiswa Universitas Muhammadiyah Kalimantan Timur, Fakultas Sains dan Teknologi, jurusan Teknik Informatika. Saat menjadi mahasiswa, penulis juga melakukan magang selama 3 bulan di Disdukcapil dilaksanakan pada semester 7. Demikian deskripsi riwayat hidup yang penulis sampaikan jika terdapat kesalahan atau kekurangan mohon dimaafkan karena kesempurnaan hanya milik Sang Maha pencipta, maka penulis mengharapkan kritik dan saran mengenai skripsi ini.

LAMPIRAN

Lampiran 1 Codingan

- Mengimport modul yang dibutuhkan

```
from pyswarm import pso
from sklearn.neighbors import KNeighborsClassifier
from skrebate import ReliefF
from imblearn.over_sampling import SMOTE
from sklearn.model_selection import cross_val_score, KFold
from sklearn.preprocessing import OneHotEncoder
from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import OrdinalEncoder
from sklearn.preprocessing import StandardScaler
from sklearn.feature_selection import SelectKBest, f_classif
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.metrics import accuracy_score, f1_score, recall_score
from sklearn.metrics import precision_score, classification_report, ConfusionMatrixDisplay
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
import pandas as pd
```

- Mengimport file csv

```
#Melakukan import terhadap dataset banjir
data = pd.read_csv('banjir.csv')
df = pd.DataFrame(data)
```

- Membuat model KNN dan melakukan 10-fold, serta confusion matrix

```
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import KFold
from sklearn.metrics import confusion_matrix, accuracy_score, classification_report
import numpy as np
import pandas as pd

# Asumsi: X_res dan y_res telah didefinisikan sebelumnya
# Membuat array k dengan nilai k dari 3 sampai 15
k_values = np.arange(3, 16)

# Variable untuk menyimpan hasil akurasi dan confusion matrices
all_fold_scores = {}
all_confusion_matrices = {}

# K-fold cross validation
kf = KFold(n_splits=10, shuffle=True, random_state=42)

for k in k_values:
    print(f"\nk = {k}")

    fold_scores = []
    fold_confusion_matrices = []
    kFold = 1

    for train_index, test_index in kf.split(X_res, y_res):
        X_train, X_test = X_res.iloc[train_index], X_res.iloc[test_index]
        y_train, y_test = y_res.iloc[train_index], y_res.iloc[test_index]
```

```

# Inisiasi algoritma k-nearest neighbors
knn = KNeighborsClassifier(n_neighbors=k, metric='euclidean')

# Melatih model KNN
knn.fit(X_train, y_train)

# Predict on the test set
y_pred = knn.predict(X_test)

# Mencari nilai confusion matrix
cMetrics = confusion_matrix(y_test, y_pred)
fold_confusion_matrices.append(cMetrics)
print(f"Fold {kFold} : Confusion Matrix")
print(cMetrics)

# Mendapatkan nilai akurasi
fold_accuracy = accuracy_score(y_test, y_pred)
fold_scores.append(fold_accuracy)

# Menampilkan nilai akurasi
print(f"Fold {kFold}: Accuracy: {fold_accuracy:.2f}")

# Report
print(f"{kFold} Report\n{classification_report(y_test, y_pred)}")

# Update nilai kFold
kFold += 1

# Menyimpan hasil rata-rata akurasi dan confusion matrices untuk setiap nilai k
all_fold_scores[k] = np.mean(fold_scores)
all_confusion_matrices[k] = fold_confusion_matrices

# Menampilkan hasil akurasi untuk setiap nilai k
for k, avg_accuracy in all_fold_scores.items():
    print(f"k = {k}: Rata-rata Akurasi = {avg_accuracy * 100:.2f}%")

k = 3
Fold 1 : Confusion Matrix
[[68 17]

```

- Membuat fungsi untuk melakukan perhitungan Relief

```
# Import pustaka yang diperlukan
from skrebate import ReliefF
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Inisiasi ReliefF dengan parameter yang sesuai
relief = ReliefF(n_neighbors=15) # Parameter n_neighbors dapat disesuaikan

# Melakukan fit transform pada data
X_transformed = relief.fit_transform(X.values, y.values)

# Mendapatkan indeks fitur yang dipilih berdasarkan ranking
selected_features_indices = relief.top_features_

# Mendapatkan nama fitur yang dipilih
selected_features_names = X.columns[selected_features_indices]

# Mendapatkan bobot fitur
feature_weights = relief.feature_importances_[selected_features_indices]

# Membuat DataFrame dengan nama fitur dan bobotnya
feature_importances = pd.DataFrame({
    'Feature': selected_features_names,
    'Weight': feature_weights
})

# Mengurutkan fitur berdasarkan bobot
feature_importances = feature_importances.sort_values(by='Weight', ascending=False)

# Menampilkan fitur terbaik beserta bobotnya
print("Fitur terbaik beserta bobotnya:")
print(feature_importances)

# Membuat grafik bobot fitur
plt.figure(figsize=(10, 6))
sns.barplot(data=feature_importances, x='Weight', y='Feature', palette='viridis')
plt.title('Feature Importance Scores from ReliefF')
plt.xlabel('Importance Score (Weight)')
plt.ylabel('Feature')
plt.show()
```

- Membuat ulang dataframe serta penerapan ulang KNN, 10-fold, Confusion Matrix dan fitur terpilih

```

# Fitur terbaik dari seleksi fitur ReliefF
top_features = ['Arah-angin-terbanyak', 'Kecepatan-angin', 'Kecepatan-angin-rata-rata', 'Arah-angin-maksimum']

# Memisahkan variabel atribut dan target berdasarkan fitur yang dipilih
X_res_selected = X_res[top_features]
y_res = y_res

# Membuat array k dengan nilai k dari 3 sampai 15
k_values = np.arange(3, 16)

# Variable untuk menyimpan hasil akurasi dan confusion matrices
all_fold_scores = {}
all_confusion_matrices = {}

# K-fold cross validation
kf = KFold(n_splits=10, shuffle=True, random_state=42)

for k in k_values:
    print(f"\nk = {k}")

    fold_scores = []
    fold_confusion_matrices = []
    kFold = 1

    for train_index, test_index in kf.split(X_res_selected, y_res):
        X_train, X_test = X_res_selected.iloc[train_index], X_res_selected.iloc[test_index]
        y_train, y_test = y_res.iloc[train_index], y_res.iloc[test_index]

        # Inisiasi algoritma k-nearest neighbors
        knn = KNeighborsClassifier(n_neighbors=k, metric='euclidean')

        # Melatih model KNN
        knn.fit(X_train, y_train)

        # Predict on the test set
        y_pred = knn.predict(X_test)

        # Mencari nilai confusion matrix
        cMetrics = confusion_matrix(y_test, y_pred)
        fold_confusion_matrices.append(cMetrics)
        print(f"Fold {kFold} : Confusion Matrix")
        print(cMetrics)

        # Mendapatkan nilai akurasi
        fold_accuracy = accuracy_score(y_test, y_pred)
        fold_scores.append(fold_accuracy)

        # Menampilkan nilai akurasi
        print(f"Fold {kFold}: Accuracy: {fold_accuracy:.2f}")

        # Report
        print(f"{kFold} Report\n{classification_report(y_test, y_pred)}")

        # Update nilai kFold
        kFold += 1

    # Menyimpan hasil rata-rata akurasi dan confusion matrices untuk setiap nilai k
    all_fold_scores[k] = np.mean(fold_scores)
    all_confusion_matrices[k] = fold_confusion_matrices

# Menampilkan hasil akurasi untuk setiap nilai k
for k, avg_accuracy in all_fold_scores.items():
    print(f"k = {k}: Rata-rata Akurasi = {avg_accuracy * 100:.2f}%")

```

- Penerapan PSO

```
# Function to optimize KNN
def optimize_knn(x, k):
    leaf_size = int(x[0])
    p = int(x[1])

    clf = KNeighborsClassifier(n_neighbors=k, leaf_size=leaf_size, p=p)
    scores = cross_val_score(clf, X_res, y_res, cv=10)

    return -scores.mean() # Maximizing accuracy by minimizing the negative score

# Bounds for KNN hyperparameters
lb = [10, 1] # Lower bounds for leaf_size and p
ub = [50, 2] # Upper bounds for leaf_size and p

# List of odd k values to optimize
k_values = [3, 5, 7, 9, 11, 13, 15]

optimal_parameters = {}

for k in k_values:
    # Perform PSO optimization for each k
    xopt, fopt = pso(optimize_knn, lb, ub, args=(k,), swarmsize=50, maxiter=30)

    # Extract optimal parameters for this k
    optimal_leaf_size = int(xopt[0])
    optimal_p = int(xopt[1])
    optimal_score = -fopt

    # Store the optimal parameters and corresponding score for this k
    optimal_parameters[k] = {
        'optimal_leaf_size': optimal_leaf_size,
        'optimal_p': optimal_p,
        'optimal_score': optimal_score
    }

    # Print the optimal parameters and the corresponding score
    print(f"Optimal parameters for k = {k}:")
    print(f"Optimal leaf size: {optimal_leaf_size}")
    print(f"Optimal p: {optimal_p}")
    print(f"Optimal score: {optimal_score}")
    print("")

# Display all optimal parameters
for k, params in optimal_parameters.items():
    print(f"k = {k}:")
    print(f"  Optimal leaf size: {params['optimal_leaf_size']}")
    print(f"  Optimal p: {params['optimal_p']}")
    print(f"  Optimal score: {params['optimal_score']}")
```

- Membuat ulang dataframe serta penerapan PSO, 10-fold, Confusion Matrix dan fitur terpilih

```

# Memisahkan variabel atribut dan target
X_res = X_res
y_res = y_res

# Optimal parameters from PSO
optimal_k_values = [3, 5, 7, 9, 11, 13, 15]
optimal_leaf_sizes = [10, 34, 11, 49, 12, 21, 11]
optimal_p = 1

# Variable untuk menyimpan hasil akurasi dan confusion matrices untuk setiap nilai k
all_fold_scores = {}
all_confusion_matrices = {}

# K-fold cross validation
kf = KFold(n_splits=10, shuffle=True, random_state=42)

for i, k in enumerate(optimal_k_values):
    fold_scores = []
    fold_confusion_matrices = []
    leaf_size = optimal_leaf_sizes[i]

    kFold = 1
    for train_index, test_index in kf.split(X_res, y_res):
        X_train, X_test = X_res.iloc[train_index], X_res.iloc[test_index]
        y_train, y_test = y_res.iloc[train_index], y_res.iloc[test_index]

        # Inisiasi algoritma k-nearest neighbors dengan parameter hasil optimasi PSO
        knn = KNeighborsClassifier(n_neighbors=k, leaf_size=leaf_size, p=optimal_p, metric='minkowski')

        # Melatih model KNN
        knn.fit(X_train, y_train)

        # Predict on the test set
        y_pred = knn.predict(X_test)

        # Mencari nilai confusion matrix
        cMetrics = confusion_matrix(y_test, y_pred)
        fold_confusion_matrices.append(cMetrics)
        print(f"Fold {kFold}, k = {k} : Confusion Matrix")
        print(cMetrics)

        # Mendapatkan nilai akurasi
        fold_accuracy = accuracy_score(y_test, y_pred)
        fold_scores.append(fold_accuracy)

        # Menampilkan nilai akurasi
        print(f"Fold {kFold}, k = {k}: Accuracy: {fold_accuracy:.2f}")

        # Report
        print(f"Fold {kFold}, k = {k} Report\n{classification_report(y_test, y_pred)}")

        # Update nilai kFold
        kFold += 1

    # Menyimpan hasil rata-rata akurasi dan confusion matrices untuk setiap nilai k
    all_fold_scores[k] = np.mean(fold_scores)
    all_confusion_matrices[k] = fold_confusion_matrices

# Menampilkan hasil akurasi rata-rata untuk setiap nilai k
for k, avg_accuracy in all_fold_scores.items():
    print(f"k = {k}: Rata-rata Akurasi = {avg_accuracy * 100:.2f}%")

```

Lampiran 2 Surat Pengantar Pengambilan Data BMKG

 **UMKT**
Program Studi
Teknik Informatika
Fakultas Sains dan Teknologi

Telp. 0541-748511 Fax. 0541-766832
Website <http://informatika.umkt.ac.id>
email: informatika@umkt.ac.id

UNIVERSITAS MUHAMMADIYAH
KALIMANTAN TIMUR
Berakhlak | Berprestasi | Berkeadilan

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Nomor : 003-011/FST.1/A.7/C/2024
Lampiran : -
Perihal : Permohonan Pengambilan Data

Kepada Yth.
Kepala Badan Meteorologi, Klimatologi dan Geofisika (BMKG)
di -

Tempat

Assalamu'alaikum Warrahmatullahi Wabarrakatuh

Puji Syukur kepada Allah Subhanahu wa ta'ala yang senantiasa melimpahkan Rahmat-Nya kepada kita sekalian. Aamiin.

Sehubungan untuk memenuhi Tugas Akhir/Skripsi Tahun Akademik 2023/2024, maka dengan ini kami mengajukan permohonan untuk melakukan pengambilan data di Meteorologi, Klimatologi dan Geofisika (BMKG) Kota Samarinda. Adapun data yang diminta yaitu data parameter (temperatur maksimum, temperatur minimum, temperatur rata-rata, kelembaban rata-rata, curah hujan, lamanya penyinaran matahari, kecepatan angin maksimum, arah angin maksimum, kecepatan angin rata-rata, dan arah angin terbanyak), dengan nama mahasiswa sebagai berikut:

No	Nama	NIM	Program Studi
1	Raenald Syaputra	2011102441040	Teknik Informatika
2	Ilham Taufiq	2011102441152	Teknik Informatika
3	Anggiq Karisma Aji Restu	2011102441089	Teknik Informatika
4	Vito Junivan Rivaldo	2011102441019	Teknik Informatika
5	Faldy Alfareza Pambudi	2011102441097	Teknik Informatika

Demikian surat permohonan ini dibuat. Atas perhatian dan kerjasamanya kami mengucapkan terima kasih.

Wassalamu'alaikum Warrahmatullahi Wabarrakatuh

Samarinda, 7 Ramadhan 1445 H
18 Maret 2024 M

 Kepala Program Studi S1 Teknik Informatika

Muhammad Fauzan, S.Kom., M.TI
N. 1118019203

Kampus 1 : Jl. Ir. H. Juanda, No.15, Samarinda
Kampus 2 : Jl. Pelita, Pesona Mahakam, Samarinda

Lampiran 3 Surat Pengantar Pengambilan Data BPBD



UMKT
Program Studi
Teknik Informatika
Fakultas Sains dan Teknologi

Telp. 0541-748511 Fax. 0541-766832
Website <http://informatika.umkt.ac.id>
email: informatika@umkt.ac.id



بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Nomor : 003-008/FST.1/A.7/C/2024
Lampiran : -
Perihal : Permohonan Pengambilan Data

Kepada Yth.
Kepala Badan Penanggulangan Bencana Daerah Kota Samarinda
di -
Tempat

Assalamu'alaikum Warrahmatullahi Wabarrakatuh

Puji Syukur kepada Allah Subhanahu wa ta'ala yang senantiasa melimpahkan Rahmat-Nya kepada kita sekalian. Aamiin.

Sehubungan untuk memenuhi Tugas Akhir/Skripsi Tahun Akademik 2023/2024, maka dengan ini kami mengajukan permohonan untuk melakukan pengambilan data di Badan Penanggulangan Bencana Daerah Kota Samarinda. Adapun data yang diminta yaitu berupa data banjir tahun 2021-2023 Kota Samarinda, dengan nama mahasiswa sebagai berikut:

No	Nama	NIM	Program Studi
1	Raenald Syaputra	2011102441040	Teknik Informatika
2	Ilham Taufiq	2011102441152	Teknik Informatika
3	Anggiq Karisma Aji Restu	2011102441089	Teknik Informatika
4	Faldy Alfareza Pambudi	2011102441097	Teknik Informatika
5	Vito Junivan Rivaldo	2011102441019	Teknik Informatika

Demikian surat permohonan ini dibuat. Atas perhatian dan kerjasamanya kami mengucapkan terima kasih.

Wassalamu'alaikum Warrahmatullahi Wabarrakatuh

Samarinda, 7 Ramadhan 1445 H
18 Maret 2024 M



Program Studi S1 Teknik Informatika

[Signature]
Ansyah, S.Kom., M.TI
IDN. 1118019203

Kampus 1 : Jl. Ir. H. Juanda, No.15, Samarinda
Kampus 2 : Jl. Pelita, Pesona Mahakam, Samarinda



Program Studi Teknik Informatika Universitas Pamulang ISSN: 2654-3788
Jl. Raya Pasirjati No. 40, Booran, Setono, Kota Tangerang Selatan, Banten, Indonesia 15110 e-ISSN: 2654-4229

Jurnal Teknologi Sistem Informasi dan Aplikasi

ISSN: SK no. 0005-26543788/31.1.1/SK-ISSN/2018.10 - 5 Oktober 2018 (mulai edisi Vol. 1, No. 1, Oktober 2018)
e-ISSN: SK no. 0005-26544229/31E.3.1/SK-ISSN/2018.10 - 29 Oktober 2018 (mulai edisi Vol. 1, No. 1, Oktober 2018)



Date: July 02nd, 2024

Letter of Acceptance

Dear Authors:
Anggiq Karisma Aji Restu, Taghfirul Azhima Yoga Siswa*, Wawan Joko Pranoto
Teknik Informatika, Universitas Muhammadiyah Kalimantan Timur, Samarinda, Indonesia
75124
e-mail: tay758@umkt.ac.id

It's a great pleasure to inform you that after the peer review process, your article entitled "**Model Optimasi KNN-PSORF dalam Menangani High Dimensional Data Banjir Kota Samarinda**" has been **Accepted** and considered for publication in Jurnal Teknologi Sistem Informasi dan Aplikasi (ISSN: 2654-3788 e-ISSN: 2654-4229) **Volume 7, Issue 3, July 2024**.

Thank you for submitting your work to this journal. We hope to receive in future too.

Best Regards,
Editor-in-Chief
Jurnal Teknologi Sistem Informasi dan Aplikasi

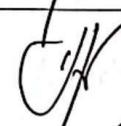


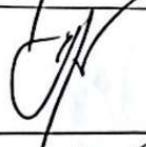
(Aries Saifudin)

<http://openjournal.unpam.ac.id/index.php/JTISI>

KARTU KENDALI BIMBINGAN LAPORAN KARYA ILMIAH

Nama : Anggiq Karisma Aji Restu
 NIM : 2011102441089
 Nama Dosen Pembimbing : Taghfirul Azhima Yoga Siswa, S.Kom, M.Kom
 Judul Penelitian : Model Optimasi KNN-PSORF dalam menangani High Dimensional Data Banjir Kota Samarinda

No	Tanggal	Uraian Pembimbingan	Paraf Dosen
1	22-01-2024	Pembahasan tahapan dalam Penelitian SKRIPSI	
2	7-02-2024	Pertemuan Kedua, mencari Paper/artikel rujukan Penelitian	
3	16-02-2024	Pertemuan Ketiga, Renew survei Paper untuk mencari permasalahan pada klasifikasi data mining	
4	17-02-2024	Pertemuan Keempat, Renew Technical Paper dan road maps Penelitian	
5	22-02-2024	Pertemuan Kelima, mencari Paper dan artikel sesuai rujukan terkait objek	
6	29-02-2024	Pertemuan Keenam, Penentuan judul Penelitian	
7	13-03-2024	Pertemuan Ketujuh, Pembuatan Canvas Penelitian	
8	15-03-2024	Pertemuan Kedelapan, Pengajuan surat permohonan data untuk Penelitian	
9	18-03-2024	Pertemuan Kesembilan, Revisi Proposal bab 1, bab 2 dan Perbaiki Penulisan	
10	24-04-2024	Pertemuan Kesepuluh, Revisi Proposal bab 1, bab 2 dan Perbaiki Penulisan	

11.	10 Mei 2024	Pertemuan Keseberas, bimbingan Rempahasan penulisan Bab 3 dan bab 4	
12.	17 Mei 2024	Pertemuan kedua belas, Revisi naskah skripsi bab 3 dan bab 4	
13.	27 Mei 2024	Konsultasi Perbalkan Penulisan hasil Penelitian dan artikel Jurnal	
14.	23-06-2024	Konsultasi dan Perbalkan penyusunan Penulisan artikel Jurnal	

Dosen Pembimbing



Taghfirul Azhima Yoga Siswa, S.Kom, M.Kom
NIDN. 1118038805



Mengetahui

Kaplan Program Studi



Taghfirul Azhima Yoga Siswa, S.Kom, M.TI
NIDN. 1118019203

SKRIPSI ANGGIQ KARISMA AJI RESTU

by Teknik Informatika Universitas Muhammadiyah Kalimantan Timur



Submission date: 12-Jul-2024 12:59PM (UTC+0800)

Submission ID: 2415566973

File name: Model_Optimasi_KNNPSORF_-_ANGGIQ_KARISMA_AJI_RESTU.docx (759.6K)

Word count: 8174

Character count: 52503

SKRIPSI ANGGIQ KARISMA AJI RESTU

ORIGINALITY REPORT

26% SIMILARITY INDEX	22% INTERNET SOURCES	15% PUBLICATIONS	5% STUDENT PAPERS
--------------------------------	--------------------------------	----------------------------	-----------------------------

PRIMARY SOURCES

1	dspace.umkt.ac.id Internet Source	9%
2	Raenald Syaputra, Taghfirul Azhima Yoga Siswa, Wawan Joko Pranoto. "Model Optimasi SVM Dengan PSO-GA dan SMOTE Dalam Menangani High Dimensional dan Imbalance Data Banjir", Teknika, 2024 Publication	2%
3	Ari Ahmad Dhani, Taghfirul Azhima Yoga Siswa, Wawan Joko Pranoto. "Perbaikan Akurasi Random Forest Dengan ANOVA Dan SMOTE Pada Klasifikasi Data Stunting", Teknika, 2024 Publication	2%
4	repository.ub.ac.id Internet Source	2%
5	journal.umpr.ac.id Internet Source	1%
6	ejournal.ikado.ac.id Internet Source	1%