# Income Imputation Assignment Report

**Name:** selvamanian S
**Date of submission: 18-sept-2025**

## 1. Introduction

The purpose of this assignment is to build an **Income Imputation Model** that predicts proxy income (`final_tpv`) for loan applicants. Income is a key driver in credit risk assessment, but it is often missing or underreported. By imputing income using other available variables, lenders can strengthen their risk frameworks, automate decisioning, and improve compliance with regulatory standards such as Ind AS 109 and Basel III/IV.

## 2. Data Exploration & Cleaning

**Dataset:** `Income Imputation Base Data csv`

**Key steps performed:**

- **Missing values**:
  - Dropped `Tier` and `Margin` (100% missing).
  - Filled categorical nulls with `"Unknown"`.
  - Filled numeric nulls (`age`, `score`, `disbursed_amount`, `final_tpv`) with median.
- **Duplicates**:
  - Removed duplicate `loan_application_no`.
- **Outliers**:
  - Applied **winsorization** at 1st and 99th percentiles for `disbursed_amount` and `final_tpv`.
- **Datatypes**:
  - Converted `disbursed_date` → datetime.
  - Converted `age`, `score`, `disbursed_amount` → numeric.
    After cleaning: Dataset had consistent values, no missing data, and stable numeric distributions.

## 3. Feature Engineering

Derived new features to capture risk and income signals:

- **Date features**:
  - `disbursed_year`, `disbursed_month`, `disbursed_quarter`.
- **Risk bands**:
  - `score_band` (`<600`, `600–699`, `700–749`, `>=750`).
  - `age_band` (`<=21`, `22–25`, `26–30`, `31–35`, `36–45`, `46–60`, `60+`).
- **City features**:
  - `is_metro_city` (binary flag for metros like Mumbai, Delhi, Chennai, etc.).
- **Ratios/Interactions**:
  - `loan_to_tpv` = disbursed_amount ÷ TPV.
  - `amount_per_age` = disbursed_amount ÷ age.
  - `amount_x_score` = disbursed_amount × score.

Feature engineering ensured the model captures both demographic and transactional effects on income.

## 4. Model Development

Three models were developed and compared:

1. **Linear Regression** – Baseline model for interpretability.
2. **Random Forest Regressor** – Ensemble model to capture non-linearities.
3. **Gradient Boosting Regressor** – Boosted trees for fine-grained accuracy.

### Model Metrics

| Model | MAE | RMSE | $R^2$ |
|---|---|---|---|
| Linear Regression | 26,216 | 42,444 | 0.64 |
| Random Forest | 1,308 | 4,368 | 0.996 |
| Gradient Boosting | 2,483 | 4,331 | 0.996 |

**Random Forest** performed best, achieving the lowest MAE and strong $R^2$. Gradient Boosting was also competitive.

## 5. Model Validation

- Performed **5-fold cross-validation** on Random Forest.
- Results were stable: low variance across folds.
- Confirms model is not overfitting.

## 6. Feature Importance

Top 5 drivers of predicted income:

1. **Disbursed amount**
2. **Score**
3. **Loan-to-TPV ratio**
4. **Age bands**
5. **Metro city flag**

**Interpretation:**

- Higher **disbursed amounts** and **scores** strongly predict higher income.
- Applicants in metro cities tend to show higher income capacity.
- A high **loan-to-TPV ratio** may indicate income stress.

## 7. Business Insights

- Predicted income can be applied in **credit risk decisioning** as follows:
  **Eligibility assessment**: Use proxy income to calculate FOIR/DTI when declared income is missing.
- **Risk segmentation**: Distinguish between high and low-income proxy groups for policy rules.
- **Credit limits**: Recommend appropriate loan limits and reduce over-leverage.
- **Portfolio monitoring**: Identify early warning signals when predicted income deviates from actual repayment behavior.
- **Regulatory compliance**: Strengthen Ind AS 109 / Basel models by improving Expected Credit Loss (ECL) calculations.

## 8. Deliverables

- `income_imputation.ipynb` – notebook with step-by-step code.
- `cleaned_with_predicted_income.csv` – dataset with predicted income.
- `feature_importance.csv` – variable importance ranking.
- `model_metrics.json` – model performance summary.
- `best_income_imputer.joblib` – trained Random Forest model.
- This **report** (PDF).

## Conclusion:

The assignment successfully demonstrated an **end-to-end machine learning pipeline** for income imputation: from data cleaning to feature engineering, model training, validation, and business insights.

**Random Forest** was chosen as the final model due to superior accuracy and stable validation results. The predicted income column can be directly used in **credit risk frameworks** to support better underwriting, monitoring, and regulatory compliance.