

```

In [6]: import pandas as pd
from sklearn.model_selection import train_test_split
from catboost import CatBoostClassifier
from sklearn.metrics import roc_auc_score, roc_curve
import matplotlib.pyplot as plt

# Загрузка данных
data = pd.read_csv('grant_data_imb_pro.csv')

# Отделение целевой переменной и заполнение пропусков
X = data.drop('Grant.Status', axis=1)
y = data['Grant.Status']
X.fillna(-999, inplace=True)

# Разделение данных на обучающую и тестовую выборки
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,

# Определение категориальных признаков
cat_features = list(X_train.select_dtypes(include=['object', 'category']))

# Создание и обучение модели CatBoostClassifier
model = CatBoostClassifier(
    iterations=1000,
    learning_rate=0.1,
    depth=6,
    cat_features=cat_features,
    eval_metric='AUC',
    verbose=200,
    early_stopping_rounds=50
)
model.fit(
    X_train, y_train,
    eval_set=(X_test, y_test),
    use_best_model=True
)

# Прогнозирование и расчет ROC AUC
predictions = model.predict_proba(X_test)[:, 1]
roc_auc = roc_auc_score(y_test, predictions)
print(f"ROC AUC Score: {roc_auc}")

# Расчет параметров для ROC кривой
fpr, tpr, thresholds = roc_curve(y_test, predictions)

# Создание графика
plt.figure(figsize=(8, 6))
plt.plot(fpr, tpr, label=f'CatBoost (area = {roc_auc:.2f})')
plt.plot([0, 1], [0, 1], 'k--') # случайная модель
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('ROC Curve')
plt.legend(loc="lower right")
plt.show()

```

0: test: 0.8555135 best: 0.8555135 (0) total: 5.88ms remaining: 5.88s

Stopped by overfitting detector (50 iterations wait)

bestTest = 0.9549560143

bestIteration = 127

Shrink model to first 128 iterations.

ROC AUC Score: 0.95495601428447

