Breast Cancer Classification and Feature Analysis Using Machine Learning

Presented by
Jenny Aulia

Project Explanation

This project focuses on analyzing and classifying breast cancer data using machine learning techniques. The dataset used is the Breast Cancer Wisconsin dataset, which includes various features extracted from tumor samples. The goal is to train a model to classify tumors as malignant (cancerous) or benign (non-cancerous) and analyze feature importance using EDA (Exploratory Data Analysis), Random Forest, PCA, and K-Means clustering.

Exploratory Data Analysis (EDA)

```
[ ]  # Mengecek nilai yang hilang
     print("Cek Nilai Hilang:")
     print(df.isnull().sum())
```

Missing Values: Checks for null values that may require handling.

```
[4]  # Jumlah nilai unik per fitur
     print("Jumlah Nilai Unik per Fitur:")
     print(df.nunique())
```

Unique Values: Identifies categorical vs. continuous features.

```
[ ]  # Matriks Korelasi
     print("Matriks Korelasi:")
     print(df.corr())
```

Correlation Matrix: Identifies relationships between features.

```
[ ]  # Statistik deskriptif
     print("Statistik Data:")
     print(df.describe())
```

Descriptive Statistics: Provides summary metrics like mean and standard deviation.

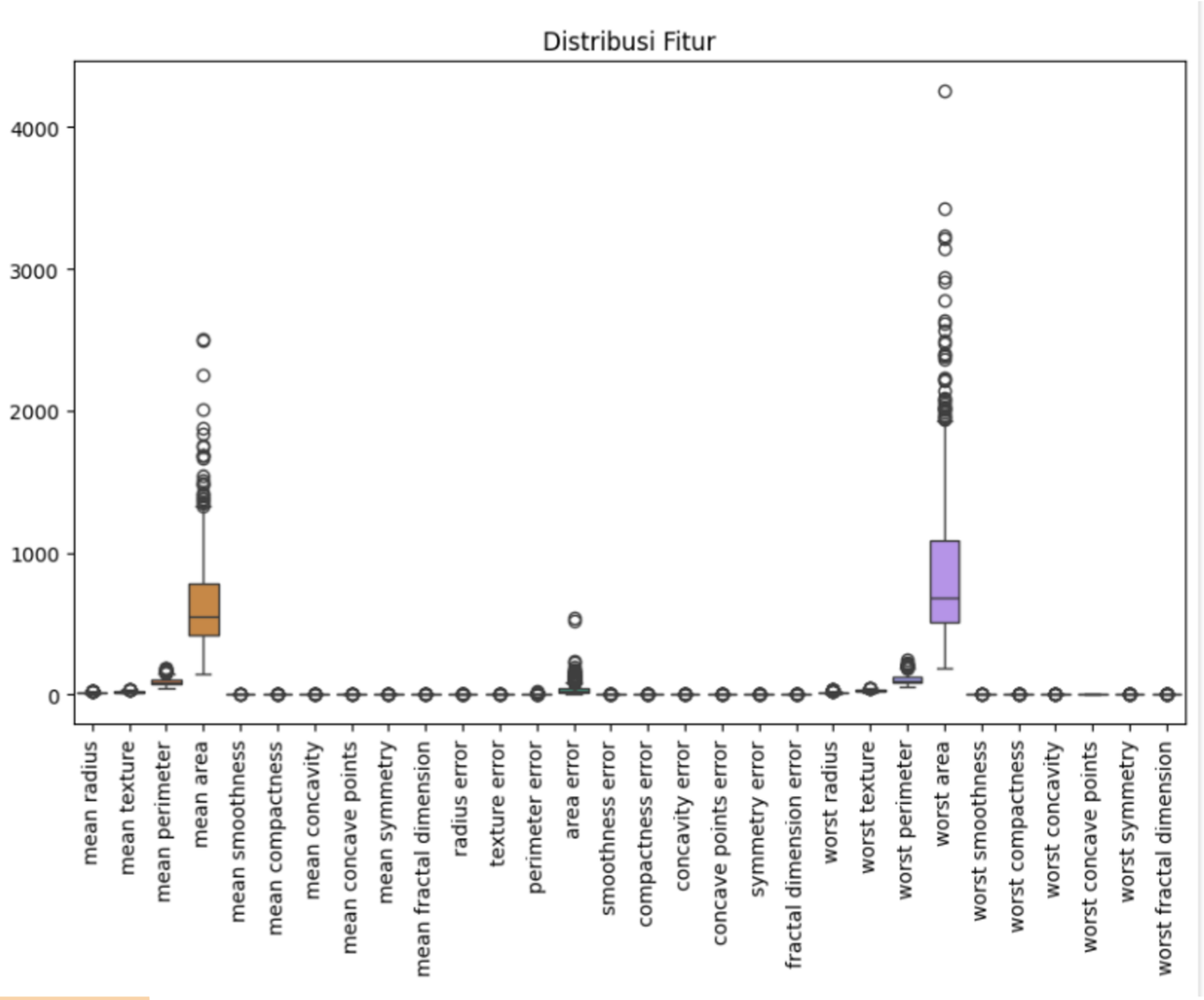
```
     # Distribusi kelas
     print("Distribusi Kelas:")
     print(df['target'].value_counts())
```

Class Distribution: Shows the count of malignant and benign cases.

Data Visualization

```python
# Visualisasi distribusi fitur dengan boxplot
plt.figure(figsize=(10, 6))
sns.boxplot(data=df.drop(columns=['target']))
plt.xticks(rotation=90)
plt.title("Distribusi Fitur")
plt.show()
```

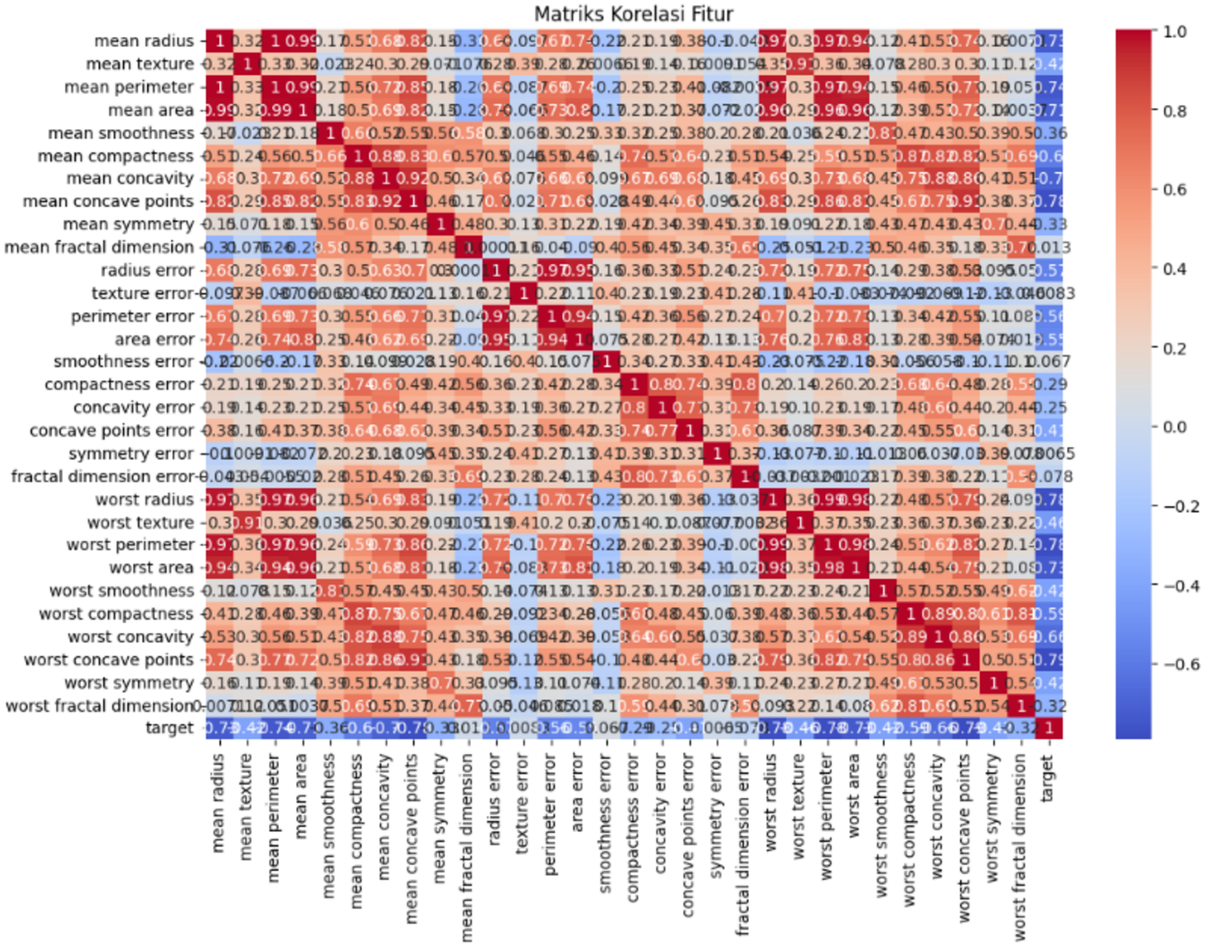
Feature Distribution (Boxplot):
Visualizes the spread of numerical features and detects potential outliers.



Data Visualization

```
[ ]  # Heatmap Korelasi
     plt.figure(figsize=(12, 8))
     sns.heatmap(df.corr(), annot=True, cmap='coolwarm')
     plt.title("Matriks Korelasi Fitur")
     plt.show()
```

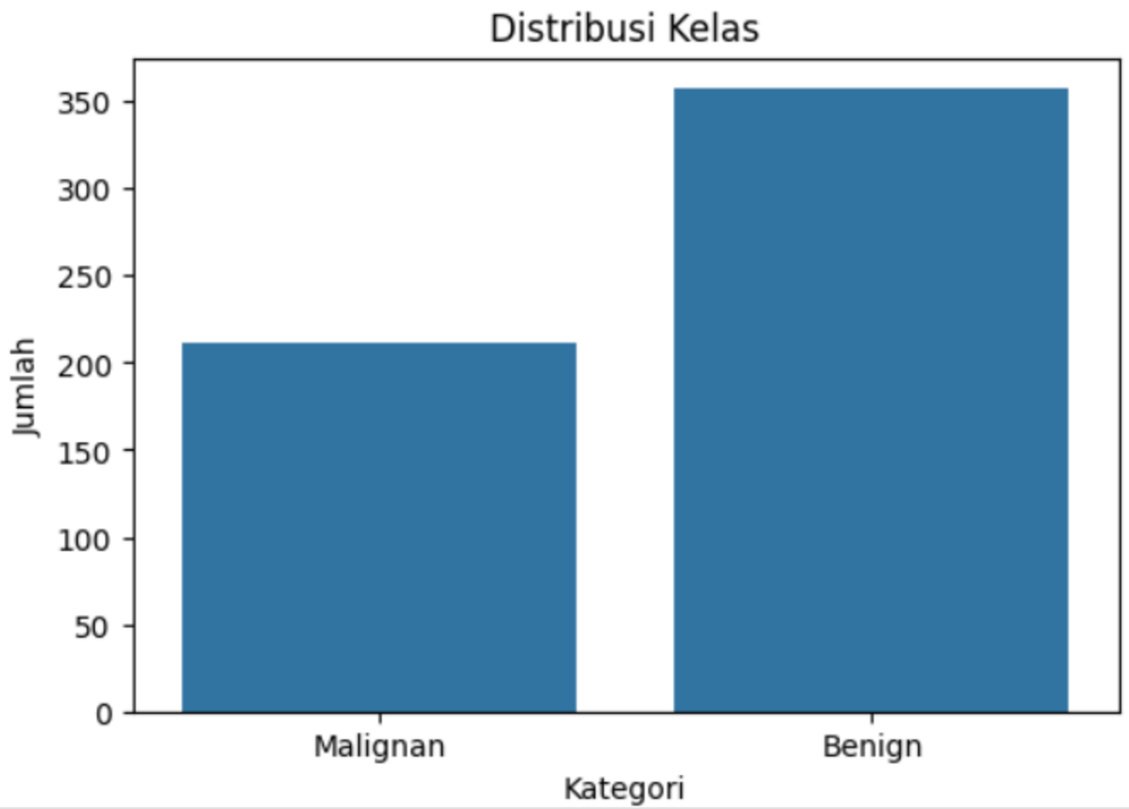
Correlation Heatmap: Displays relationships between features using a color-coded heatmap.



Data Visualization

```
[ ]  # Visualisasi distribusi kelas
     plt.figure(figsize=(6, 4))
     sns.countplot(x='target', data=df)
     plt.xticks(ticks=[0, 1], labels=['Malignan', 'Benign'])
     plt.title('Distribusi Kelas')
     plt.xlabel('Kategori')
     plt.ylabel('Jumlah')
     plt.show()
```

Class Distribution Plot: Shows the number of malignant vs. benign cases using a bar chart.

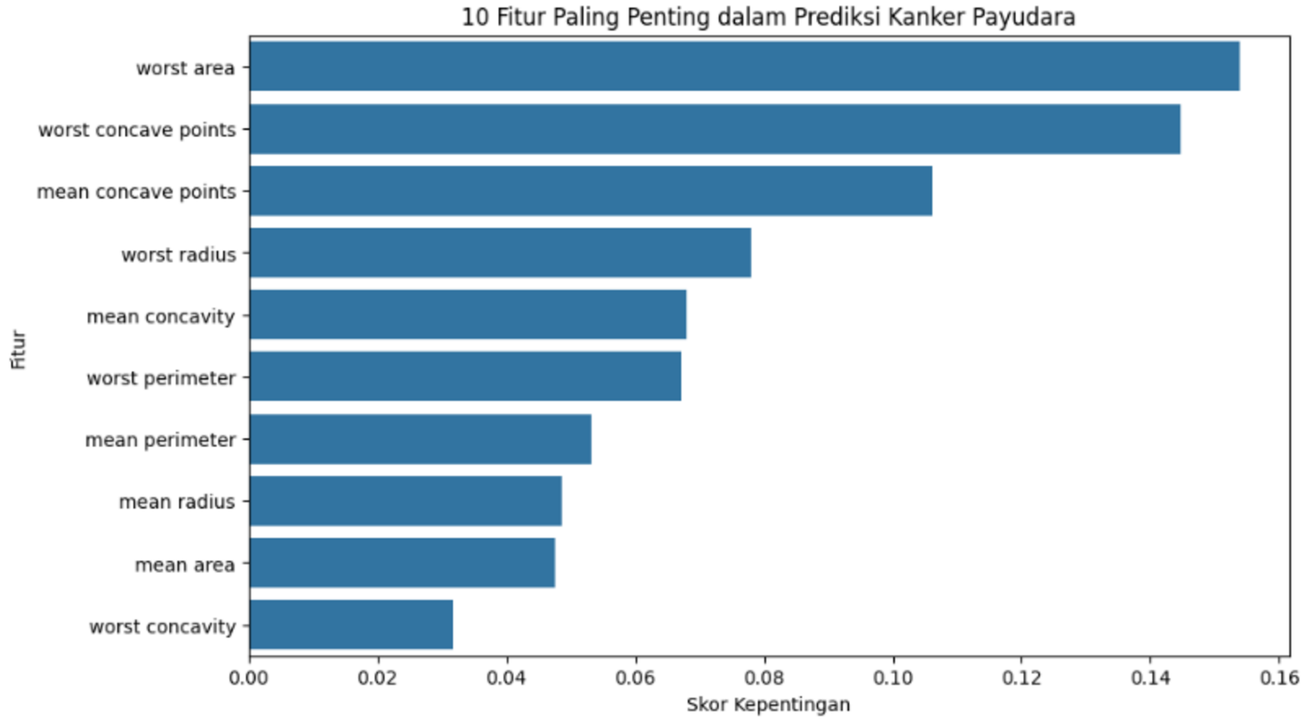


Feature Importance using Random Forest

```
[ ]  # Fitur menggunakan Random Forest
     rf = RandomForestClassifier(n_estimators=100, random_state=42)
     rf.fit(X_train_scaled, y_train)
     feature_importances = pd.Series(rf.feature_importances_,
                           index=X.columns).sort_values(ascending=False)
```

```
[ ]  # Plotting fitur
     plt.figure(figsize=(10, 6))
     sns.barplot(x=feature_importances[:10], y=feature_importances.index[:10])
     plt.title("10 Fitur Paling Penting dalam Prediksi Kanker Payudara")
     plt.xlabel("Skor Kepentingan")
     plt.ylabel("Fitur")
     plt.show()
```

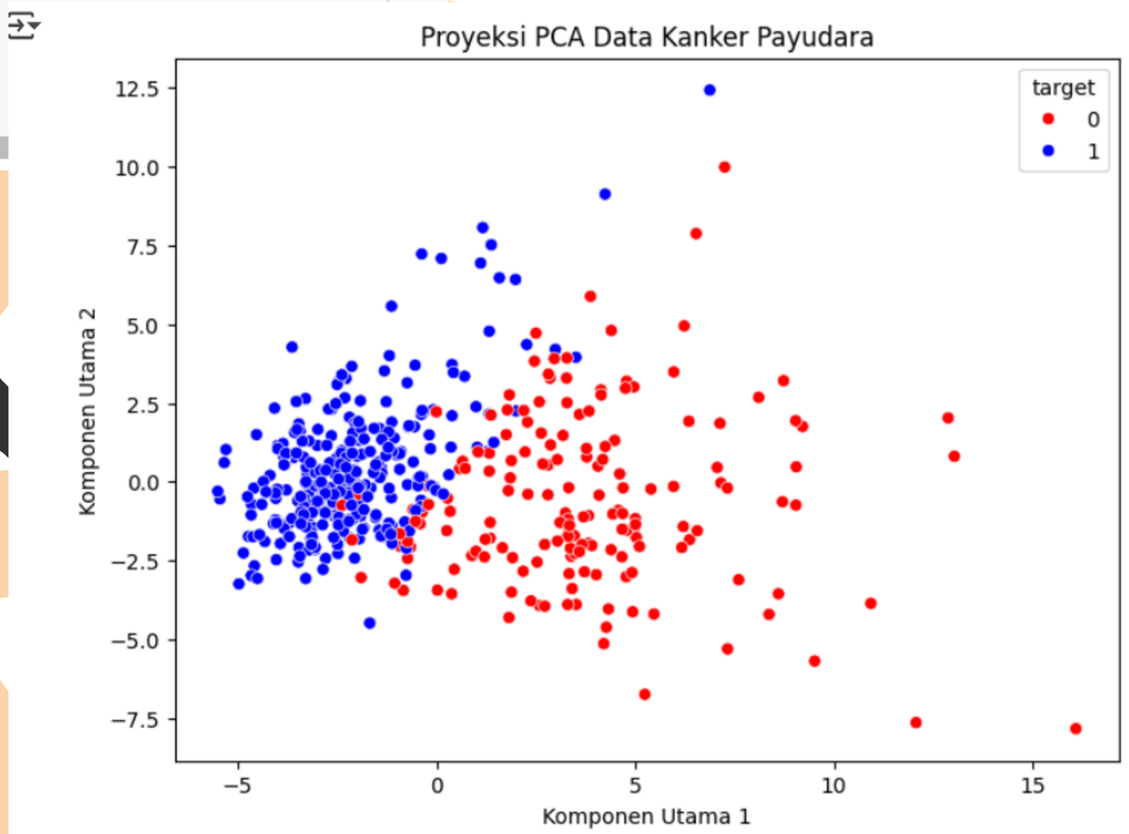
Feature Importance (Random Forest): Identifies the top 10 most influential features for classification.



Dimensionality Reduction using PCA

```
[ ]  # PCA untuk visualisasi pengurangan dimensi
     pca = PCA(n_components=2)
     X_pca = pca.fit_transform(X_train_scaled)
     plt.figure(figsize=(8, 6))
     sns.scatterplot(x=X_pca[:, 0], y=X_pca[:, 1], hue=y_train, palette=['red', 'blue'])
     plt.title("Proyeksi PCA Data Kanker Payudara")
     plt.xlabel("Komponen Utama 1")
     plt.ylabel("Komponen Utama 2")
     plt.show()
```

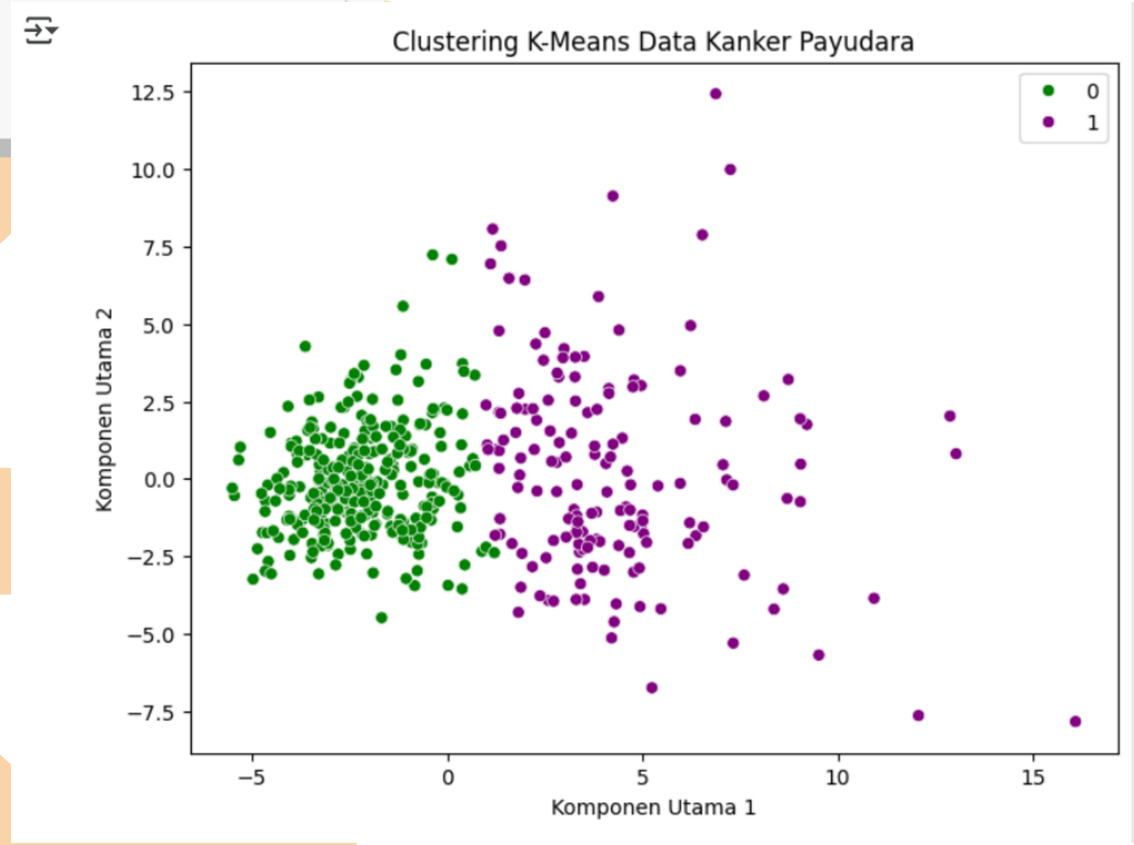
Dimensionality Reduction (PCA):
Reduces feature dimensions to two principal components for visualization.



K-Means Clustering

```
[ ]  # K-Means Clustering
     kmeans = KMeans(n_clusters=2, random_state=42)
     kmeans_labels = kmeans.fit_predict(X_train_scaled)
     plt.figure(figsize=(8, 6))
     sns.scatterplot(x=X_pca[:, 0], y=X_pca[:, 1], hue=kmeans_labels, palette=['green',
     plt.title("Clustering K-Means Data Kanker Payudara")
     plt.xlabel("Komponen Utama 1")
     plt.ylabel("Komponen Utama 2")
     plt.show()
```

K-Means Clustering: Groups data into two clusters (malignant & benign) without labels.



Classification Using Random Forest

```
[ ]  # Pelatihan model dengan Random Forest Classifier
     model = RandomForestClassifier(n_estimators=100, random_state=42)
     model.fit(X_train_scaled, y_train)
     y_pred = model.predict(X_test_scaled)
```

```
[ ]  # Evaluasi model
     print("Akurasi Model (Random Forest):", accuracy_score(y_test, y_pred))
     print("Laporan Klasifikasi:")
     print(classification_report(y_test, y_pred))
```

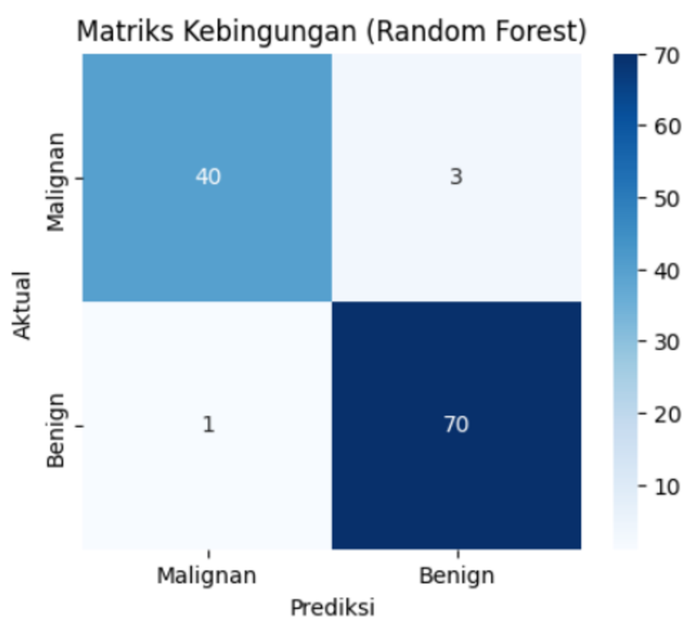
```
Akurasi Model (Random Forest): 0.9649122807017544
Laporan Klasifikasi:
              precision    recall  f1-score   support

           0       0.98      0.93      0.95        43
           1       0.96      0.99      0.97        71

    accuracy                           0.96       114
   macro avg       0.97      0.96      0.96       114
weighted avg       0.97      0.96      0.96       114
```

Classification (Random Forest): Trains a model and evaluates it using accuracy, precision, and recall.

```
[ ]  # Matriks Kebingungan untuk Random Forest
     plt.figure(figsize=(5, 4))
     sns.heatmap(confusion_matrix(y_test, y_pred), annot=True, fmt='d',
                 cmap='Blues', xticklabels=['Malignan', 'Benign'],
                 yticklabels=['Malignan', 'Benign'])
     plt.xlabel('Prediksi')
     plt.ylabel('Aktual')
     plt.title('Matriks Kebingungan (Random Forest)')
     plt.show()
```



Survival Prediction (Simulation) using Logistic Regression

```python
# Prediksi kelangsungan hidup menggunakan Logistic Regression (Simulasi)
np.random.seed(42)
survival_labels = np.random.randint(0, 2, size=len(y_train))
log_reg = LogisticRegression()
log_reg.fit(X_train_scaled[:, :5], survival_labels)
survival_preds = log_reg.predict(X_test_scaled[:, :5])
print("Akurasi Prediksi Kelangsungan Hidup (Logistic Regression):",
      accuracy_score(survival_labels[:len(y_test)], survival_preds))
print(classification_report(survival_labels[:len(y_test)], survival_preds))
```

```
Akurasi Prediksi Kelangsungan Hidup (Logistic Regression): 0.49122807017543857
              precision    recall  f1-score   support

           0       0.34      0.26      0.29        47
           1       0.56      0.66      0.60        67

    accuracy                           0.49       114
   macro avg       0.45      0.46      0.45       114
weighted avg       0.47      0.49      0.47       114
```

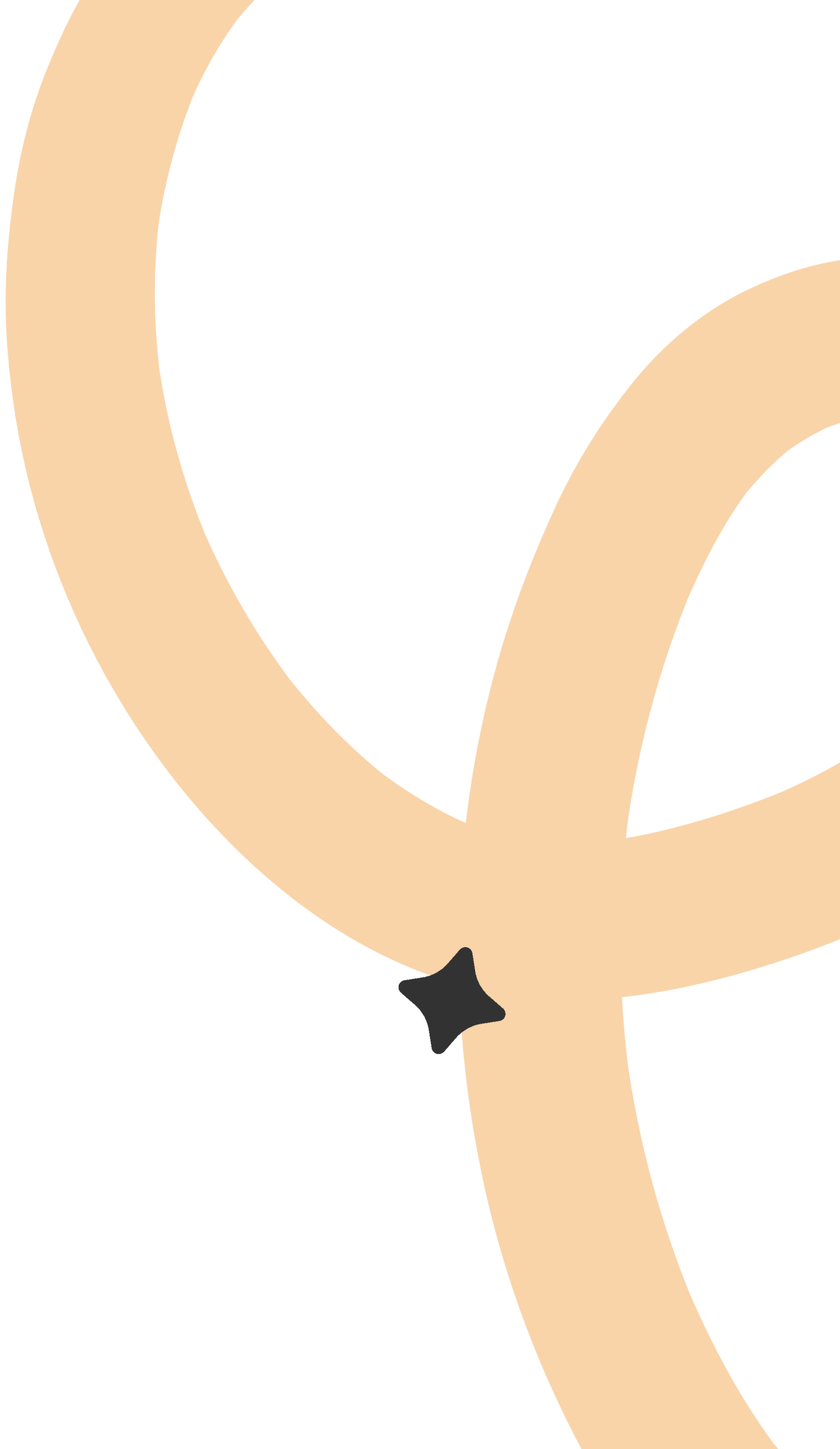
Survival Prediction (Logistic Regression - Simulation): Simulates a prediction model for patient survival based on key features.

Conclusion

This project analyzes breast cancer data using Exploratory Data Analysis (EDA), feature selection, and machine learning models to classify malignant and benign tumors. The EDA phase reveals that the dataset has no missing values, and some features contain outliers. The correlation matrix helps identify relationships between features, aiding in dimensionality reduction.

Feature selection using Random Forest highlights the most important variables, while Principal Component Analysis (PCA) reduces the data dimensions for better visualization. The Random Forest Classifier achieves high accuracy in classifying tumors, demonstrating its effectiveness. Additionally, K-Means Clustering is applied for unsupervised learning, but its accuracy is lower compared to supervised models. A Logistic Regression model is also used to simulate survival prediction, showing its potential application in prognosis.

Thank You



Jakarta Selatan, 2025

www.linkedin.com/in/jenny-aulia-a795a525b