

RAMALAN TEMPOH TINGGAL PESAKIT DI ICU MENGGUNAKAN PEMBELAJARAN MESIN

Izzah Hanani Binti Tusiman^{1*}, Ts. Dr. Norsamsiah Sani²

¹*Fakulti Teknologi & Sains Maklumat, Universiti Kebangsaan Malaysia, 43600 UKM Bangi, Selangor Darul Ehsan, Malaysia*

²*Center for Artificial Intelligence Technology, Universiti Kebangsaan Malaysia, 43600 UKM Bangi, Selangor Darul Ehsan, Malaysia*

ABSTRAK

Kajian ini memfokuskan kepada aplikasi teknik pembelajaran mesin untuk meramalkan tempoh tinggal pesakit di Unit Rawatan Rapi (ICU). Ketidakpastian mengenai tempoh tinggal pesakit di ICU menimbulkan cabaran besar untuk peruntukan sumber hospital dan penjagaan pesakit. Objektif utama projek ini adalah untuk membangunkan model ramalan tempoh tinggal pesakit di ICU berdasarkan pembelajaran mesin untuk membantu profesional penjagaan kesihatan dalam menguruskan penempatan pesakit di ICU dengan berkesan, dengan itu meningkatkan hasil pesakit dan penggunaan sumber. Metodologi CRISP-DM digunakan dalam projek ini sebagai panduan untuk reka bentuk dan metodologi projek. Data dikumpulkan daripada set data sumber terbuka Microsoft, yang termasuk demografi pesakit, tanda-tanda penting dan tempoh penginapan. Tiga model pembelajaran mesin, termasuk Hutan Rawak, XGBoost dan Regresi Linear, telah digunakan untuk meramalkan tempoh penginapan ICU. Analisis prestasi perbandingan antara ketiga-tiga model mengenal pasti XGBoost sebagai model berprestasi terbaik dalam meramalkan tempoh tinggal ICU, yang mengatasi prestasi dua model lain dengan mencapai Ralat Purata Purata Akar (RMSE) terendah iaitu 0.46 dan R-kuadrat (R^2) tertinggi iaitu nilai 0.96. Model ini juga mempunyai perbezaan prestasi yang signifikan secara statistik berbanding model kedua terbaik, Random Forest, dalam meramalkan tempoh tinggal ICU. Penemuan ini menyerlahkan keberkesanan XGBoost dalam meramalkan tempoh penginapan ICU dengan tepat, yang boleh membantu hospital mengoptimumkan peruntukan sumber dan mengurangkan masa menunggu pesakit. Kajian itu mengesyorkan memfokuskan pada jenis pesakit tertentu dan bekerjasama dengan hospital untuk mendapatkan cerapan terperinci untuk meningkatkan lagi ketepatan ramalan dan penjagaan pesakit.

Kata kunci: Ramalan Tempoh Tinggal Pesakit di ICU, Pembelajaran Mesin, XGBoost, Hospital

PENGENALAN

Unit rawatan rapi hospital, atau ICU, merupakan satu jabatan khas di hospital dengan alat khusus dan kakitangan perubatan yang berkelayakan untuk merawat pesakit yang sakit parah atau cedera. Apabila seseorang memerlukan penjagaan kecemasan untuk menangani keadaan

yang mengancam nyawa, unit ini bertanggungjawab menyediakannya. Di samping itu, bilangan pesakit dan tempoh duduk di ICU adalah tidak tetap dan berbeza-beza disebabkan tahap kesakitan dan komplikasi yang berbeza. Kadang-kala ada sesetengah pesakit mengambil tempoh yang sekejap seperti sehari atau dua hari sahaja. Namun, terdapat juga pesakit yang duduk di unit tersebut yang mengambil tempoh masa yang lama seperti berminggu-minggu disebabkan tahap kesihatannya yang kritikal. Ini mengakibatkan isu yang berterusan mengenai ketersediaan atau kekosongan yang terhad di Unit Rawatan Rapi (ICU) di sektor kesihatan menimbulkan satu cabaran. Ini adalah disebabkan pesakit yang memerlukan rawatan kritikal sering mengalami masa menunggu yang berpanjangan untuk kemasukan. Inti cabaran ini terletak pada tempoh tinggal yang tidak dapat diramalkan untuk pesakit yang sedia ada, kerana keadaan dan status kesihatan mereka yang tidak menentu. Ketidakpastian ini bukan sahaja menimbulkan kebimbangan bagi pesakit yang memerlukan penempatan segera ICU tetapi juga memburukkan lagi tekanan pada kapasiti keseluruhan hospital untuk menangani kes kritikal dengan segera.

Oleh itu, skop objektif utama projek ini dijalankan adalah untuk membangunkan aplikasi berasaskan web yang diintegrasikan dengan pembelajaran mesin bagi meramalkan tempoh tinggal pesakit di ICU berdasarkan kondisi pesakit. Bagi mencapai objektif utama projek, terdapat tiga proses yang ditekankan iaitu, yang pertama, proses pembangunan model ramalan tempoh tinggal pesakit di ICU dan mengenalpasti model ramalan yang terbaik berasaskan teknik pembelajaran mesin. Kemudian, yang kedua ialah pemilihan fitur bagi melakukan ramalan tempat tinggal pesakit di ICU. Seterusnya, proses yang ketiga ialah membangunkan aplikasi berasaskan web yang bertujuan untuk pengguna melakukan ramalan tempoh tinggal pesakit di ICU berasaskan model pembelajaran mesin yang terbaik. Aplikasi berasaskan web ini merangkumi halaman utama, panduan kepenggunaan laman web, dan halaman ramalan tempoh tinggal pesakit di ICU. Maklumat seperti ramalan tempoh tinggal pesakit, dan tarikh jangkaan pesakit keluar dari ICU akan dipaparkan di halaman ramalan tempoh tinggal pesakit di ICU.

Skop projek ini melibatkan pembangunan aplikasi berasaskan web yang menggunakan rangka kerja *Streamlit* dan bahasa pengaturcaraan *Python* untuk melakukan ramalan tempoh tinggal pesakit di ICU melalui teknik pembelajaran mesin. Sasaran pengguna laman web ini ialah pekerja di sektor kesihatan, terutamanya di hospital. Teras laman web ini berdasarkan model pembelajaran mesin yang menggunakan bahasa pengaturcaraan *Python*, memanfaatkan pembelajaran mesin dan perpustakaan pemprosesan seperti *scikit-learn* dan *numpy*. Model ini akan dilatih dengan set data yang merangkumi tahap kondisi bagi setiap tanda vital seperti kadar denyutan jantung dan kadar glukosa, tempoh tinggal pesakit, dan demografi pesakit seperti jantina. Set data pesakit ini akan melalui pra-pemprosesan data bagi mengoptimumkan kondisi data bagi latihan model dijalankan.

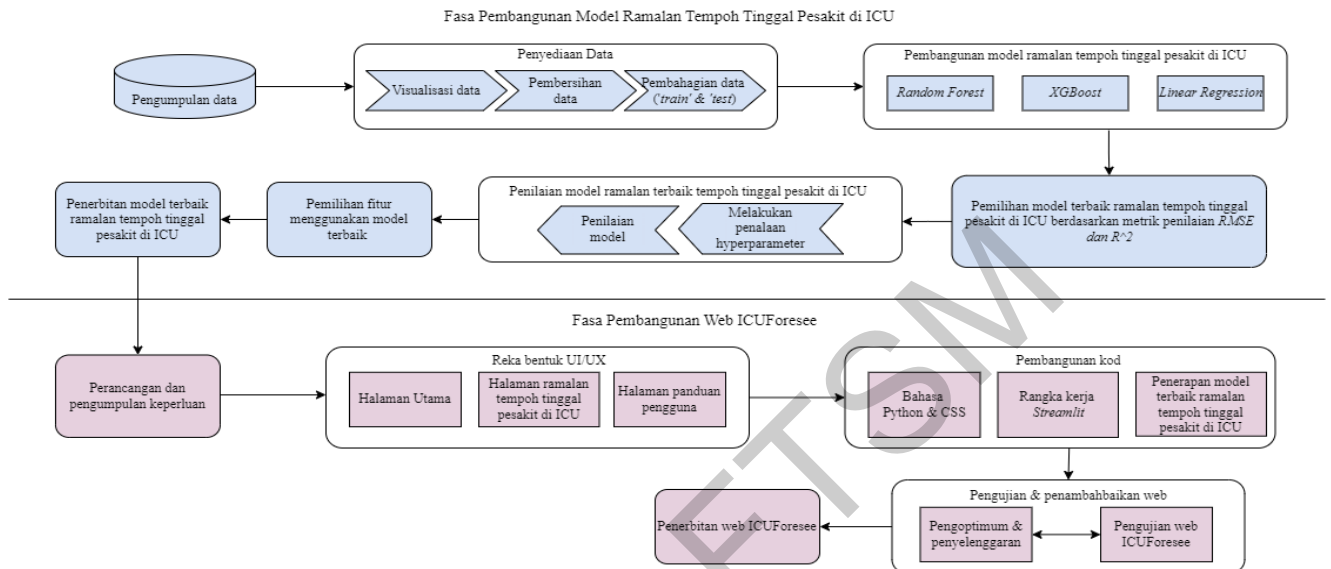
Projek ini mempunyai kepentingan yang besar kerana ia bertujuan untuk membangunkan aplikasi berasaskan web yang menggunakan pembelajaran mesin untuk meramalkan tempoh tinggal pesakit di ICU. Hasil daripada projek ini dapat memberikan

manfaat yang signifikan terhadap industri kesihatan. Dengan laman web ini, ia mampu membantu pekerja sektor kesihatan dalam mendapatkan ramalan tempoh pendudukan seseorang pesakit dan mendapatkan tarikh keluar masuk pesakit ICU. Akibatnya, ia boleh memberi manfaat kepada hospital dengan memperuntukkan sumber yang lebih baik, termasuk katil ICU, kakitangan perubatan dan peralatan. Ini dapat membantu dalam mengoptimumkan penggunaan sumber dan mengurangkan masa menunggu pesakit yang akan memasuki ICU. Tambahan pula, ramalan ini membolehkan hospital berisiko tinggi yang memerlukan ICU tinggal lebih lama pada awal proses rawatan. Ini membolehkan campur tangan awal dan pengurusan proaktif, yang berpotensi meningkatkan hasil pesakit.

Dalam projek ini, proses pembangunan mengikut metodologi CRISP-DM (Proses Standard Merentas Industri untuk Perlombongan Data), merangkumi empat fasa utama. Fasa 1 ialah Kajian Kebolehlaksanaan, di mana masalah dan objektif dikenal pasti, dan penilaian kemajuan penyelidikan dalam pembelajaran mesin dijalankan. Fasa 2 melibatkan Pengumpulan dan Penyediaan Data, di mana set data pesakit di ambil daripada sumber terbuka yang disediakan oleh *Microsoft*, diikuti dengan pra-pemprosesan data. Fasa 3 memberi tumpuan kepada Pemodelan dan Penilaian Model, yang melibatkan pembangunan model pembelajaran mesin untuk meramalkan tempoh tinggal pesakit di ICU dan menilai keberkesannya. Fasa terakhir, Pembangunan Aplikasi Berasaskan Web, menggunakan model terbaik untuk mencipta sistem ramalan tempoh tinggal pesakit di ICU dan memaparkan maklumat penting seperti ramalan tempoh tinggal pesakit dan tarikh jangkaan keluar pesakit.

Laporan teknikal ini disusun dengan jelas dan teratur. Ia bermula dengan bahagian Pengenalan, yang merangkumi ringkasan latar belakang projek, objektif, skop, justifikasi, kepentingan, metodologi dan organisasi laporan. Berikutan ini ialah bahagian Metodologi Kajian, yang memperincikan kaedah dan pendekatan yang digunakan. Bahagian Keputusan dan Perbincangan membentangkan dapatan dan menyediakan perbincangan berdasarkan keputusan. Seterusnya, bahagian Kesimpulan meringkaskan perkara utama laporan. Akhir sekali, laporan diakhiri dengan bahagian Pengakuan dan Rujukan.

METODOLOGI KAJIAN



Rajah 1 Menunjukkan fasa-fasa pembangunan model dan web ramalan tempoh tinggal pesakit di ICU

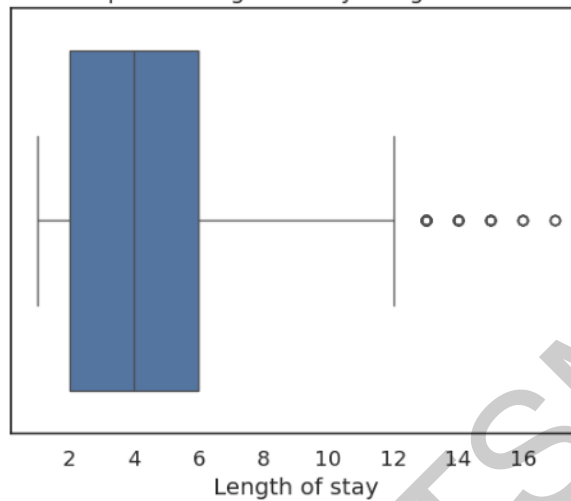
Carta alir yang menggambarkan kemajuan aplikasi berasaskan web ICUForesee dari fasa ke fasa penciptaan model pembelajaran mesin kepada pembangunan aplikasi berasaskan web adalah seperti di dalam Rajah 1 di atas. Platform *Google Collaboratory*, bahasa pengaturcaraan *Python* dan perpustakaan terkenal seperti *scikit-learn* dan *Matplotlib* digunakan sepanjang fasa penciptaan model pembelajaran mesin. Dengan menggunakan rangka kerja *Streamlit* dan bahasa pengaturcaraan *Python*, model pembelajaran mesin terbaik akan dimasukkan ke dalam laman web ICUForesee semasa peringkat pembangunan laman web ICUForesee. Pengujian laman web ICUForesee dilakukan pada fasa akhir ini untuk memastikan keperluan dan onjektif kajian dapat dipenuhi dengan sewajarnya.

Penyediaan Data

Set data pesakit diambil daripada set data sumber terbuka yang disediakan oleh *Microsoft*, yang tersedia di *GitHub*. Set data ini mengandungi maklumat komprehensif tentang pesakit ICU, termasuk pelbagai pembolehubah perubatan dan demografi yang penting untuk meramalkan tempoh tinggal. Kebolehcapaian dan kebolehpercayaan set data ini adalah penting untuk kejayaan projek, kerana ia menyediakan asas yang kukuh untuk latihan dan pengesahan model. Pada masa yang sama, prosedur pembersihan data dan normalisasi data yang menyeluruh dimulakan untuk membetulkan sebarang ralat, data nombor yang hilang atau ketidakkonsistenan dalam set data. Kualiti data dipertingkatkan dengan pemeriksaan yang teliti, menyediakan asas yang kukuh untuk pemodelan ramalan yang tepat. Selepas pembersihan dan normalisasi, set data pesakit akan terbahagi kepada 'training' dan 'testing' untuk mengajar model pembelajaran mesin mengenali corak dan perhubungan dalam data.

a. Pembersihan data

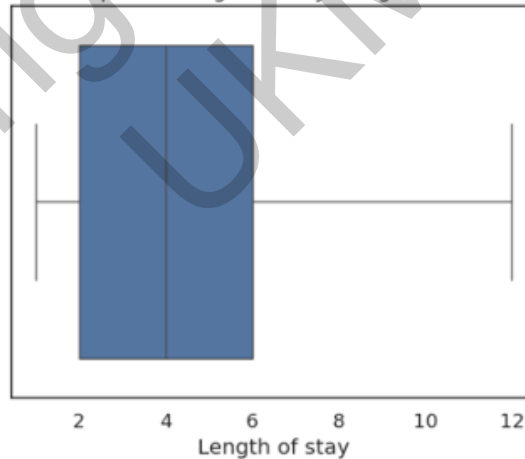
Boxplot of Length of stay using Seaborn



Rajah 2 Boxplot tempoh tinggal pesakit sebelum penyingkiran 'outlier'

Rajah 2 di atas, menunjukkan terdapat kewujudan *outlier* di dalam dataset. Outlier ialah titik data yang berada di luar julat *whisker*, seperti yang ditunjukkan di rajah di atas dimana titik yang lebih jauh daripada *whisker*. Nilai ini dianggap luar biasa tinggi atau rendah berbanding dengan data yang lain. Terdapat beberapa outlier dalam plot ini dengan tinggal antara 12 hingga 16 hari. Selepas pembersihan dilakukan dengan melakukan pembuangan 'outlier', data menjadi lebih bersih seperti yang ditunjukkan di rajah 3 di bawah.

Boxplot of Length of stay using Seaborn



Rajah 3 Boxplot tempoh tinggal pesakit selepas penyingkiran 'outlier'

b. Pembahagian Data

Pembahagian data merupakan peringkat penting dalam analisis data dan pembelajaran mesin. Di dalam peringkat ini, set data dibahagikan kepada dua atau lebih subset untuk pelbagai kegunaan, termasuk ujian model, latihan dan penilaian. Bagi mencari pembahagian data yang terbaik bagi melatih dan menilai model, empat set pembahagian data dilakukan dengan nisbah

(90:10), (80:20), (70:30), dan (60:40). Di bawah menunjukkan rajah-rajah pengaturcaraan dalam pembahagian data.

```
[ ] # Define the split ratios (train+validation, test)
ratios = [(0.9, 0.1), (0.8, 0.2), (0.7, 0.3), (0.6, 0.4)]

# Dictionary to hold the splits
splits = {}

for train_val_ratio, test_ratio in ratios:
    # Split into train+validation and test
    X_train_val, X_test, y_train_val, y_test = train_test_split(X, y, test_size=test_ratio, random_state=42)

    # Further split train+validation into train and validation
    validation_ratio = 0.2 # Assuming 20% of train+validation is for validation
    train_ratio = 1 - validation_ratio
    X_train, X_val, y_train, y_val = train_test_split(X_train_val, y_train_val, test_size=validation_ratio, random_state=42)

    key = f'{int(train_val_ratio*100)}_{int(test_ratio*100)}'
    splits[key] = {
        'X_train': X_train,
        'X_val': X_val,
        'X_test': X_test,
        'y_train': y_train,
        'y_val': y_val,
        'y_test': y_test
    }
}
```

Rajah 4 Pembahagian set data mengikut nisbah

Coretan kod ini mencipta kamus pemisahan data untuk latihan pembelajaran mesin kepada pengesahan dan ujian. Ia berulang mengikut nisbah set data latihan dan dan ujian yang ditentukan.

Pembangunan Model Pembelajaran Mesin

Fasa pemodelan ini akan mengaplikasikan pelbagai teknik pemodelan yang terpilih, menghasilkan reka bentuk ujian, dan penyelarasan tetapan parameter kepada nilai yang ideal. Fasa ini juga tertumpu kepada pencarian atribut yang berkesan untuk meramalkan tinggal pesakit di unit rawatan rapi. Terdapat tiga jenis model pembelajaran mesin yang akan dibangunkan di dalam projek kajian ini iaitu *Random Forest (RF)*, *eXtreme Gradient Boost (XGBoost)*, dan *Linear Regression (LR)*. Set data yang akan digunakan untuk sesi latihan model bagi ketiga-tiga model pembelajaran mesin ini akan mengandungi fitur-fitur seperti data sejarah perubatan, rekod kemasukan, keputusan makmal dan tanda-tanda vital. Prestasi model akan dinilai menggunakan metrik ralat kuasa dua (RMSE) dan pekali penentuan R^2 .

a. Pembangunan model *Random Forest*

Random Forest (RF) merupakan satu algoritma yang dibangunkan oleh Leo Breiman dan Adele Cutler. (R, 2023). *Random Forest* mengagregasi beberapa keluaran pokok keputusan dalam menghasilkan satu keputusan. Ramalan beberapa pokok keputusan digabungkan dalam teknik pembelajaran algoritma ini meningkatkan kebolehgeneralisasian model dan prestasi ramalan keseluruhan. Pendekatan ensemble *Random Forest* meningkatkan daya tahannya apabila digunakan dengan data yang belum diuji dan mengurangkan peluang untuk model mengalami overfitting.

```

import numpy as np
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_squared_error, r2_score
# model training and evaluation using RandomForestRegressor

modelrf= RandomForestRegressor(random_state=42)

# Train the model on the training set for 90,10
modelrf.fit(X_train_90_10, y_train_90_10)

y_val_pred = modelrf.predict(X_val_90_10)
val_rmse_90_10 = np.sqrt(mean_squared_error(y_val_90_10, y_val_pred))
val_r2 = r2_score(y_val_90_10, y_val_pred)
print(f'Validation RMSE for 90-10 split: {val_rmse_90_10:.2f}')
print(f'Validation R2 for 90-10 split: {val_r2:.2f}')

# Evaluate the model on the test set
y_test_pred = modelrf.predict(X_test_90_10)
test_rmse = np.sqrt(mean_squared_error(y_test_90_10, y_test_pred))
print(f'Test RMSE for 90-10 split: {test_rmse:.2f}')
test_r2 = r2_score(y_test_90_10, y_test_pred)
print(f'Test R2 for 90-10 split: {test_r2:.2f}')

# Train the model on the training set for 80,20
modelrf.fit(X_train_80_20, y_train_80_20)

y_val_pred = modelrf.predict(X_val_80_20)
val_rmse = np.sqrt(mean_squared_error(y_val_80_20, y_val_pred))
val_r2 = r2_score(y_val_80_20, y_val_pred)
print()
print(f'Validation RMSE for 80-20 split: {val_rmse:.2f}')
print(f'Validation R2 for 80-20 split: {val_r2:.2f}')

```

Rajah 5 Kod latihan dan ujian model Random Forest dengan setiap nisbah pembahagian data

```

# Evaluate the model on the test set
y_test_pred = modelrf.predict(X_test_80_20)
test_rmse = np.sqrt(mean_squared_error(y_test_80_20, y_test_pred))
print(f'Test RMSE for 80-20 split: {test_rmse:.2f}')
test_r2 = r2_score(y_test_80_20, y_test_pred)
print(f'Test R2 for 80-20 split: {test_r2:.2f}')

# Train the model on the training set for 90,10
modelrf.fit(X_train_70_30, y_train_70_30)

y_val_pred = modelrf.predict(X_val_70_30)
val_rmse = np.sqrt(mean_squared_error(y_val_70_30, y_val_pred))
val_r2 = r2_score(y_val_70_30, y_val_pred)
print()
print(f'Validation RMSE for 70_30 split: {val_rmse:.2f}')
print(f'Validation R2 for 70_30 split: {val_r2:.2f}')

# Evaluate the model on the test set
y_test_pred = modelrf.predict(X_test_70_30)
test_rmse = np.sqrt(mean_squared_error(y_test_70_30, y_test_pred))
print(f'Test RMSE for 70-30 split: {test_rmse:.2f}')
test_r2 = r2_score(y_test_70_30, y_test_pred)
print(f'Test R2 for 70_30 split: {test_r2:.2f}')

# Train the model on the training set for 80,20
modelrf.fit(X_train_60_40, y_train_60_40)

y_val_pred = modelrf.predict(X_val_60_40)
val_rmse = np.sqrt(mean_squared_error(y_val_60_40, y_val_pred))
val_r2 = r2_score(y_val_60_40, y_val_pred)
print()
print(f'Validation RMSE for 60_40 split: {val_rmse:.2f}')
print(f'Validation R2 for 60_40 split: {val_r2:.2f}')

# Evaluate the model on the test set
y_test_pred = modelrf.predict(X_test_60_40)
test_rmse = np.sqrt(mean_squared_error(y_test_60_40, y_test_pred))
print(f'Test RMSE for 60_40 split: {test_rmse:.2f}')
test_r2 = r2_score(y_test_60_40, y_test_pred)
print(f'Test R2 for 60_40 split: {test_r2:.2f}')

```

Rajah 6 Kod latihan dan ujian model Random Forest dengan setiap nisbah pembahagian data

b. Pembangunan model *eXtreme Gradient Boost (XGBoost)*

eXtreme Gradient Boost (XGBoost) ialah satu algoritma yang mempunyai perpustakaan peningkatan kecerunan teragih yang dioptimumkan untuk kecekapan dan kebolehskalaan dalam latihan model pembelajaran mesin. Rangsangan kecerunan (Gradient Boosting) ialah pendekatan di mana model baharu dicipta yang meramalkan sisa atau ralat model terdahulu dan kemudian ditambah bersama untuk membuat ramalan akhir. Ia dipanggil rangsangan kecerunan kerana ia menggunakan algoritma penurunan kecerunan untuk meminimumkan kehilangan apabila menambah model baharu. (Brownlee, 2021). Oleh kerana XGBoost boleh memodelkan tak linear, ia boleh digunakan untuk menangkap corak kompleks dalam data, menjadikan ia mempunyai salah satu aspek dalam membangunkan model ramalan yang berkesan.

```

import xgboost as xgb
from sklearn.metrics import mean_squared_error, r2_score
from xgboost import XGBRegressor

modelxgb = XGBRegressor(random_state=42)

# Train the model on the training set for 80,20
modelxgb.fit(X_train_90_10, y_train_90_10)

y_val_pred = modelxgb.predict(X_val_90_10)
xgbval_rmse = np.sqrt(mean_squared_error(y_val_90_10, y_val_pred))
xgbval_r2 = r2_score(y_val_90_10, y_val_pred)
print(f'Validation RMSE for 90_10 split: {xgbval_rmse:.2f}')
print(f'Validation R2 for 90_10 split: {xgbval_r2:.2f}')

# Evaluate the model on the test set
y_test_pred = modelxgb.predict(X_test_90_10)
xgbtest_rmse = np.sqrt(mean_squared_error(y_test_90_10, y_test_pred))
print(f'Test RMSE for 90_10 split: {xgbtest_rmse:.2f}')
xgbtest_r2 = r2_score(y_test_90_10, y_test_pred)
print(f'Test R2 for 90_10 split: {xgbtest_r2:.2f}')

# Train the model on the training set for 80,20
modelxgb.fit(X_train_80_20, y_train_80_20)

y_val_pred = modelxgb.predict(X_val_80_20)
xgbval_rmse = np.sqrt(mean_squared_error(y_val_80_20, y_val_pred))
xgbval_r2 = r2_score(y_val_80_20, y_val_pred)
print()
print(f'Validation RMSE for 80_20 split: {xgbval_rmse:.2f}')
print(f'Validation R2 for 80_20 split: {xgbval_r2:.2f}')

# Evaluate the model on the test set
y_test_pred = modelxgb.predict(X_test_80_20)
xgbtest_rmse = np.sqrt(mean_squared_error(y_test_80_20, y_test_pred))
print(f'Test RMSE for 80_20 split: {xgbtest_rmse:.2f}')
xgbtest_r2 = r2_score(y_test_80_20, y_test_pred)
print(f'Test R2 for 80_20 split: {xgbtest_r2:.2f}')

```

Rajah 7 Kod latihan dan ujian model XGBoost dengan setiap nisbah pembahagian data


```

# Train the model on the training set for 80,20
modelxgb.fit(X_train_70_30, y_train_70_30)

y_val_pred = modelxgb.predict(X_val_70_30)
xgbval_rmse = np.sqrt(mean_squared_error(y_val_70_30, y_val_pred))
xgbval_r2 = r2_score(y_val_70_30, y_val_pred)
print()
print(f'Validation RMSE for 70_30 split: {xgbval_rmse:.2f}')
print(f'Validation R2 for 70_30 split: {xgbval_r2:.2f}')

# Evaluate the model on the test set
y_test_pred = modelxgb.predict(X_test_70_30)
xgbtest_rmse = np.sqrt(mean_squared_error(y_test_70_30, y_test_pred))
print(f'Test RMSE for 70_30 split: {xgbtest_rmse:.2f}')
xgbtest_r2 = r2_score(y_test_70_30, y_test_pred)
print(f'Test R2 for 70_30 split: {xgbtest_r2:.2f}')

# Train the model on the training set for 80,20
modelxgb.fit(X_train_60_40, y_train_60_40)

y_val_pred = modelxgb.predict(X_val_60_40)
xgbval_rmse = np.sqrt(mean_squared_error(y_val_60_40, y_val_pred))
xgbval_r2 = r2_score(y_val_60_40, y_val_pred)
print()
print(f'Validation RMSE for 60_40 split: {xgbval_rmse:.2f}')
print(f'Validation R2 for 60_40 split: {xgbval_r2:.2f}')

# Evaluate the model on the test set
y_test_pred = modelxgb.predict(X_test_60_40)
xgbtest_rmse = np.sqrt(mean_squared_error(y_test_60_40, y_test_pred))
print(f'Test MSE for 60_40 split: {xgbtest_rmse:.2f}')
xgbtest_r2 = r2_score(y_test_60_40, y_test_pred)
print(f'Test R2 for 60_40 split: {xgbtest_r2:.2f}')

```

Rajah 8 Kod latihan dan ujian model XGBoost dengan setiap nisbah pembahagian data

c. Pembangunan model *Linear Regression*

Linear Regression ialah kaedah statistik yang digunakan untuk mewujudkan hubungan antara pembolehubah peramal dan pembolehubah hasil, dengan tujuan untuk meramalkan hasil berdasarkan nilai peramalan. Tafsiran model melibatkan pemeriksaan pekali untuk memahami kesan setiap pembolehubah peramal. Pekali positif menunjukkan perkaitan positif dengan tempoh tinggal ICU, manakala pekali negatif mencadangkan perkaitan negatif. Persamaan *Linear Regression* boleh digunakan untuk membuat ramalan bagi data baharu.

```

from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score

lr_model = LinearRegression()

# Train the model on the training set for 80,20
lr_model.fit(X_train_90_10, y_train_90_10)

y_val_pred = lr_model.predict(X_val_90_10)
lrval_rmse = np.sqrt(mean_squared_error(y_val_90_10, y_val_pred))
lrval_r2 = r2_score(y_val_90_10, y_val_pred)
print(f'Validation RMSE for 90_10 split: {lrval_rmse:.2f}')
print(f'Validation R2 for 90_10 split: {lrval_r2:.2f}')

# Evaluate the model on the test set
y_test_pred = lr_model.predict(X_test_90_10)
lrtest_rmse = np.sqrt(mean_squared_error(y_test_90_10, y_test_pred))
print(f'Test RMSE for 90_10 split: {lrtest_rmse:.2f}')
lrtest_r2 = r2_score(y_test_90_10, y_test_pred)
print(f'Test R2 for 90_10 split: {lrtest_r2:.2f}')

# Train the model on the training set for 80,20
lr_model.fit(X_train_80_20, y_train_80_20)

y_val_pred = lr_model.predict(X_val_80_20)
lrval_rmse = np.sqrt(mean_squared_error(y_val_80_20, y_val_pred))
lrval_r2 = r2_score(y_val_80_20, y_val_pred)
print()
print(f'Validation RMSE for 80_20 split: {lrval_rmse:.2f}')
print(f'Validation R2 for 80_20 split: {lrval_r2:.2f}')

# Evaluate the model on the test set
y_test_pred = lr_model.predict(X_test_80_20)
lrtest_rmse = np.sqrt(mean_squared_error(y_test_80_20, y_test_pred))
print(f'Test RMSE for 80_20 split: {lrtest_rmse:.2f}')
lrtest_r2 = r2_score(y_test_80_20, y_test_pred)
print(f'Test R2 for 80_20 split: {lrtest_r2:.2f}')

```

Rajah 9 Kod latihan dan ujian model Linear Regression dengan setiap nisbah pembahagian data

```

# Train the model on the training set for 80,20
lr_model.fit(X_train_70_30, y_train_70_30)

y_val_pred = lr_model.predict(X_val_70_30)
lrval_rmse = np.sqrt(mean_squared_error(y_val_70_30, y_val_pred))
lrval_r2 = r2_score(y_val_70_30, y_val_pred)
print()
print(f'Validation RMSE for 70_30 split: {lrval_rmse:.2f}')
print(f'Validation R2 for 70_30 split: {lrval_r2:.2f}')

# Evaluate the model on the test set
y_test_pred = lr_model.predict(X_test_70_30)
lrtest_rmse = np.sqrt(mean_squared_error(y_test_70_30, y_test_pred))
print(f'Test RMSE for 70_30 split: {lrtest_rmse:.2f}')
lrtest_r2 = r2_score(y_test_70_30, y_test_pred)
print(f'Test R2 for 70_30 split: {lrtest_r2:.2f}')

# Train the model on the training set for 80,20
lr_model.fit(X_train_60_40, y_train_60_40)

y_val_pred = lr_model.predict(X_val_60_40)
lrval_rmse = np.sqrt(mean_squared_error(y_val_60_40, y_val_pred))
lrval_r2 = r2_score(y_val_60_40, y_val_pred)
print()
print(f'Validation RMSE for 60_40 split: {lrval_rmse:.2f}')
print(f'Validation R2 for 60_40 split: {lrval_r2:.2f}')

# Evaluate the model on the test set
y_test_pred = lr_model.predict(X_test_60_40)
lrtest_rmse = np.sqrt(mean_squared_error(y_test_60_40, y_test_pred))
print(f'Test RMSE for 60_40 split: {lrtest_rmse:.2f}')
lrtest_r2 = r2_score(y_test_60_40, y_test_pred)
print(f'Test R2 for 60_40 split: {lrtest_r2:.2f}')

```

Rajah 10 Kod latihan dan ujian model Linear Regression dengan setiap nisbah pembahagian data

Penalaan Hiperparameter

Penalaan hiperparameter bertujuan untuk mencari set hiperparameter yang ideal untuk model pembelajaran mesin untuk memaksimumkan prestasinya. Hiperparameter, seperti kadar pembelajaran dalam keturunan kecerunan, bilangan pokok dalam hutan rawak, atau kedalaman rangkaian saraf, ialah tetapan konfigurasi yang mengawal proses pembelajaran. Hiperparameter ini adalah tetap dan mempunyai kesan yang ketara terhadap ketepatan dan kecekapan model, berbeza dengan parameter model, yang ditentukan oleh data semasa latihan.

Terdapat dua pendekatan biasa untuk penalaan hiperparameter: *GridSearchCV* dan *RandomizedSearchCV*. Di dalam projek ini, *RandomizedSearchCV* digunakan kerana ia memerlukan lebih sedikit masa untuk dijalankan berbanding *GridSearchCV*, menjadikannya pilihan yang lebih praktikal apabila berurusan dengan set data besar atau model kompleks dengan banyak hyperparameter.

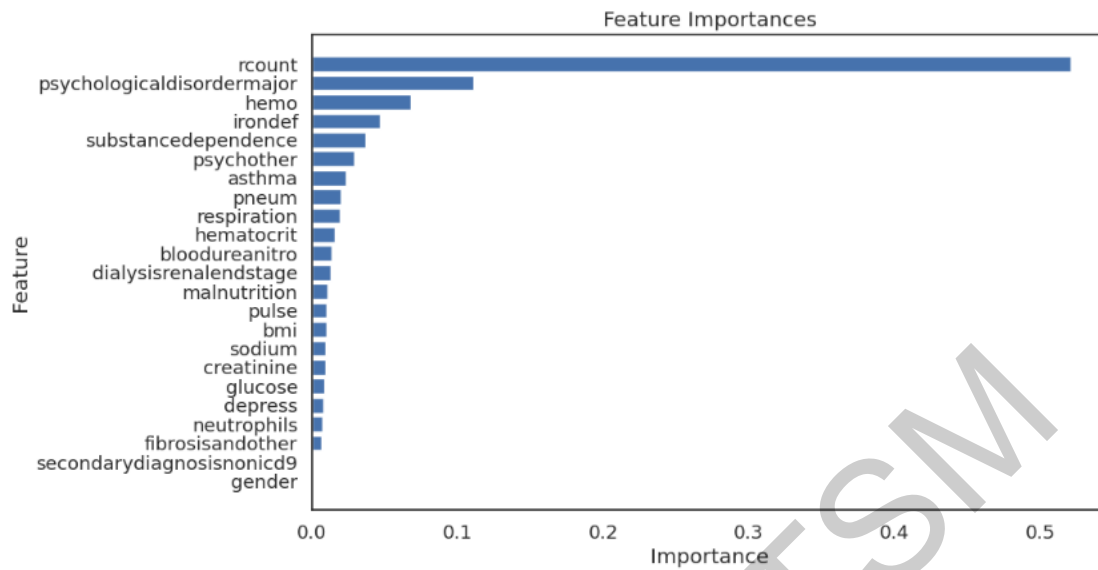
Jadual 1 Parameter yang terpilih daripada *RandomizeSearchCV*

Hiperparameter	Nilai
<i>Subsample</i>	1.0
<i>Max_depth</i>	200
<i>Learning_rate</i>	6
<i>Colsample_bytree</i>	0.8

Model yang terpilih dilatih menggunakan set data gabungan yang terdiri daripada set latihan dan pengesahan untuk memaksimumkan jumlah data yang tersedia untuk pembelajaran, setelah *RandomizedSearchCV* telah menentukan gabungan optimum hiperparameter. *Subsample=1.0*, *n_estimators=200*, *max_depth=6*, *learning_rate=0.2*, dan *colsample_bytree=0.8* ialah hiperparameter yang dipilih. Model ini dijamin mempunyai pemahaman yang paling teliti tentang corak data berkat latihan yang meluas ini.

Pemilihan Fitur Penting

Pemilihan atribut dalam model terbaik melibatkan pengekstrakan kepentingan ciri menggunakan kaedah seperti `'feature_importances_'` atau `'get_booster().get_score()'`. Kaedah ini menilai ciri berdasarkan kesannya pada ramalan, membimbing keputusan untuk mengutamakan ciri berkepentingan tinggi, berpotensi menghapuskan ciri yang kurang penting dan menumpukan pada memperhalusi ciri yang sederhana penting. Memvisualisasikan kepentingan ciri dengan alatan seperti carta bar meningkatkan pemahaman dan membantu dalam tafsiran dan pengoptimuman model, memastikan prestasi dan kebolehtafsiran dimaksimumkan. Di dalam fasa ini juga, kaedah SHAP analisis akan digunakan untuk proses pemilihan fitur.



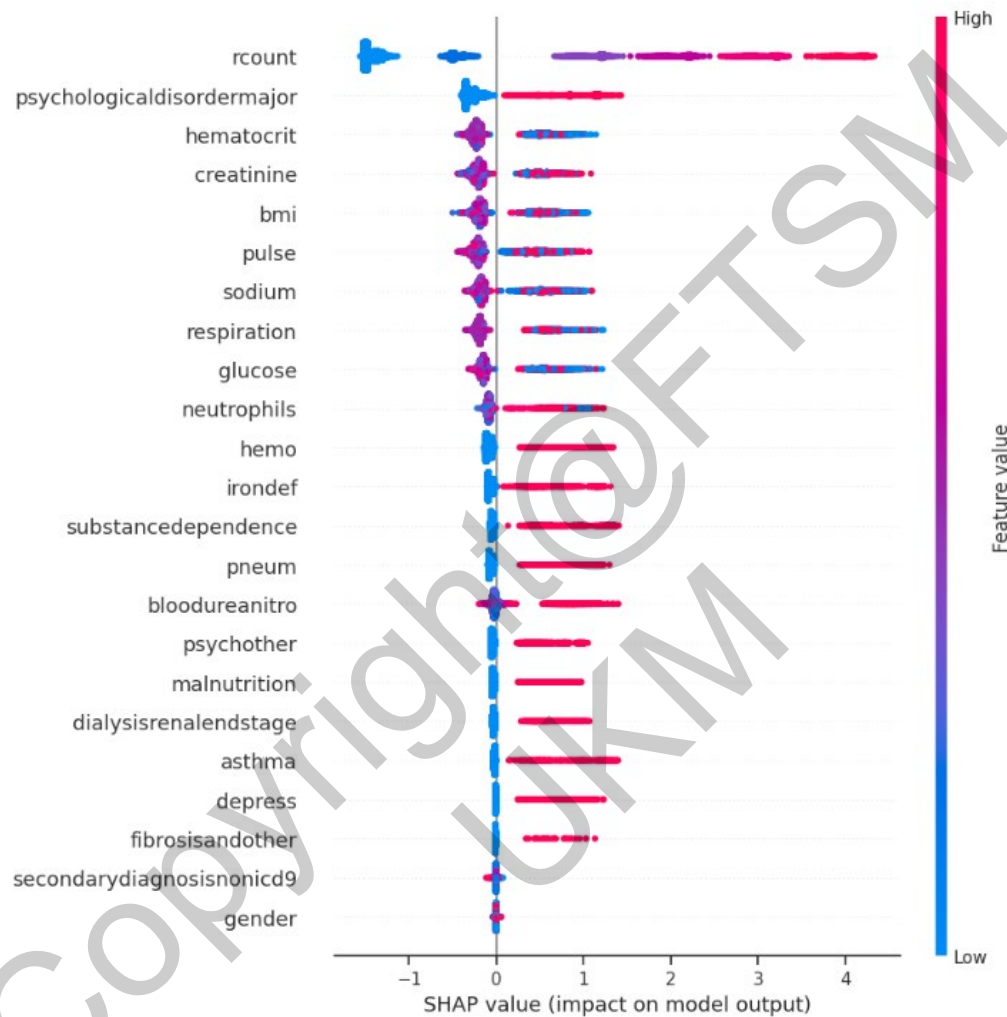
Rajah 11 Carta Bar Susunan Fitur Menggunakan Model Ramalan Terbaik

Jadual 2 Setiap fitur dan berat penting mengikut susunan paling berat ke ringan

Fitur	Berat Kepentingan
rcount	0.520514
psychologicaldisordermajor	0.111332
hemo	0.068214
irondef	0.046613
substancedependence	0.036687
psychother	0.028918
asthma	0.023311
pneum	0.019930
respiration	0.019462
hematocrit	0.015761
bloodureanitro	0.013674
dialysisrenalendstage	0.013250
malnutrition	0.010608
pulse	0.010489
bmi	0.010433
sodium	0.009861
creatinine	0.009563
glucose	0.008470
depress	0.008234
neutrophils	0.007482
fibrosisandother	0.006349
secondarydiagnosisnonicd9	0.000501
gender	0.000344

a. Visualisasi fitur penting menggunakan SHAP analisis

SHAP (SHapley Additive exPlanations) ialah alat yang berkuasa untuk mendedahkan pengaruh fitur individu pada ramalan model. Ia memberikan nilai kepada setiap fitur, menunjukkan berapa banyak ia menyumbang kepada output model. (Malladi, 2023). Analisis SHAP bukan sahaja membantu memahami kepentingan fitur, tetapi juga memberikan pandangan yang jelas tentang bagaimana setiap fitur mempengaruhi ramalan individu, meningkatkan kebolehtafsiran dan kebolehpercayaan model pembelajaran mesin.



Rajah 12 Jadual SHAP Analisis

Plot yang dijana daripada analisis SHAP menawarkan pandangan bernuansa bagaimana setiap ciri mempengaruhi ramalan model. Ia secara berkesan menyusun ciri mengikut kepentingan, dengan rcount memberikan impak yang paling ketara dan jantina paling sedikit. Setiap titik pada plot mewakili nilai SHAP sebagai contoh, menggambarkan sama ada nilai ciri meningkat (titik merah) atau menurun (titik biru) ramalan model. Kedudukan paksi-x plot menunjukkan magnitud kesan ini, manakala warna menunjukkan julat nilai ciri. Kebolehubahan sepanjang paksi-x mendedahkan bahawa ciri-ciri tertentu, seperti kreatinin, mempunyai kesan yang tidak konsisten merentas kejadian yang berbeza. Kesan simetri, diperhatikan dalam ciri seperti nadi dan natrium, mempengaruhi ramalan dalam kedua-dua arah positif dan negatif, manakala ciri seperti rcount mempamerkan pengaruh yang lebih terarah.

Cerapan daripada plot hangat ini meningkatkan pemahaman tentang kepentingan keseluruhan dan sumbangan bernuansa setiap ciri kepada keupayaan ramalan model.

Secondary diagnosis nononicd9 dan *gender* mempunyai nilai SHAP dalam rajah yang berkelompok rapat sekitar sifar, menunjukkan bahawa perbezaan dalam fitur-fitur ini mempunyai sedikit kaitan dengan jangka masa tinggal ICU diramalkan. Kedua-dua fitur ini telah dikecualikan daripada fitur terpilih. Fitur impak minimum mungkin menambah hingar tanpa menawarkan maklumat berguna, mungkin merumitkan model tanpa meningkatkan prestasinya.

Fasa Pembangunan Aplikasi Berasaskan Web ICUForesee

Di dalam fasa pembangunan aplikasi web ICUForesee, model ramalan terbaik akan diintegrasikan ke dalam ICUForesee untuk meramalkan tempoh tinggal pesakit di ICU berdasarkan data pesakit. ICUForesee ini dibangunkan menggunakan Python dan memanfaatkan rangka kerja Streamlit yang menyediakan pengalaman pengguna yang intuitif merentas setiap halaman web ICUForesee.

a. Halaman utama aplikasi web ICUForesee



Rajah 13 Menunjukkan halaman utama bagi web ICUForesee

Berdasarkan rajah 13 di atas terdapat dua interaktif komponen di halaman utama ICUForesee iaitu yang pertama, butang yang akan menavigasi pengguna ke halaman untuk melakukan ramalan tempoh tinggal pesakit. Kemudian, yang kedua, butang yang menavigasi pengguna ke halaman panduan penggunaan web ICUForesee ini.

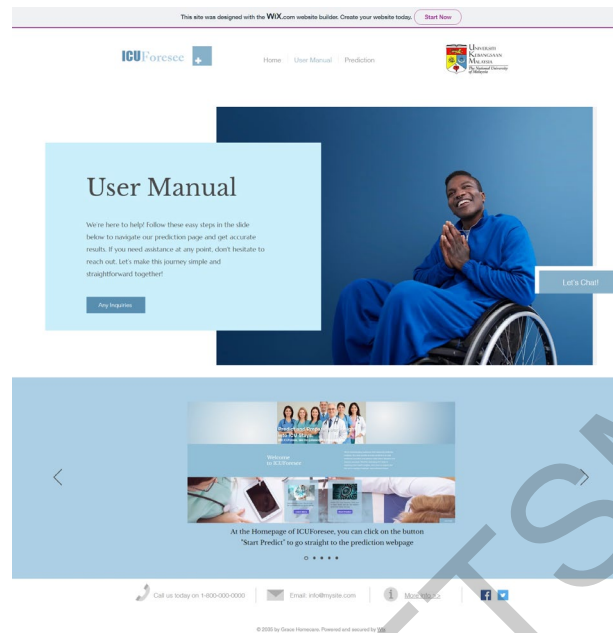
b. Halaman ramalan tempoh tinggal pesakit di ICU

Rajah 14 Menunjukkan halaman untuk mendapatkan ramalan tempoh tinggal pesakit di ICU

Rajah 15 Menunjukkan halaman untuk mendapatkan ramalan tempoh tinggal pesakit di ICU

Rajah 14 dan 15 di atas menunjukkan halaman untuk pengguna mendapatkan ramalan tempoh tinggal pesakit di ICU. Dengan memasukkan setiap kondisi yang diperlukan dan seterusnya menekan butang 'Predict' untuk menghantar maklumat kondisi pesakit ke *back-end* sistem untuk melakukan proses ramalan. Keputusan ramalan tempoh tinggal pesakit dan tarikh jangkaan keluar pesakit tersebut akan dipaparkan. Pengguna boleh pulang semula ke halaman utama dengan menekan butang 'Back' yang terletak di bahagian atas halaman ramalan.

c. Halaman panduan pengguna



Rajah 15 Menunjukkan halaman manual fungsi penggunaan setiap fitur di web ICUForesee

Di dalam rajah 15 di atas menunjukkan halaman panduan pengguna web ICUForesee. Ia menunjukkan langkah-langkah yang jelas dan teratur bagi pengguna untuk mudah memahami proses dalam melakukan ramalan tempoh tinggal pesakit di ICU.

d. Pembangunan kod laman web ICUForesee

```

139 # Streamlit app
140 st.markdown("<h1 class='title'>Length of ICU Stay Prediction</h1>", unsafe_allow_html=True)
141 # Center content
142 with st.container():
143     st.markdown("<div class='centered'>", unsafe_allow_html=True)
144
145     # Input fields for features
146     rcount = st.number_input("Number of Readmission Patient to ICU", min_value=0, key='rcount', help="Number of times the patient was readmitted")
147
148     # Psychological Disorder Major
149     psychologicaldisordermajor = st.radio(
150         "Psychological Disorder Major",
151         (0, 1),
152         index=0,
153         help="Presence of a major psychological disorder during encounter (0 - None, 1 - Presence)"
154     )
155     # Iron Deficiency
156     iron_def = st.radio(
157         "Iron Deficiency",
158         (0, 1),
159         index=0,
160         help="Presence of iron deficiency during encounter (0 - None, 1 - Presence)"
161     )
162     pneum = st.radio(
163         "Pneumonia",
164         (0, 1),
165         index=0,
166         help="Presence of Pneumonia during encounter (0 - None, 1 - Presence)"
167     )
168     substancdependence = st.radio(
169         "Substance Dependence",
170         (0, 1),
171         index=0,
172         help="Presence of Substance Dependence during encounter (0 - None, 1 - Presence)"
173     )
174     asthma = st.radio(
175         "Asthma",
176         (0, 1),
177         index=0,
178         help="Presence of Asthma during encounter (0 - None, 1 - Presence)"
179     )

```

Rajah 16 Kod pembangunan halaman ramalan tempoh tinggal pesakit di ICU


```

180 psychother = st.radio(
181     "Other Psychological Disorder",
182     (0, 1),
183     index=0,
184     help="Presence of Other Psychological Disorder during encounter (0 - None, 1 - Presence)"
185 )
186 malnutrition = st.radio(
187     "Malnutrition",
188     (0, 1),
189     index=0,
190     help="Presence of Malnutrition during encounter (0 - None, 1 - Presence)"
191 )
192 depress = st.radio(
193     "Depression",
194     (0, 1),
195     index=0,
196     help="Presence of Depression during encounter (0 - None, 1 - Presence)"
197 )
198 fibrosisandother = st.radio(
199     "Fibrosis",
200     (0, 1),
201     index=0,
202     help="Presence of Fibrosis during encounter (0 - None, 1 - Presence)"
203 )
204 dialysisrenalendstage = st.radio(
205     "Renal Disease",
206     (0, 1),
207     index=0,
208     help="Presence of Renal Disease during encounter (0 - None, 1 - Presence)"
209 )
210 hemo = st.radio(
211     "Blood Disorder",
212     (0, 1),
213     index=0,
214     help="Presence of Blood Disorder during encounter (0 - None, 1 - Presence)"
215 )
216 pulse = st.number_input("Average Pulse Rate", min_value=0, help="Average pulse during encounter (beats/m)")
217 respiration = st.number_input("Average Respiration", min_value=0.0, help="Average respiration during encounter (breaths/m)")
218 bloodureanitro = st.number_input("Average Blood Urea Nitrogen", min_value=0.0, help="Average blood urea nitrogen value during encounter (mg/dL)")
219 neutrophils = st.number_input("Average Neutrophils Value", min_value=0.0, help="Average neutrophils value during encounter (cells/microl)")
220 hematocrit = st.number_input("Average Hematocrit Value", min_value=0.0, help="Average hematocrit value during encounter (g/dL)")
221 bmi = st.number_input("Average Bmi", min_value=0.0, help="Average BMI during encounter (kg/m2)")
222 sodium = st.number_input("Average Sodium Value", min_value=0.0, help="Average sodium value during encounter (mmol/L)")
223 glucose = st.number_input("Average Glucose Value", min_value=0.0, help="Average glucose value during encounter (mmol/L)")
224 creatinine = st.number_input("Average Creatinine Value", min_value=0.0, help="Average creatinine value during encounter (mg/dL)")
225
226 # Input field for admission start date
227 start_date = st.date_input("Admission Start Date")
228
229 # Initialize variables to store prediction results
230 rounded_prediction = None
231 discharge_date = None
232 rmse_all = None
233

```

Rajah 17 Kod pembangunan halaman ramalan tempoh tinggal pesakit di ICU

```

234 # Predict button
235 if st.button("Predict"):
236     # Validation to ensure no input is zero
237     if (pulse == 0 or respiration == 0.0 or bloodureanitro == 0.0 or neutrophils == 0.0 or
238         hematocrit == 0.0 or bmi == 0.0 or sodium == 0.0 or glucose == 0.0 or creatinine == 0.0):
239         st.error("Please enter non-zero values for all numeric inputs based on its specific range for each parameter.")
240     else:
241         # Create a DataFrame for the input features
242         input_data = pd.DataFrame([[rcount, psychologicaldisordermajor, hemo, iron_def, pneum, respiration, bloodureanitro, substancedependence,asthma,
243             psychother, malnutrition, neutrophils, hematocrit, bmi, sodium, glucose, dialysisrenalendstage, depress, fibrosisandother,
244             creatinine, pulse]], columns=features)
245         input_data_scaled = input_data
246
247         # Predict length of stay
248         prediction = model.predict(input_data_scaled)[0]
249
250         # Round prediction to nearest integer
251         rounded_prediction = cell(prediction)
252
253         # Calculate discharge date
254         discharge_date = start_date + timedelta(days=rounded_prediction)
255
256         # Update the original DataFrame with the new patient data
257         new_patient_df = input_data.copy()
258         new_patient_df['lengthofstay'] = rounded_prediction
259         combined_df = pd.concat([df, new_patient_df], ignore_index=True)
260
261         # Separate features and target variable from the combined DataFrame
262         X_combined = combined_df[features]
263         y_combined = combined_df['lengthofstay']
264
265         # Recalculate RMSE with the new patient data
266         rmse_all = np.sqrt(mean_squared_error(y_combined, model.predict(X_combined)))
267
268         # Display the prediction results
269         if rounded_prediction is not None and discharge_date is not None:
270             st.markdown(
271                 """
272                 <div class="output-box">
273                 <h3>Predicted Length of Stay: {rounded_prediction} days</h3>
274                 <p>Expected Discharge Date: {discharge_date.strftime('%Y-%m-%d')}</p>
275                 <p>Root Mean Squared Error (RMSE) for all data: {rmse_all:.6f}</p>
276                 </div>
277                 """,
278                 unsafe_allow_html=True
279             )
280             st.markdown('</div>', unsafe_allow_html=True)

```

Rajah 18 Kod pembangunan halaman ramalan tempoh tinggal pesakit di ICU

Fasa Pengujian Aplikasi Web ICUForesee

Di dalam fasa ini, maklum balas telah dikumpulkan melalui *Google Form* yang direka untuk menilai pelbagai aspek tapak web, termasuk navigasi, kejelasan arahan, kemudahan input data, ketepatan ramalan, kefungsian dan pengalaman pengguna keseluruhan. Analisis ini memberikan pandangan berharga tentang kepuasan pengguna, menyerlahkan bidang untuk penambahbaikan dan membimbing peningkatan masa depan untuk memenuhi keperluan pengguna kami dengan lebih baik.

Data maklum balas pengguna dianalisis melalui kaedah analisis data yang bernama statistik deskriptif dengan menggunakan skor min bagi setiap aspek. Jadual 3 menunjukkan Tafsiran Skala Skor Min.

Jadual 3 Tafsiran Skala Skor Min

Skor Min	Tafsiran
1.00 – 2.32	Rendah
2.33 – 3.65	Sederhana
3.66 – 5.00	Tinggi

KEPUTUSAN DAN PERBINCANGAN

Di dalam projek ini, terdapat tiga jenis pengujian yang dilakukan iaitu pengujian pengesahan model, pengujian bagi pemilihan fitur, dan pengujian aplikasi berasaskan web ICUForesee. Pengujian pengesahan model dilakukan untuk menguji keupayaan setiap model dalam meramalkan tepoh tinggal pesakit di ICU berdasarkan metrik penilaian ralat purata kuasa dua (RMSE) dan pekali penentuan R^2 . Berdasarkan pengujian model ini, model ramalan terbaik akan dipilih, dan seterusnya akan menjalani penalaan hiperparameter. Hasil daripada penalaan hiperparameter tersebut, perbandingan antara model terbaik dan model kedua terbaik akan dilakukan untuk melihat perbezaan prestasi yang signifikan secara statistik ke atas dua model tersebut. Bagi pengujian pemilihan fitur tersebut akan dilakukan dengan melihat perbandingan semua fitur dan fitur terpilih menggunakan metrik penilaian RMSE dan R^2 , untuk menilai keberkesanan antara semua fitur dan fitur terpilih terhadap prestasi model dalam melakukan ramalan.

Penilaian model ramalan

Proses penilaian bermula dengan pengiraan R Kuasa Dua (R^2) dan ralat min kuasa dua akar (RMSE) untuk tiga model: Hutan Rawak, XGBoost dan Regresi Linear. Selepas itu, penalaan hiperparameter dijalankan pada model terbaik menggunakan set data yang sama untuk mengoptimumkan prestasi. Akhir sekali, ujian T-Test dan P-value membandingkan model berprestasi tinggi, memastikan pengesahan yang rapi dan meningkatkan ketepatan ramalan.

a. R kuasa dua (R^2)

Jadual 3 Perbandingan R kuasa dua bagi set pengesahan

Nisbah	RF	XGBoost	Linear Regression
90:05:05	0.92	0.96	0.75

80:10:10	0.92	0.96	0.75
70:20:10	0.92	0.96	0.75
60:20:20	0.91	0.96	0.74

Jadual 4 Perbandingan R kuasa dua bagi set pengujian

Nisbah	RF	XGBoost	Linear Regression
90:05:05	0.93	0.96	0.76
80:10:10	0.92	0.96	0.74
70:20:10	0.92	0.96	0.74
60:20:20	0.92	0.96	0.74

Berdasarkan jadual 3 dan 4, XGBoost secara konsisten mengatasi Hutan Rawak dan Regresi Linear dalam meramalkan pembolehubah sasaran. XGBoost mencapai nilai R^2 yang tinggi antara 0.95 hingga 0.96 merentas set data yang berbeza, menunjukkan ketepatan ramalan yang kukuh dan generalisasi yang mantap kepada data baharu. Random Forest juga menunjukkan prestasi yang baik dengan skor R^2 secara konsisten sekitar 0.92 hingga 0.93 pada set ujian, walaupun kurang berkesan sedikit daripada XGBoost. Sebaliknya, Regresi Linear menunjukkan nilai R^2 yang lebih rendah daripada 0.74 hingga 0.76, menunjukkan prestasi yang lebih lemah dalam menangkap varians data. Oleh itu, XGBoost dikenal pasti sebagai model berprestasi tinggi untuk ramalan yang tepat, diikuti oleh Random Forest, manakala Regresi Linear kurang sesuai kerana prestasinya yang lebih rendah.

b. Ralat purata kuasa dua (RMSE)

Jadual 5 Perbandingan ralat kuasa dua (RMSE) bagi set pengesahan

Nisbah	RF	XGBoost	Linear Regression
90:05:05	0.65	0.46	1.18
80:10:10	0.66	0.47	1.18
70:20:10	0.68	0.47	1.18
60:20:20	0.68	0.47	1.18

Jadual 6 Perbandingan ralat kuasa dua (RMSE) bagi set pengujian

Nisbah	RF	XGBoost	Linear Regression
90:05:05	0.64	0.46	1.15
80:10:10	0.65	0.46	1.19
70:20:10	0.66	0.46	1.19
60:20:20	0.67	0.47	1.19

Berdasarkan jadual 5 dan 6, XGBoost secara konsisten menunjukkan nilai RMSE terendah

merentas semua senario ujian, antara 0.46 hingga 0.47. Ini menunjukkan ketepatan ramalan yang unggul berbanding model Random Forest (RF) dan Regresi Linear. RF menunjukkan prestasi yang cukup baik dengan skor RMSE antara 0.64 hingga 0.68 pada set ujian tetapi kurang konsisten daripada XGBoost. Sebaliknya, Regresi Linear secara konsisten mempamerkan nilai RMSE yang lebih tinggi antara 1.15 hingga 1.19, menunjukkan prestasi ramalan yang lebih lemah. Oleh itu, XGBoost muncul sebagai model yang paling berkesan untuk meminimumkan ralat ramalan, diikuti oleh RF, manakala Regresi Linear menunjukkan ketepatan paling sedikit dalam meramalkan pembolehubah sasaran dalam kajian ini.

c. Penalaan hiperparameter

Berdasarkan pengujian dan penilaian ralat purata kuasa dua (RMSE) dan R kuasa dua, kita boleh mendapati bahawa model XGBoost merupakan model terbaik berbanding model *Random Forest* dan *Linear Regression*. Oleh itu, penalaan hiperparameter akan dilakukan ke atas model XGBoost.

Jadual 7 Nilai prestasi model terpilih (XGBoost) selepas proses penalaan hiperparameter

Metrik Prestasi	Nilai
RMSE	0.43
R kuasa dua (R^2)	0.97

Di jadual 7, kita dapati prestasi model terpilih iaitu XGBoost, menjadi lebih baik. RMSE=0.43 menunjukkan bahawa ramalan model, secara purata, 0.43-unit dari nilai sebenar, manakala $R^2=0.97$ menandakan bahawa model menerangkan 97% varians dalam data, mempamerkan ketepatan ramalan yang sangat baik.

d. Pengujian T-Test dan P-value ke atas dua model terbaik

Jadual 8 Nilai ujian T-Test dan P-value bagi melihat perbezaan ketara antara Hutan Rawak dan XGBoost

Metrik Prestasi	Nilai
Ujian T (<i>T-Test</i>)	-74.29461735674322
Nilai P (<i>P-value</i>)	1.966968892592749e-07

Melalui ujian ini, pengkaji dapati statistik t -74.29 dan nilai p 1.97e-07 berdasarkan keputusan ujian-t berpasangan yang membandingkan ralat min kuasa dua (MSE) bagi model Hutan Rawak dan XGBoost sepanjang 5 kali ganda silang- pengesahan. Terdapat perbezaan yang signifikan secara statistik antara prestasi kedua-dua model, seperti yang ditunjukkan oleh nilai-p, yang jauh lebih kecil daripada tahap keertian konvensional 0.05. Oleh itu, kita boleh membuat kesimpulan bahawa perbezaan prestasi yang diperhatikan antara model XGBoost dan *Random Forest* tidak mungkin disebabkan oleh peluang rawak, dan satu model adalah lebih ketara mengatasi yang lain.

Pengujian Pemilihan Fitur

Jadual 9 Perbandingan RMSE dan R^2 bagi set data pengesahan semua fitur dan fitur terpilih

Fitur	XGBoost	
	RMSE	R2
Semua Fitur	0.44	0.97
Fitur Terpilih (<i>rcount, psychologicaldisordermajor, hematocrit, creatinine, bmi, pulse, sodium, respiration, glucose, neutrophils, hemo, irondef, substancedependence, pneum, bloodureanitro, pyschother, malnutrition, dialysisrenalendstage, asthma, depress, fibrosisandother</i>)	0.37	0.98

Jadual 10 Perbandingan RMSE dan R^2 bagi set data pengujian semua fitur dan fitur terpilih

Fitur	XGBoost	
	RMSE	R2
Semua Fitur	0.44	0.96
Fitur Terpilih (<i>rcount, psychologicaldisordermajor, hematocrit, creatinine, bmi, pulse, sodium, respiration, glucose, neutrophils, hemo, irondef, substancedependence, pneum, bloodureanitro, pyschother, malnutrition, dialysisrenalendstage, asthma, depress, fibrosisandother</i>)	0.44	0.96

Berdasarkan kedua-dua jadual perbandingan RMSE dan R^2 , bersama-sama dengan keputusan ujian-T dan nilai-P untuk model XGBoost terbaik, ciri terpilih meningkatkan prestasi ramalan dengan ketara berbanding menggunakan semua ciri. Jadual 9 menunjukkan bahawa dengan menggunakan semua ciri menghasilkan RMSE sebanyak 0.44 dan R^2 sebanyak 0.97, menunjukkan ralat ramalan yang lebih tinggi dan kebolehpercayaan model yang lebih rendah sedikit. Sebaliknya, Jadual 4.14 menunjukkan bahawa menggunakan atribut terpilih (*rcount, gangguan psikologi utama, hematokrit, kreatinin, bmi, nadi, natrium, pernafasan, glukosa, neutrofil, hemo, irondef, pergantungan bahan, pneum, bloodureanitro, pyschother, malnutrisi, dialisis renalendstage, asma, kemurungan, fibrosis dan lain-lain*) menghasilkan RMSE yang lebih baik sebanyak 0.37 dan R^2 yang lebih tinggi sebanyak 0.98. Ini menekankan bahawa pemilihan ciri meningkatkan prestasi model dengan mengurangkan ralat ramalan dan meningkatkan kebolehpercayaan. Oleh itu, memfokuskan pada ciri terpilih dan bukannya

menggunakan semua ciri yang ada terbukti bermanfaat untuk mengoptimumkan ketepatan dan kecekapan model XGBoost dalam membuat ramalan.

Pengujian Aplikasi Berasaskan Web ICUForesee

Jadual 11 Skor Min Maklum Balas Laman Web ICUForesee

No	Soalan	Min
1	How easy was it to navigate the Home Page?	4.62
2	How clear were the instructions provided on the Instruction Page?	4.46
3	How easy was it to input the patient's condition and features on the Prediction Page?	4.38
4	How accurate do you feel the prediction was for the length of ICU stay?	4.07
5	Did the website load quickly and function smoothly?	5.00
6	How likely are you to recommend ICUForesee to a colleague or friend?	4.38
7	How would you rate your overall experience with ICUForesee?	4.46

Berdasarkan julat pemarkahan, jadual 11 menunjukkan maklum balas pengguna yang sangat positif merentasi pelbagai aspek aplikasi web ICUForesee. Pengguna mendapati menavigasi Halaman Utama (4.62), memahami arahan pada Halaman Arahan (4.46), dan memasukkan data pesakit pada Halaman Ramalan (4.38) adalah sangat mudah, semuanya mendapat markah dalam julat tafsiran "Tinggi". Selain itu, responden berpendapat ketepatan ramalan tinggal ICU (4.07) sebagai agak tinggi. Mereka sebulat suara melaporkan bahawa laman web dimuatkan dengan cepat dan berfungsi dengan lancar (5.00), menunjukkan prestasi cemerlang. Pengguna juga berkemungkinan besar untuk mengesyorkan ICUForesee kepada orang lain (4.38) dan menilai pengalaman keseluruhan mereka (4.46) dengan sangat positif. Secara keseluruhannya, markah ini menggambarkan kepuasan pengguna yang kukuh dengan aplikasi ICUForesee, mengesahkan kebolegunaan, ketepatan dan keberkesanan keseluruhannya dalam meramalkan tempoh penginapan ICU.

Jadual 12 Maklum balas Laman Web ICUForesee

No	Soalan	Maklum balas
1	<i>Were there any instructions that you found confusing or unclear? If so, please specify.</i>	<i>Some medical terms</i>
2	<i>Did you encounter any issues while entering the patient's data? If yes, please describe.</i>	<i>Minor lag when to predict Some of the data needed use unfamiliar term for normal person.</i>
3	<i>How long did it take the user to complete the task (Prediction Length of ICU Stay)?</i>	<i>3 minutes 4 minutes 5 minutes. 8 minutes.</i>

		<p><i>About 5 to 7 minutes.</i></p> <p><i>Around 2 minutes.</i></p> <p><i>Less than 1 minutes</i></p> <p><i>Maybe in 3 minutes</i></p> <p><i>Only a few minutes</i></p> <p><i>Under 2 minutes</i></p>
4	<p><i>Were there any features or functionalities that you found missing or would like to see added?</i></p>	<p><i>Nope.</i></p> <p><i>No.</i></p> <p><i>Maybe can add more details about the company/'About Us'</i></p> <p><i>No.</i></p> <p><i>Maybe you can add some placeholder for certain input box.</i></p> <p><i>More visual aids for me to understand.</i></p> <p><i>I wish I can use the website in the dark mode.</i></p> <p><i>Detailed error message.</i></p> <p><i>I think it would be good if there's an option to save patient data.</i></p> <p><i>For me, it is better to have description of every medical term used in collecting the user health data.</i></p>
5	<p><i>Do you have any additional comments or suggestions to improve ICUForesee?</i></p>	<p><i>In my laptop the font colour is white and the background is light gray, so it makes it hard for me to kick-in the input and see the output results.</i></p> <p><i>Maybe you can make user to store their data prediction, so they dont need to fill again and again whenever they want to see back.</i></p> <p><i>Very user-friendly</i></p> <p><i>Very helpful and efficient</i> 🇲🇾</p> <p><i>Make it support for both dark mode and light mode</i></p>

Cadangan Penambahbaikan

Selepas menjalankan kajian yang menyeluruh, beberapa penambahbaikan utama disyorkan untuk memastikan ICUForesee kekal berdaya saing dalam industri. Keutamaan utama termasuk menjalin kerjasama dengan pelbagai hospital untuk meningkatkan pengumpulan

data, merangkumi julat faktor kritikal yang lebih luas yang boleh meningkatkan ketepatan dan kesesuaian model. Selain itu, menjalankan kajian pengesahan luaran menggunakan set data yang pelbagai daripada populasi yang pelbagai dan tetapan penjagaan kesihatan akan mengesahkan dan meningkatkan kebolehgneralisasian model. Tambahan pula, membangunkan model yang disesuaikan untuk subkumpulan pesakit yang berbeza untuk menampung variasi amalan demografi dan klinikal adalah dinasihatkan. Matlamat strategik ini, yang diutamakan berdasarkan pandangan yang diperoleh daripada penyiasatan, bertujuan untuk mengukuhkan kaitan dan keberkesanan ICUForesee dalam meramalkan tempoh tinggal ICU dengan berkesan pada masa hadapan..

KESIMPULAN

Secara Keseluruhannya, projek ini bertujuan untuk meramalkan tempoh tinggal ICU menggunakan model pembelajaran mesin, khususnya Hutan Rawak, *Linear Regression* dan XGBoost. Selepas latihan dan ujian yang meluas, XGBoost dikenal pasti sebagai model yang paling berkesan dan seterusnya disepadukan ke dalam alat ramalan berasaskan web Bernama ICUForesee. Proses pemilihan fitur menggunakan XGBoost dalam mengenal pasti lagi faktor utama yang mempengaruhi tempoh tinggal ICU. Kedua-dua proses pemilihan model terbaik dalam melakukan ramalan dan pemilihan fitur telah dibincangkan di dalam bab sebelumnya. Oleh itu, projek ini berjaya mencapai objektifnya untuk meramalkan tempoh tinggal ICU dengan tepat dan mengenal pasti fitur-fitur penting, menunjukkan potensi untuk aplikasi praktikal dalam tetapan klinikal.

Kekuatan Sistem

Kekuatan laman web ICUForesee ini ialah ia menyediakan penyelesaian yang mudah dan boleh diakses untuk profesional penjagaan kesihatan untuk menganggarkan tempoh tinggal ICU dalam masa nyata. Selain itu, keunikan alat itu, tanpa tapak web sedia ada yang menawarkan tujuan yang sama, menggariskan sumbangan inovatifnya kepada sektor penjagaan kesihatan. Ketepatan tinggi model, dipastikan dengan menggunakan algoritma XGBoost yang unggul, meningkatkan lagi kebolehppercayaan dan potensi kesannya terhadap pembuatan keputusan klinikal.

Kelemahan Sistem

Set data yang digunakan untuk ramalan mungkin kekurangan beberapa keadaan penting dan pembolehubah yang menyumbang dengan ketara kepada tempoh tinggal ICU, yang berpotensi menghadkan kelengkapan dan ketepatan model. Tambahan pula, kebolehgneralisasian model mungkin terjejas apabila digunakan pada populasi yang berbeza atau tetapan penjagaan kesihatan yang tidak diwakili dalam set data asal. Akhir sekali, penyepaduan alat ramalan ke dalam sistem maklumat hospital sedia ada mungkin menimbulkan cabaran teknikal dan logistik, memerlukan sumber tambahan dan sokongan untuk penerimaan yang lancar.

PENGHARGAAN

Setinggi-tinggi kesyukuran saya panjatkan ke hadrat Tuhan Yang Maha Esa, dengan rahmat dan kekuatannya yang tidak terhingga telah menjadi teman setia saya sepanjang ekspedisi penyelidikan yang mendalam ini. Dalam saat-saat ketidakpastian dan cabaran, kebijaksanaan-Nya telah menerangi jalan saya, dan bimbingan-Nya telah menjadi sumber asas ketabahan saya.

Saya amat terhutang budi kepada penyelia saya yang dihormati, Dr Samsiah Sani, dengan sokongan yang tidak berbelah bahagi, nasihat dan dorongan yang tidak ternilai telah memainkan peranan penting dalam mencorakkan perjalanan projek tahun akhir saya ini. Bimbingan daripada Dr. Samsiah telah menjadi aset yang tidak ternilai, dan peranannya sebagai sumber inspirasi telah menyemarakkan dedikasi saya untuk kecemerlangan. Saya merakamkan penghargaan ikhlas saya atas kesabaran, fleksibiliti, dan maklum balas yang beliau telah berikan kepada saya, yang telah memainkan peranan penting dalam meningkatkan kualiti kerja saya.

Penghargaan juga ditujukan kepada Fakulti Teknologi Maklumat dan Sains di Universiti Kebangsaan Malaysia kerana bermurah hati menyediakan sumber penyelidikan penting dan kemudahan terkini yang telah menyumbang kepada kejayaan usaha saya dalam menjayakan projek saya ini.

Akhir sekali, terima kasih yang tulus saya tujukan kepada keluarga tersayang dan rakan-rakan yang menyokong yang telah memberi dorongan dan sokongan yang tidak berbelah bahagi. Sokongan dan dorongan mereka telah menjadi sumber motivasi dan semangat saya untuk meneruskan perjalanan dalam melakukan projek tahun akhir saya.

RUJUKAN

- R, S. E. (2023, October 26). *Understand random forest algorithms with examples (updated 2023)*. Analytics Vidhya.
<https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/>
- Brownlee, J. (2021, February 16). *A gentle introduction to XGBoost for applied machine learning*. MachineLearningMastery.com. <https://machinelearningmastery.com/gentle-introduction-xgboost-applied-machine-learning/>
- Malladi, S. V. S. A. (2023, October 1). *Unlocking the power of Shap Analysis: A comprehensive guide to feature selection*. Medium.
<https://medium.com/@msvs.akhilsharma/unlocking-the-power-of-shap-analysis-a-comprehensive-guide-to-feature-selection-f05d33698f77>
- Wirth, R., & Hipp, J. (2000). CRISP-DM: Towards a standard process model for data mining - unibo.it. CRISP-DM: Towards a Standard Process Model for Data Mining.
<http://www.cs.unibo.it/~danilo.montesi/CBD/Beatriz/10.1.1.198.5133.pdf>

Dayanithi. (2023, April 15). *Streamlit in 2 mins*. Medium. <https://medium.com/data-and-beyond/streamlit-d357935b9c>

Izzah Hanani Binti Tusiman (A188471)

Ts. Dr. Nor Samsiah Sani

Fakulti Teknologi & Sains Maklumat

Universiti Kebangsaan Malaysia

Copyright@FTSM
UKM