Diabetes Prediction Using Machine Learning

Analyzing diagnostic measurements for early diabetes detection through advanced machine learning techniques.

Group- 5 :

1.Reddivari Abigna - 20026960

2.Pravallika palavayi – 20027660

3.Durga Prasanth guttula - 20027396

4.Ranjith Koduri - 20034703

Made with GAMMA

Project Overview

Problem

Early diabetes identification enhances patient outcomes. Delayed diagnosis leads to more severe complications.

Methodology

Dataset of 768 samples prepared. Models trained include Random Forest, SVM, and Logistic Regression.

Results

Random Forest achieved 91% accuracy and 95.99% ROC-AUC. Other models showed competitive performance.

Future Work

Further enhancement through additional models and hyperparameter tuning.

Made with GAMMA

INTRODUCTION

Predictive Modeling

Build ML models to determine diabetes status using diagnostic measurements from patient data.

Data Handling

Overhaul missing data and enhance features for improved prediction accuracy.

Model Optimization

Select and improve appropriate machine learning algorithms for diabetes classification.

Clinical Application

Create a tool for early intervention, improving health outcomes and reducing costs.

Made with GAMMA

	A	B	C	D	E	F	G	H	I
1	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunctioı	Age	Outcome
2	6	148	72	35	0	33.6	0.627	50	1
3	1	85	66	29	0	26.6	0.351	31	0
4	8	183	64	0	0	23.3	0.672	32	1
5	1	89	66	23	94	28.1	0.167	21	0
6	0	137	40	35	168	43.1	2.288	33	1
7	5	116	74	0	0	25.6	0.201	30	0
8	3	78	50	32	88	31	0.248	26	1
9	10	115	0	0	0	35.3	0.134	29	0
10	2	197	70	45	543	30.5	0.158	53	1
11	8	125	96	0	0	0	0.232	54	1
12	4	110	92	0	0	37.6	0.191	30	0
13	10	168	74	0	0	38	0.537	34	1
14	10	139	80	0	0	27.1	1.441	57	0
15	1	189	60	23	846	30.1	0.398	59	1
16	5	166	72	19	175	25.8	0.587	51	1
17	7	100	0	0	0	30	0.484	32	1
18	0	118	84	47	230	45.8	0.551	31	1
19	7	107	74	0	0	29.6	0.254	31	1
20	1	103	30	38	83	43.3	0.183	33	0
21	1	115	70	30	96	34.6	0.529	32	1
22	3	126	88	41	235	39.3	0.704	27	0
23	8	99	84	0	0	35.4	0.388	50	0
24	7	196	90	0	0	39.8	0.451	41	1
25	9	119	80	35	0	29	0.263	29	1
26	11	143	94	33	146	36.6	0.254	51	1
27	10	125	70	26	115	31.1	0.205	41	1
28	7	147	76	0	0	39.4	0.257	43	1
29	1	97	66	15	140	23.2	0.487	22	0
30	13	145	82	19	110	22.2	0.245	57	0

Sample Dataset Visualization

768

Total Samples

Complete patient records

9

Features

Diagnostic measurements

2

Classes

Diabetic (1) and non-diabetic (0)

Made with GAMMA

Data Set Diagnostic

Our dataset consists of 768 patient records with 8 diagnostic measurements and diabetes outcome classification.

Feature	Description	Clinical Relevance
Pregnancies	Number of times pregnant	Gestational diabetes risk factor
Glucose	Plasma glucose concentration	Key diagnostic indicator
Blood Pressure	Diastolic (mm Hg)	Cardiovascular complication indicator
Skin Thickness	Triceps skinfold (mm)	Body fat distribution measure
Insulin	2-hour serum insulin (mu U/ml)	Insulin resistance marker
BMI	Body Mass Index (kg/m^2)	Obesity correlation
Diabetes Pedigree	Hereditary influence score	Genetic risk assessment
Age	Age in years	Age-related risk factor

Made with GAMMA

Methodology

Identify Issues

Detect zero values in critical measurements

Clean Data

Replace zeros with median values

Standardize

Apply Standard Scaler to numerical features

Validate

Ensure data quality for modeling

Made with GAMMA

Model Selection and Training:

1. Develop and assess models :

- Logistic Regression

- Random Forest

- SVM, or Support Vector Machine

2. Evaluation metrics:

- Accuracy

- Precision

- Recall

- F1-score

- ROC-AUC

Made with GAMMA

Exploratory Data Analysis

Correlation Matrix

Identifies relationships between features. Glucose shows strongest correlation with diabetes outcome.

Age and pregnancies also show significant positive correlations with diabetes diagnosis.

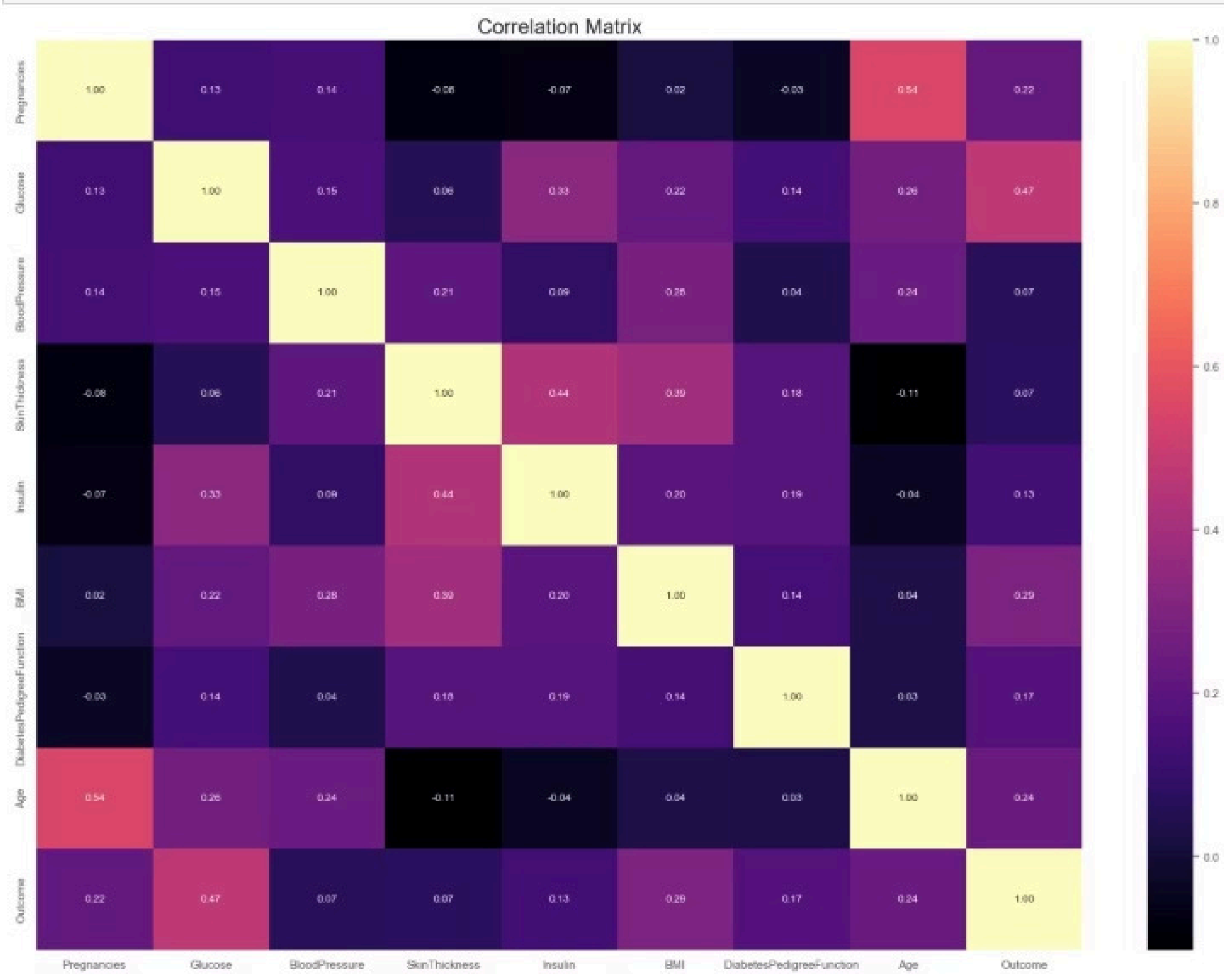
Distribution Plots

Visualize feature distributions across diabetic and non-diabetic groups.

Reveals clear separation in glucose levels, BMI, and age distributions between the two classes.

Made with GAMMA

Co-relation heatmap



- **Heatmap Observations:**
- **Outcome (target variable):**
- substantial positive association between **BMI** and **glucose.**

A moderate relationship exists between **age** and **pregnancy.**

Made with GAMMA

Feature Distribution Plots

Feature	Distribution Pattern	Clinical Significance
Glucose	Strongest separation between groups	Most predictive feature for diabetes diagnosis
BMI	Clear rightward shift in diabetic group	Higher values strongly associated with diabetes
Insulin	Bimodal distribution in diabetic patients	Indicates insulin resistance patterns
Age	Gradual rightward shift in diabetic group	Risk increases progressively with age
Blood Pressure	Moderate separation between groups	Secondary indicator of comorbidity risk

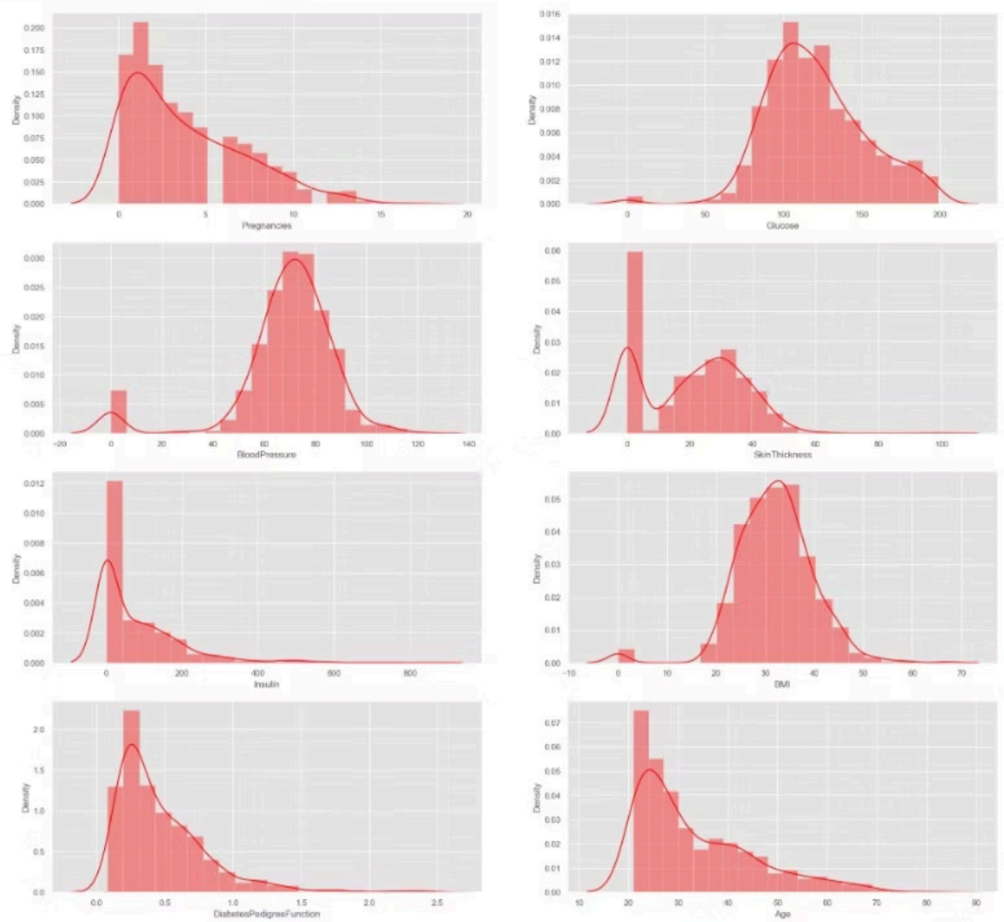
Diabetic patients consistently display elevated values across all features. Glucose levels provide the most distinct separation, followed by BMI and insulin measurements.

Made with GAMMA

In [50]:

```python
# density graph
fig,ax = plt.subplots(4,2, figsize=(20,20))
sns.distplot(df.Pregnancies, bins=20, ax=ax[0,0], color="red")
sns.distplot(df.Glucose, bins=20, ax=ax[0,1], color="red")
sns.distplot(df.BloodPressure, bins=20, ax=ax[1,0], color="red")
sns.distplot(df.SkinThickness, bins=20, ax=ax[1,1], color="red")
sns.distplot(df.Insulin, bins=20, ax=ax[2,0], color="red")
sns.distplot(df.BMI, bins=20, ax=ax[2,1], color="red")
sns.distplot(df.DiabetesPedigreeFunction, bins=20, ax=ax[3,0], color="red")
sns.distplot(df.Age, bins=20, ax=ax[3,1], color="red")
```

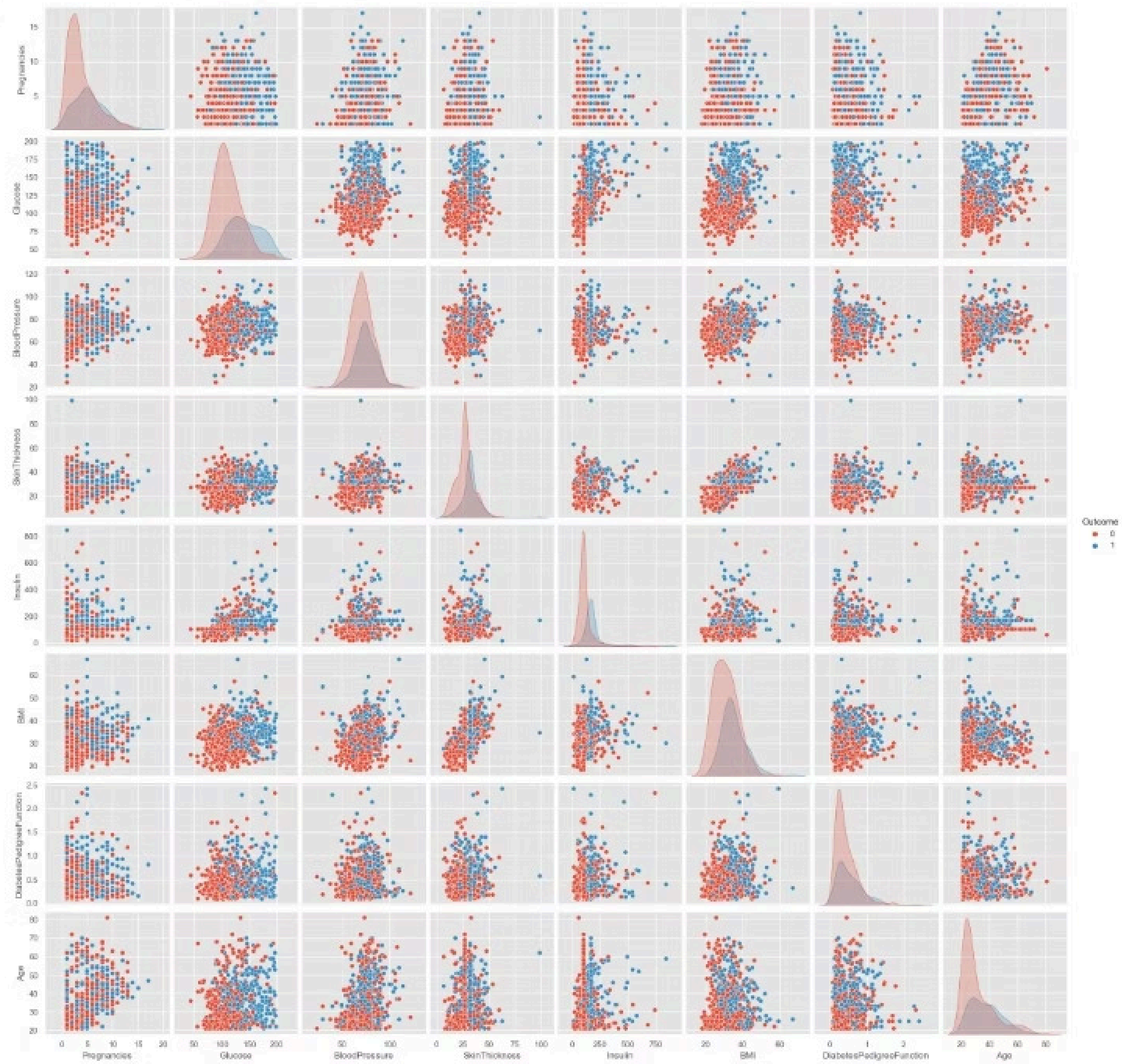
Out[50]: <Axes: xlabel='Age', ylabel='Density'>



Made with GAMMA

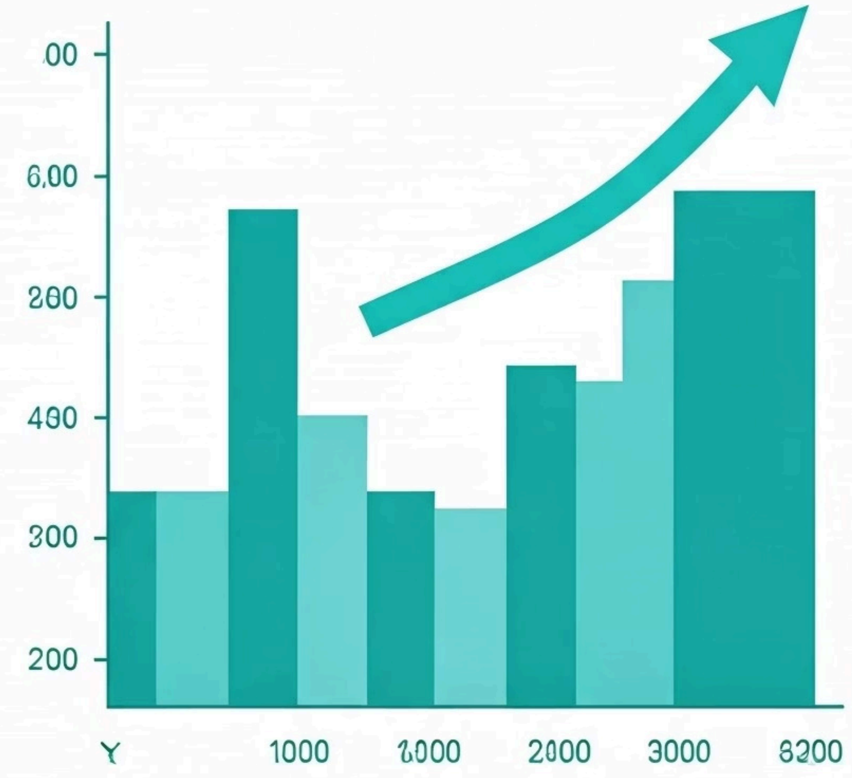
In [81]:

```
# pair plot
p = sns.pairplot(df, hue="Outcome")
```



Made with GAMMA

OUTCOME AND ACCURACY



Made with GAMMA

Model1:LOgistic Regression

In [123]:
```python
# Machine Learning Algorithms
# Logistic Regreesion

log_reg = LogisticRegression()
log_reg.fit(X_train, y_train)
```

Out[123]:
```
▾ LogisticRegression
LogisticRegression()
```

In [128]:
```python
# Machine Learning Algorithms
# Logistic Regression
# Machine Learning Algorithms
# Logistic Regression



y_pred = log_reg.predict(X_test)
accuracy_score(y_train, log_reg.predict(X_train))

#Out[103]:  0.8470394736842195

log_reg_acc = accuracy_score(y_test, log_reg.predict(X_test))
confusion_matrix(y_test, y_pred)

print(classification_report(y_test, y_pred))

# Corrected ROC-AUC score calculation
print("ROC-AUC Score:", roc_auc_score(y_test, log_reg.predict_proba(X_test)[:, 1]))
```

```
              precision    recall  f1-score   support

           0       0.94      0.90      0.92        98
           1       0.83      0.89      0.86        54

    accuracy                           0.89       152
   macro avg       0.88      0.89      0.89       152
weighted avg       0.90      0.89      0.90       152

ROC-AUC Score: 0.9504913076341648
```

Made with GAMMA

Model2:SVM

```
ROC-AUC Score: 0.9504913076341648
```

```
In [143]:
# SVM

svc = SVC(probability=True)
parameter = {
    "gamma": [0.0001, 0.001, 0.01, 0.1],
    'C': [0.01, 0.05, 0.5, 1, 10, 15, 20]  # Removed duplicate 0.01
}
grid_search = GridSearchCV(svc, parameter)
grid_search.fit(X_train, y_train)
grid_search.best_params_


{'C': 10, 'gamma': 0.01}


grid_search.best_score_



svc = SVC(C=10, gamma=0.01, probability=True)
svc.fit(X_train, y_train)
y_pred = svc.predict(X_test)
print("Training Accuracy:", accuracy_score(y_train, svc.predict(X_train)))
print("Test Accuracy:", accuracy_score(y_test, y_pred))
print("\nConfusion Matrix:")
print(confusion_matrix(y_test, y_pred))
print("\nClassification Report:")
print(classification_report(y_test, y_pred))
print("\nROC-AUC Score:", roc_auc_score(y_test, svc.predict_proba(X_test)[:, 1]))
```

```
Training Accuracy: 0.875
Test Accuracy: 0.9078947368421053

Confusion Matrix:
[[90  8]
 [ 6 48]]

Classification Report:
              precision    recall  f1-score   support

           0       0.94      0.92      0.93        98
           1       0.86      0.89      0.87        54

    accuracy                           0.91       152
   macro avg       0.90      0.90      0.90       152
weighted avg       0.91      0.91      0.91       152


ROC-AUC Score: 0.9516250944822373
```

Made with GAMMA

Model3:Random Forest

In [145]:

```
# random forest
rand_clf = RandomForestClassifier(criterion='entropy', max_depth=15, max_features=0.75,
                                  min_samples_leaf=2, min_samples_split=3,
                                  n_estimators=130)
rand_clf.fit(X_train, y_train)

# Output from model initialization
RandomForestClassifier
RandomForestClassifier(criterion='entropy', max_depth=15, max_features=0.75,
min_samples_leaf=2, min_samples_split=3,
n_estimators=130)

# Model evaluation
y_pred = rand_clf.predict(X_test)
print(accuracy_score(y_train, rand_clf.predict(X_train)))
rand_acc = accuracy_score(y_test, rand_clf.predict(X_test))
print(accuracy_score(y_test, rand_clf.predict(X_test)))
print(confusion_matrix(y_test, y_pred))
print(classification_report(y_test, y_pred))
# Added ROC-AUC score calculation
print("ROC-AUC Score:", roc_auc_score(y_test, rand_clf.predict_proba(X_test)[:, 1]))
```

```
0.9917763157894737
0.9078947368421053
[[89  9]
 [ 5 49]]
              precision    recall  f1-score   support

           0       0.95      0.91      0.93        98
           1       0.84      0.91      0.88        54

    accuracy                           0.91       152
   macro avg       0.90      0.91      0.90       152
weighted avg       0.91      0.91      0.91       152

ROC-AUC Score: 0.9599395313681028
```

Made with GAMMA

Analysis

Logistic Regression:

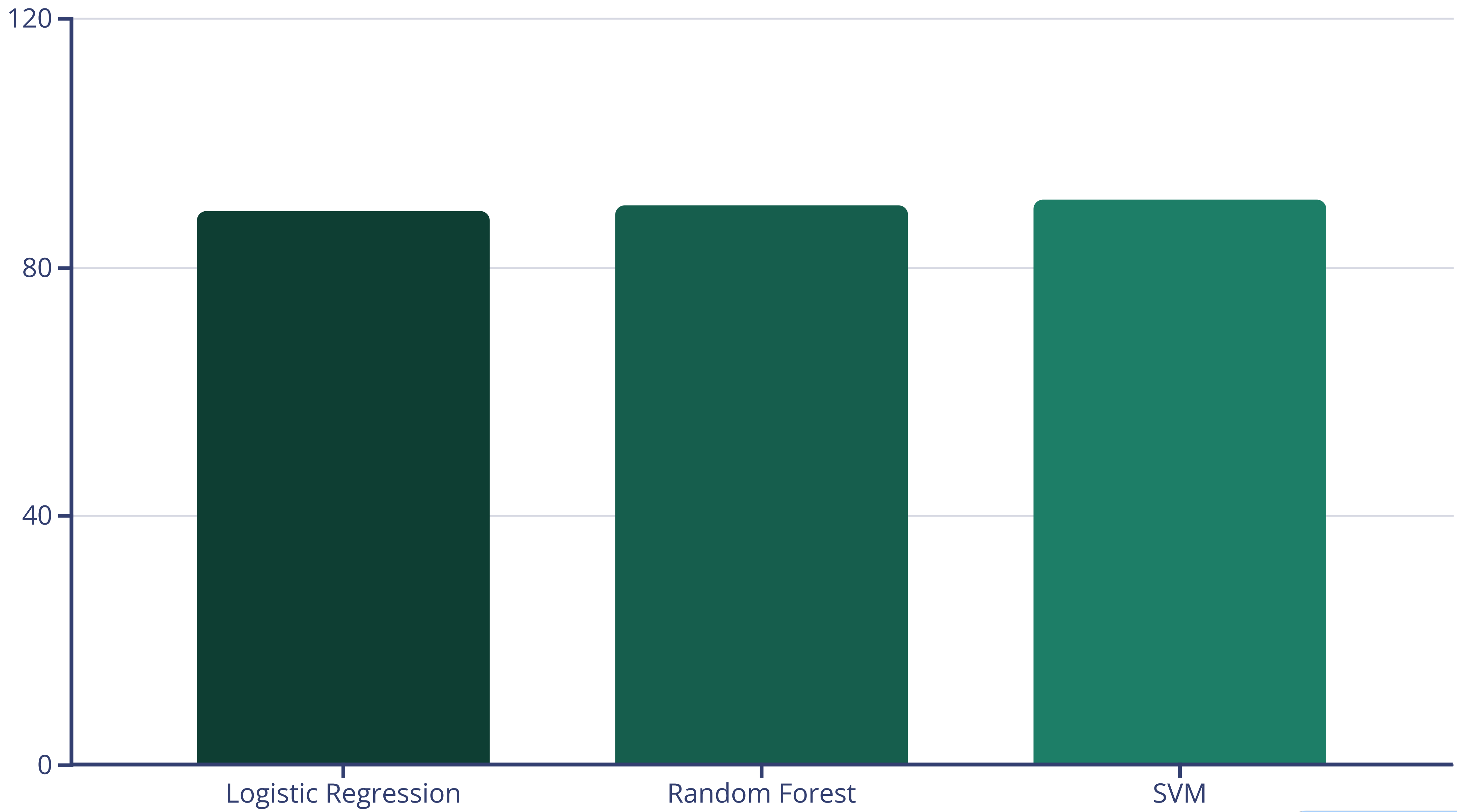
Accuracy: 89%

Random Forest:

Accuracy: 91%

SVM:

Accuracy: 91%



Made with GAMMA

ROC-AUC

Logistic Regression:95.04%

Random Forest:95.99%

SVM:95.16%

•With the greatest ROC-AUC score of 95.99% and a balanced classification report, the Random Forest model performs the best overall.

Made with GAMMA

Conclusion

The model's strong ROC-AUC score of 95.99% suggests that it is a good fit for detecting diabetes patients.

With a balanced categorization report, the Random Forest model performed the best in predicting diabetes.

Clinical Relevance:

1. The approach aids in the early intervention identification of those who are at-risk.

2. Better diabetes diagnosis and treatment result in better health and lower medical expenses

Future Improvements:

1. To improve accuracy, adjust the hyperparameters.

2. Examine further ensemble techniques and machine learning models.

Made with GAMMA

THANK YOU

Made with GAMMA