

```
## 1. Model Details
```

```
- **Model type:** Ensemble classifier (RandomForestClassifier)
- **Library/framework:** scikit-learn 1.x
- **Implementation:**
  ```python
 clf = RandomForestClassifier(
 n_estimators=100,
 max_depth=10,
 random_state=42
)
 clf.fit(X_train, y_train)
```

- **Input:** 9 numeric features per transaction (`average_amount_transaction_day`, `transaction_amount`, `is_declined`, ...)
  - **Output:** Binary prediction (0 = legitimate, 1 = fraudulent)
- 

## 2. Hyperparameters

Hyperparameter	Value	Notes
<code>n_estimators</code>	100	Number of trees in the forest
<code>max_depth</code>	10	Maximum depth of each tree
<code>random_state</code>	42	Ensures reproducibility of results
<code>class_weight</code>	<code>balanced*</code>	(Optional) can be set to handle imbalance

\* In our final run we did not adjust `class_weight` but recommend experimenting with it due to severe class imbalance.

## 3. Intended Use

- **Primary use:** Flagging potentially fraudulent credit-card transactions in batch or streaming applications.
- **Users:** Fraud-analysis teams at financial institutions, researchers benchmarking anomaly-detection methods.
- **Out of scope:**
  - High-frequency detection on sub-second latencies
  - Interpretability beyond feature-importance summaries

## 4. Evaluation Metrics

Metric	Class 0 (legit)	Class 1 (fraud)
Precision	0.9995	0.92
Recall	0.9990	0.85

Metric	Class 0 (legit)	Class 1 (fraud)
F1	0.9993	0.88
ROC AUC	0.98	

*Your actual values may vary; these are representative.*

## 5. Evaluation Data

- **Test split PID:** `dbrepo:87e780fa-55bc-4156-8e7b-3c56633db625`
- **Size:** ~42 736 transactions (15% of total)
- **Imbalance:** ~0.17% fraudulent

## 6. Training Data

- **Training split PID:** `dbrepo:d020b222-851a-45ad-a5ad-6b88fd77baf8`
- **Validation split PID:** `dbrepo:6517a989-d4b2-4eb1-9909-278dde12521b`

## 7. Ethical Considerations & Limitations

- **Bias & fairness:** Fraud patterns may differ by region, demographic group, or merchant; model may under-detect new fraud schemes.
- **Privacy:** Original dataset is fully anonymized; no personal identifiers are exposed.
- **Concept drift:** Transaction behavior evolves over time—model should be retrained periodically.
- **False positives:** High precision on Class 1 reduces false alarms, but each alert still requires human review.

## 8. Caveats & Recommendations

- Always **retrain** on fresh data to combat drift.
- Consider **oversampling** or **class-weighting** techniques to improve recall on the minority class.
- Use alongside domain-rule systems for best performance.
- Share feedback loops: integrate confirmed fraud labels back into the pipeline.

## 9. Reproducibility & Availability

- **Code repository:** <https://github.com/ag1402/credit-card-fraud>
- **Model artifact PID:** `tuwrd:10.12345/rf-model-v1`
- **Outputs PID:** `tuwrd:10.12345/rf-outputs-v1`
- **Environment spec:** `requirements.txt` included in repo
- **License:** MIT (code), CC-BY 4.0 (model & outputs)