Income Prediction Using Census Data

Agenda

- Problem Description
- Data Exploration
- Data Cleaning
- Data Normalization
- Handling Imbalanced Data
- Model Evaluation
- Feature Selection
- Model Prediction

Problem Description

- The aim is to build models to determine the income level of the people in U.S.
- It's a binary classification problem to predict if an individual has an income higher than $50k/year
- Which of the variables(age, occupation, race, etc.) are the most decisive for determining the income of a person
- Which model have the best performance

Data Exploration

- Source:https://archive.ics.uci.edu/ml/datasets/Census-Income+(KDD)
- Steps:
  a. Check Data shape
  b. Check Data types for all the columns and change if required
  c. Check target column data levels
  d. Explore and visualize the patterns in data

a) Check Data shape

We downloaded the dataset and separated into train and test dataset. Following are the observations:

Total:299285 rows and 41 columns

	TRAIN DATASET	TEST DATASET
Rows	199523	99762
Columns	41	41

b) Check Data types for all the columns and change if required

On checking the dataset we identified the Numerical Columns and Categorical Columns changed their datatypes:

- Categorical Columns identified: #34/41 Cols

factorCols = ['class_of_worker','industry_code','occupation_code','education','enrolled_in_edu_inst_lastwk', 'marital_status', 'major_industry_code', 'major_occupation_code','race', 'hispanic_origin','sex', 'member_of_labor_union', 'reason_for_unemployment', 'full_parttime_employment_stat', 'tax_filer_status', 'region_of_previous_residence', 'state_of_previous_residence', 'd_household_family_stat', 'd_household_summary', 'migration_msa', 'migration_reg', 'migration_within_reg', 'live_1_year_ago', 'migration_sunbelt', 'family_members_under_18', 'country_father', 'country_mother', 'country_self', 'citizenship', 'business_or_self_employed', 'fill_questionnaire_veteran_admin', 'year','veterans_benefits','income_level']

- Numerical Columns identified: #7/41 Cols

numCols = ['age','wage_per_hour','capital_gains','capital_losses','dividend_from_Stocks','num_person_Worked_employer','weeks_worked_in_year']

Features & Response Variable:

Response Variable:
income_level of below 50K or above 50K
Features:
Age, industry_code, occupation_code, education
marital_status, race, sex...

c) Check target column data levels

The target column is income_level

On checking initial rows identified that the target column has different denominations ie:

In TrainData we have -50000 & +50000 for income_level
In TestData we have -50000 & 50000+ for income_level

d) Explore and visualize the patterns in data

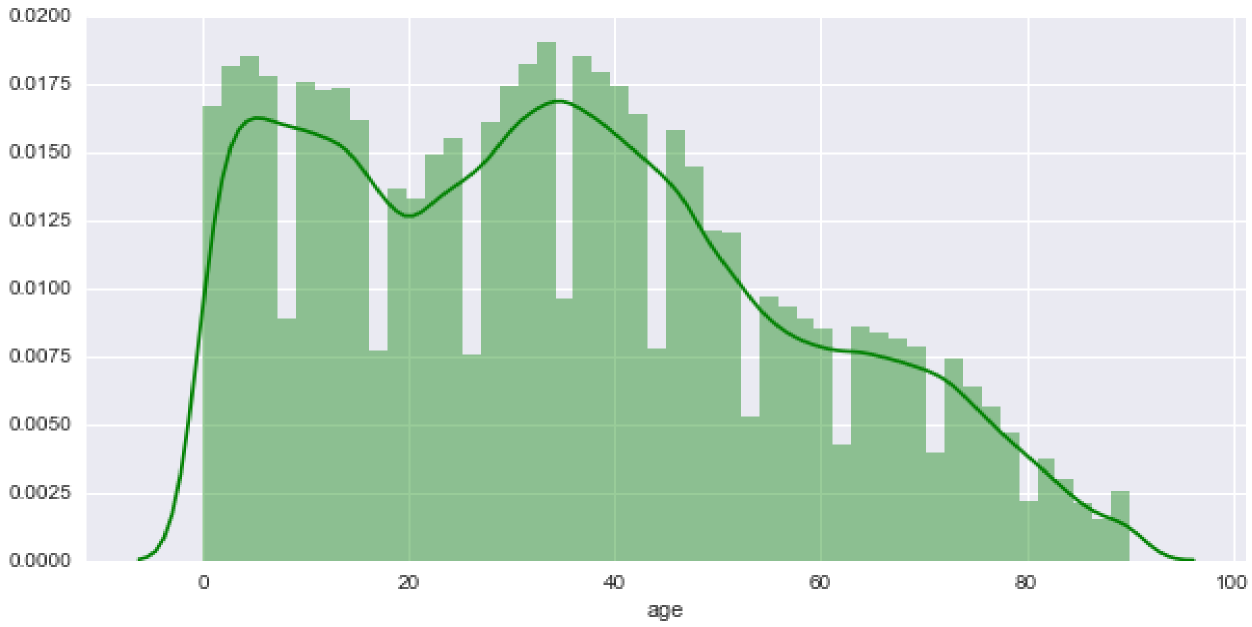
For further data exploration we separated the dataset in Numerical and Categorical Dataset. We analyzed the columns to find patterns. Below the observations we found:

- ❖ Observations on Numerical Data
- ❖ Observations on Numerical Data with Target Variable
- ❖ Observations on Categorical Data

❖ Observations on Numerical Data

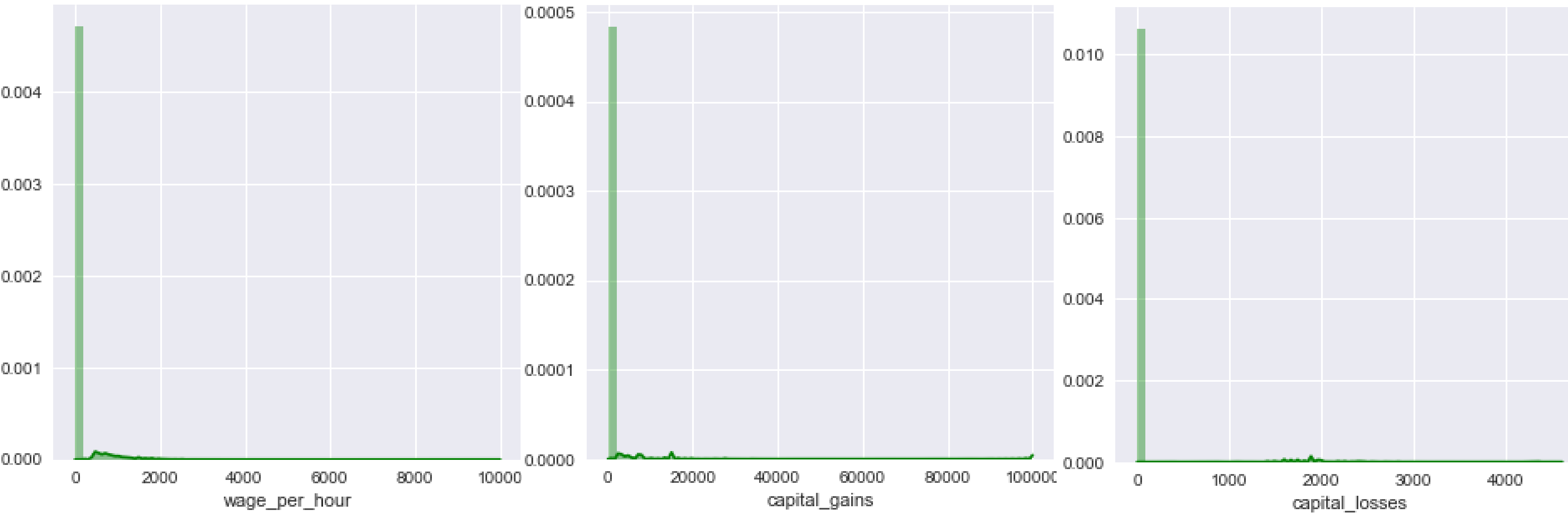
Distribution Pattern of some numerical columns on the Graph:

1. Age



Conclusion: Earning class is from 0-90

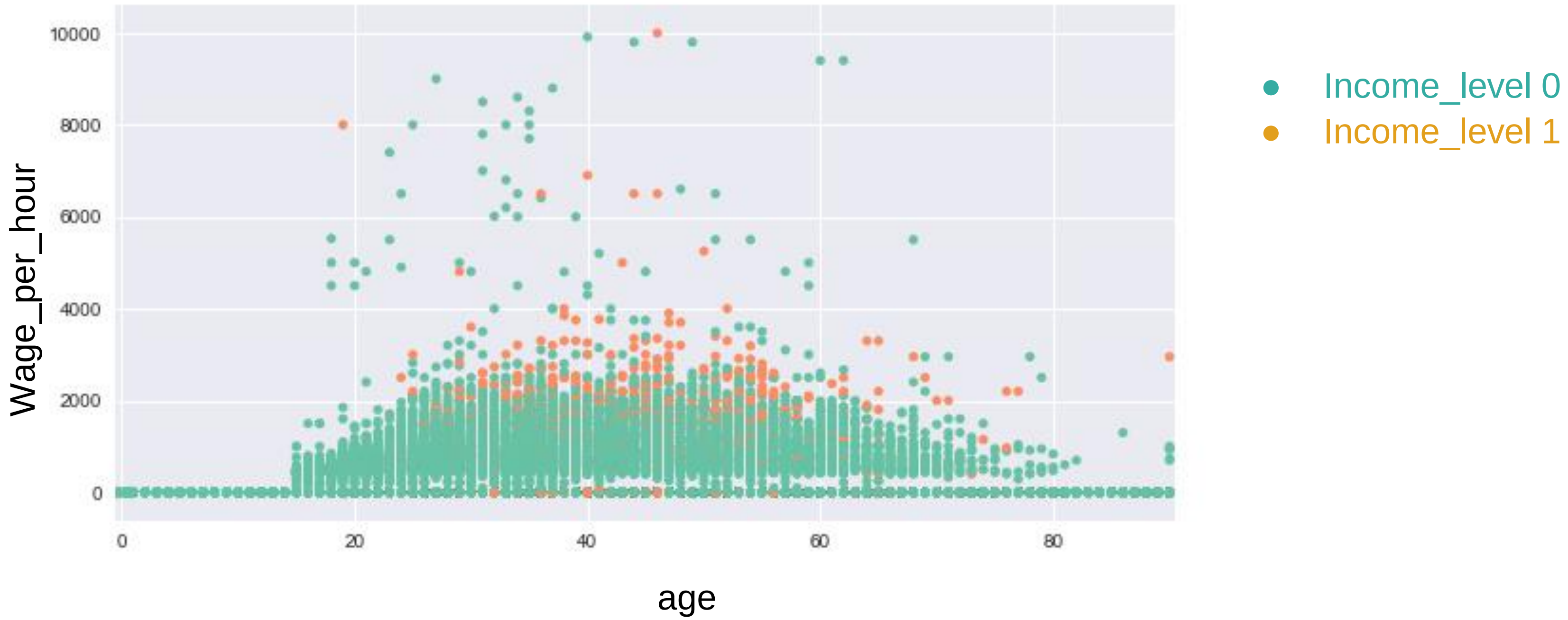
2. capital_Losses , capital_gains, wage_per_hour



Conclusion: : Highly skewed graph, We can check for unique values and may need to normalize if unique values are less.

❖ Observations on Numerical Data with Target Variable

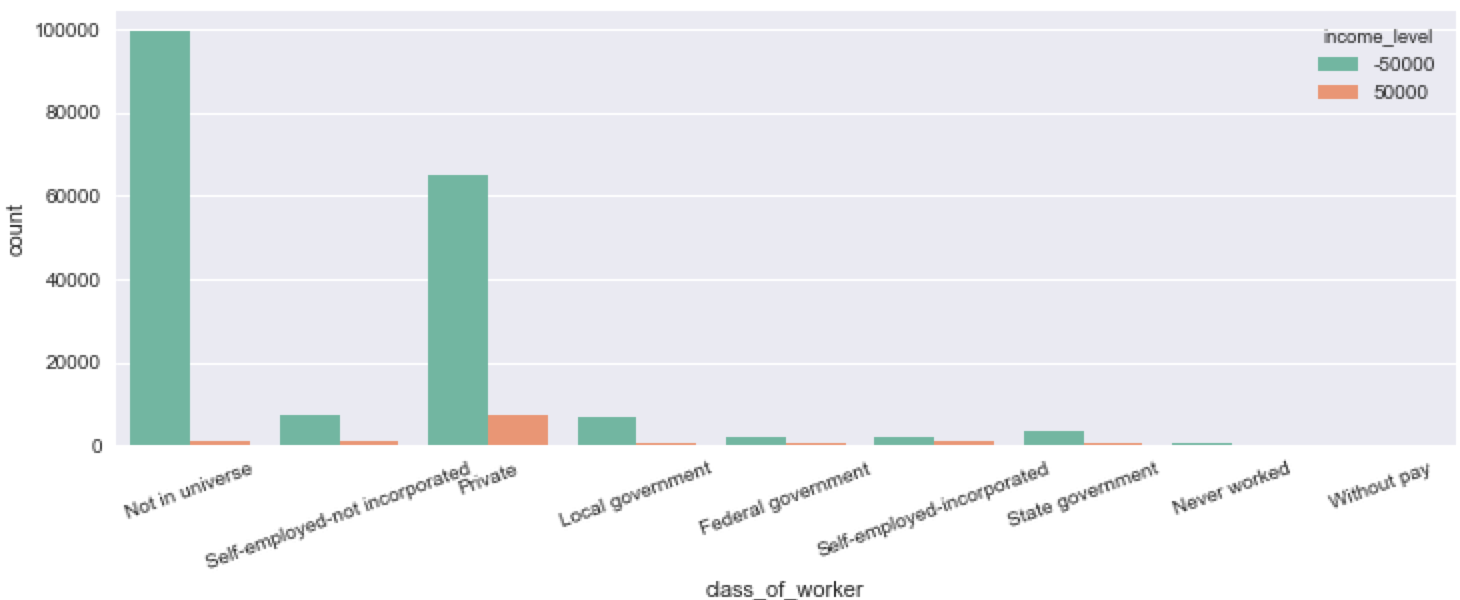
Checked the age, wage_per_hour and income_level



Conclusion:
Majority of income_level 1 fall in 25-65 age group.
Age group 0-20 have income level as 0.

❖ Observations on Categorical Data

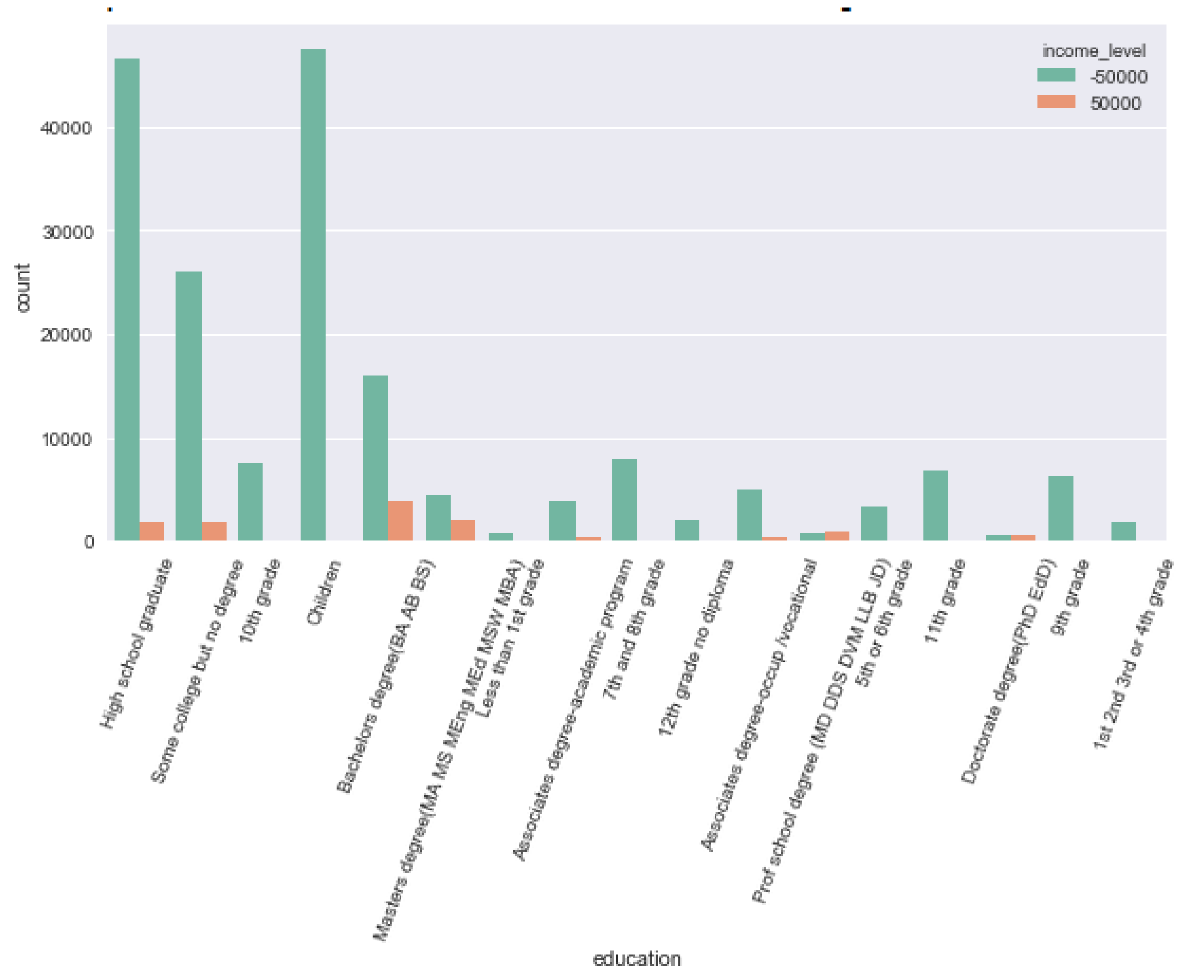
1. Class_of_worker



Conclusion:
Majority dominant in only two categories ‘Not in Universe’ and ‘Private’ rest all are very less and can be combined together to new category ‘Others’

2. Education

Conclusion:
People with the Bachelor's degree are the highest earning class with income_level 1 whereas children have no earnings and all have income_level 0.



Descriptive Statistics

	citizenship	business_or_self_employed \	num_person_Worked_employer	weeks_worked_in_year \
count	199523.000000	199523.000000	199523.000000	199523.000000
mean	0.978594	0.108003	1.956180	23.174897
std	0.335361	0.351259	2.365126	24.411488
min	0.000000	0.000000	0.000000	0.000000
25%	1.000000	0.000000	0.000000	0.000000
50%	1.000000	0.000000	1.000000	8.000000
75%	1.000000	0.000000	4.000000	52.000000
max	2.000000	2.000000	6.000000	52.000000

	fill_questionnaire_veteran_admin	year	veterans_benefits \
count	199523.000000	199523.000000	199523.000000
mean	0.009944	0.499672	0.772332
std	0.099221	0.500001	0.442407
min	0.000000	0.000000	0.000000
25%	0.000000	0.000000	1.000000
50%	0.000000	0.000000	1.000000
75%	0.000000	1.000000	1.000000
max	1.000000	1.000000	2.000000

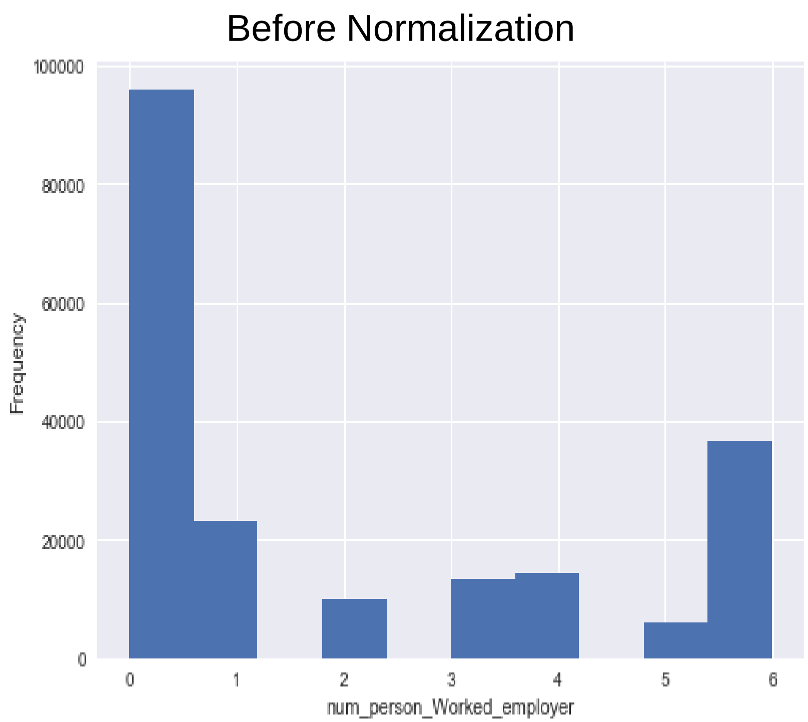
Data Cleaning

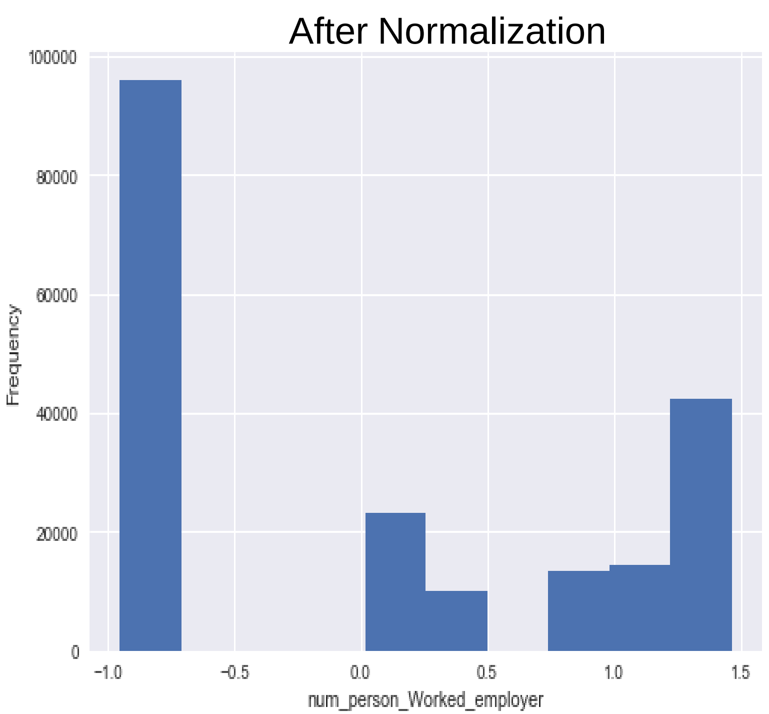
Steps

- Check for missing values
  - Delete columns with >5% missing values
  - Set missing data as ‘Unavailable’ in columns with <5%
- Encode categorical values to numerical
- Bin columns with high % of zero values to Zero and Non-Zero
- Bin age variable into age groups

Data Normalisation

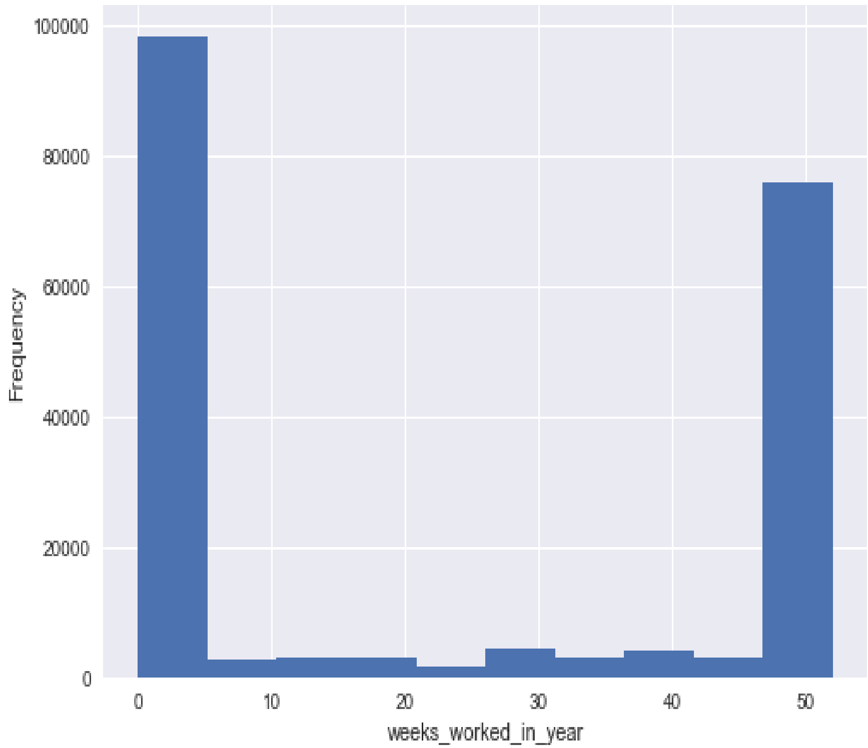
- 2 highly skewed numerical columns
  - num_person_Worked_employer
  - weeks_worked_in_year



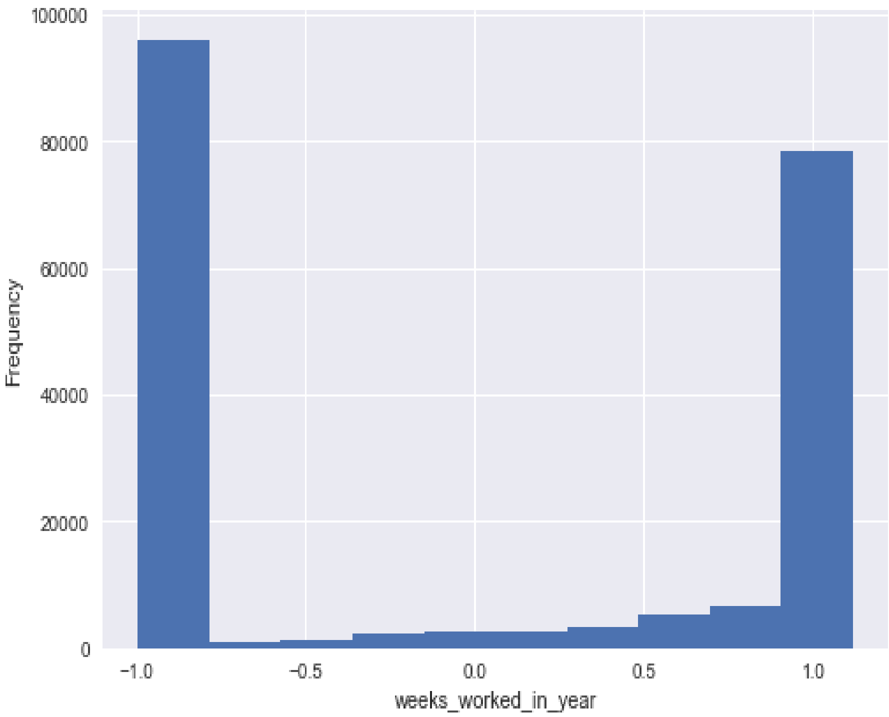


Data Normalisation

Before Normalization

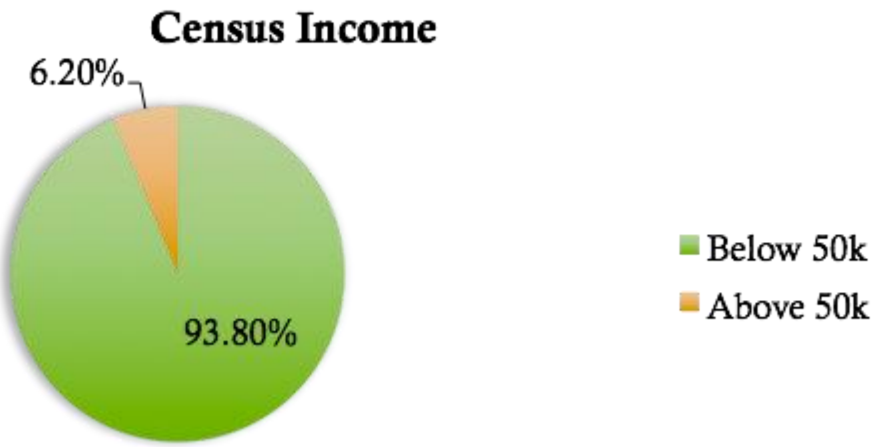


After Normalization



The Data is Imbalanced

- The dependent variable has imbalanced proportion of the classes.



Danger of Imbalanced Classes

- ML algorithms struggle with accuracy because of the unequal distribution of dependent variable
- This cause the performance of existing classifier to get biased towards majority class.
- Accuracy is very high while AUC is very low

  Accuracy 0.94 VS AUC 0.58

Handling Imbalanced Data

- Down-sampling Majority Class – lose information
- Up-sample Minority Class – over-fitting
- Synthetic Data Generation(SMOTE)

Model Evaluation

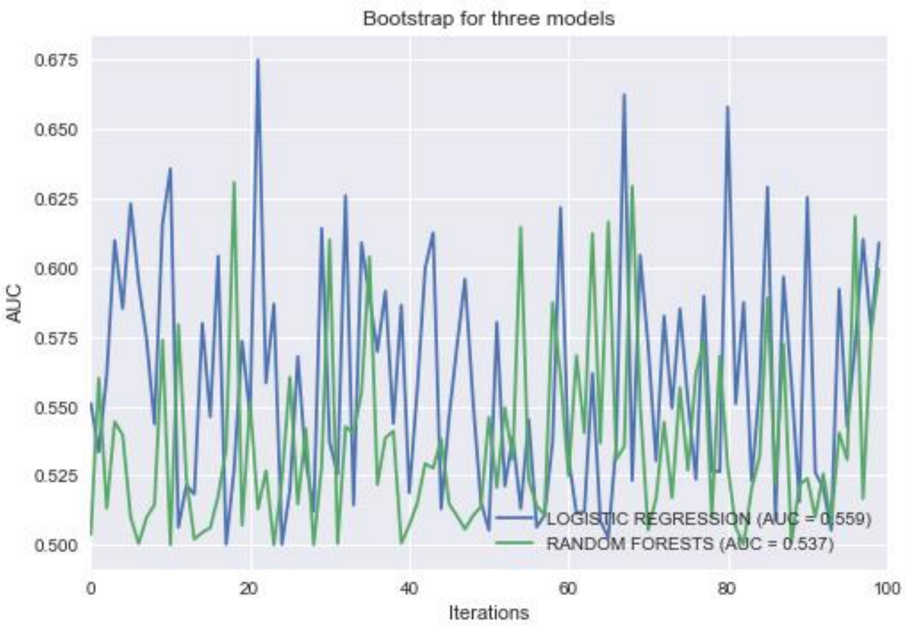
- k-fold Cross Validation
- Bootstrap
- Imbalanced vs Balanced Data

Model Evaluation

- K-fold Cross Validation

Model Evaluation

- Bootstrap



Accuracy:0.93

Accuracy:0.91

Feature Selection

- Forward Stepwise Feature Selection
- Backward Stepwise Feature Selection
- Random Forest for Feature Selection
- Boosting for Feature Selection

Feature Selection

- Sometimes, feature subsets giving better results than complete set of feature for the same algorithm.

Feature Selection

Reasons to use feature selection:

- It enables the machine learning algorithm to train faster.
- It reduces the complexity of a model and makes it easier to interpret.
- It improves the accuracy of a model if the right subset is chosen.
- It reduces overfitting.

Forward Stepwise

After applying forward selection, the best set of features obtained are :

['weeks_worked_in_year', 'dividend_from_Stocks', 'sex', 'age', 'capital_gains', 'capital_losses', 'num_person_Worked_employer']

Backward Stepwise

After applying backward elimination, the best set of features obtained are :

['sex', 'age', 'capital_gains', 'capital_losses', 'dividend_from_Stocks', 'weeks_worked_in_year']

Random Forest for Feature Selection

Feature ranking:

weeks_worked_in_year 0.16

dividend_from_Stocks 0.08

num_person_Worked_employer 0.08

major_occupation_code 0.08

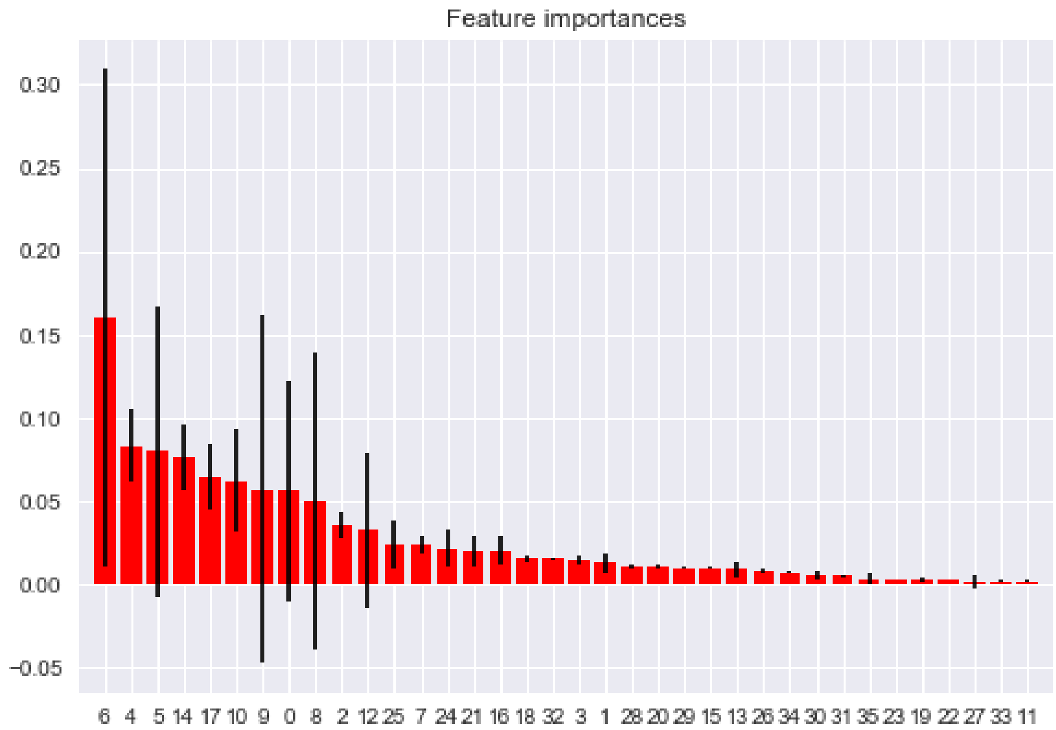
sex 0.06

education 0.06

occupation_code 0.06

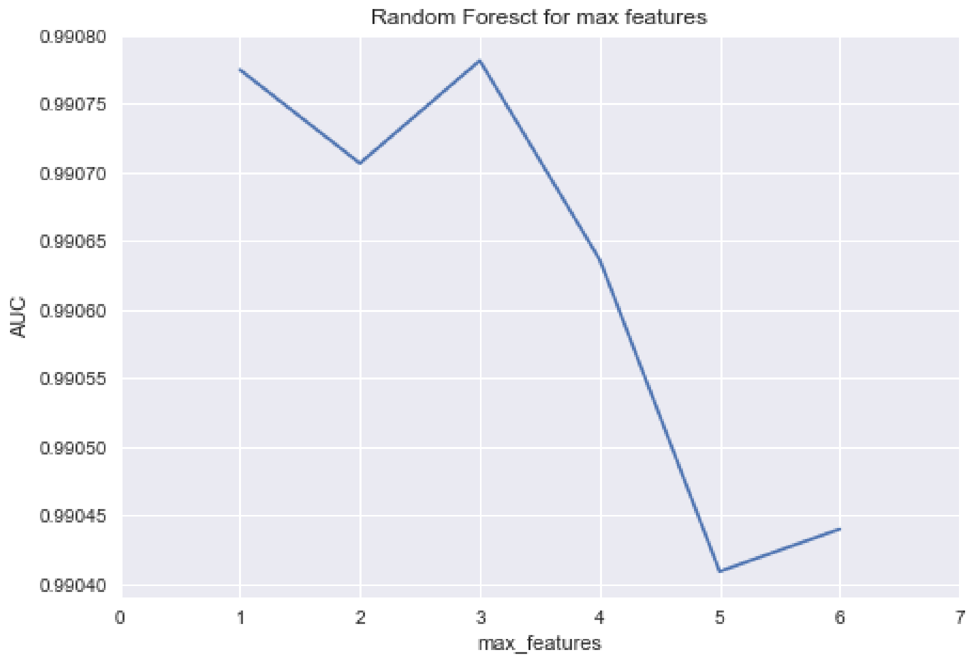
age 0.06

industry_code 0.05



Maximum Features

The max number of features for Random Forests is 3



Boosting for Feature Selection

Feature ranking:

major_occupation_code: 0.14

tax_filer_status: 0.14

weeks_worked_in_year: 0.12

d_household_summary: 0.06

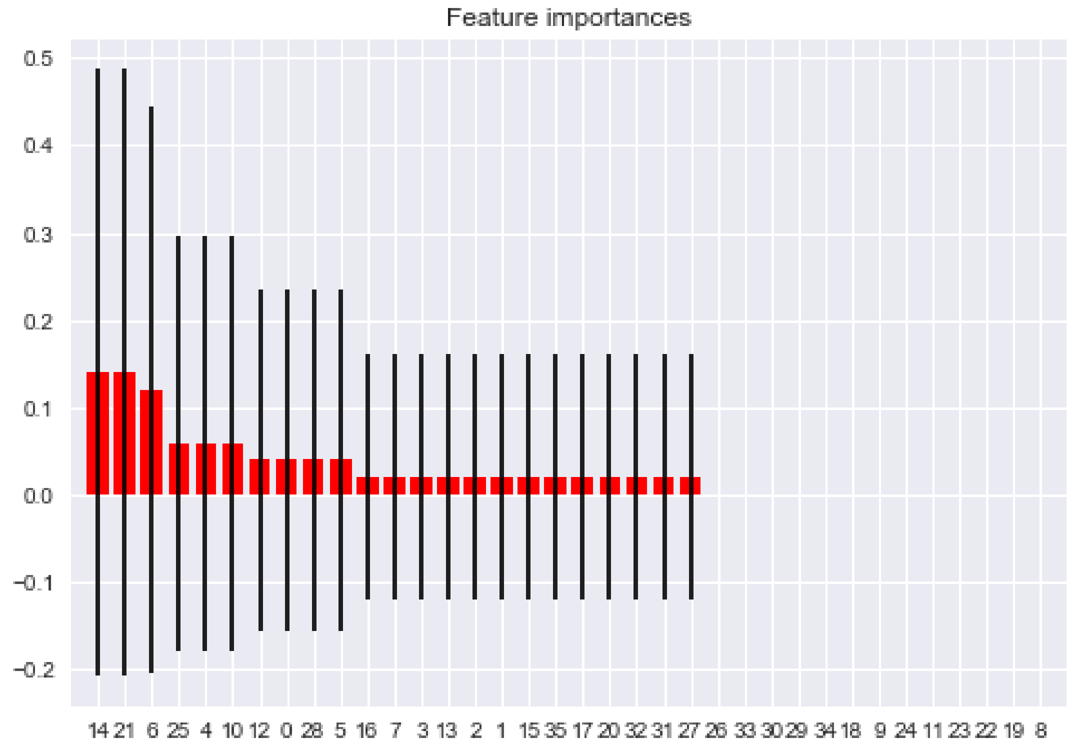
dividend_from_Stocks: 0.06

education: 0.06

marital_status: 0.04

age: 0.04

country_father: 0.04



Significant Features

1. weeks_worked_in_year
2. dividend_from_Stocks
3. major_occupation_code , education

Backward stepwise	Forward stepwise	Random forest	Boosting
weeks_worked_in_year	weeks_worked_in_year	weeks_worked_in_year	major_occupation_code
dividend_from_Stocks	dividend_from_Stocks	dividend_from_Stocks	tax_filer_status
num_person_Worked_employer	sex	num_person_Worked_employer	weeks_worked_in_year
major_occupation_code	age	major_occupation_code	d_household_summary
sex	capital_gains	sex	dividend_from_Stocks
education	capital_losses	education	education
occupation_code	num_person_Worked_employer	occupation_code	marital_status

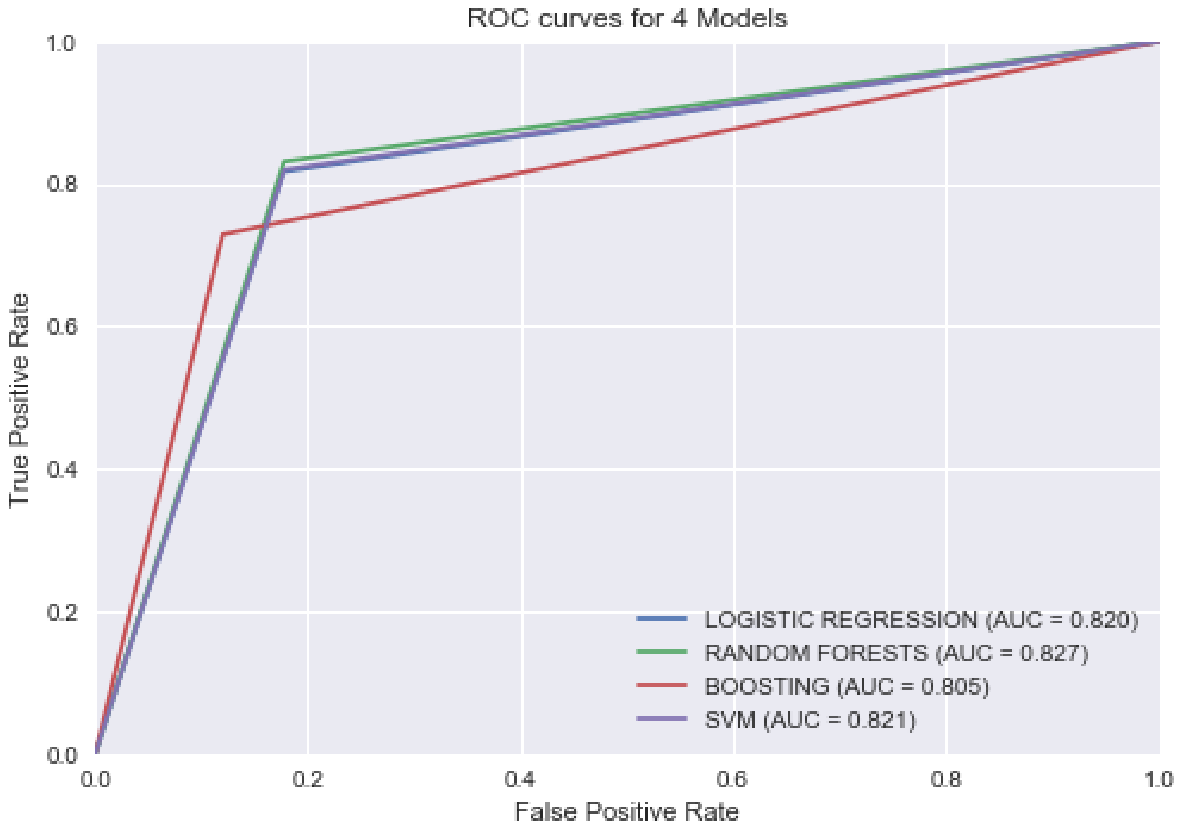
Model Prediction

- Logistic Regression
- Random Forest
- Boosting
- Support Vector Machine(SVM)

Summary

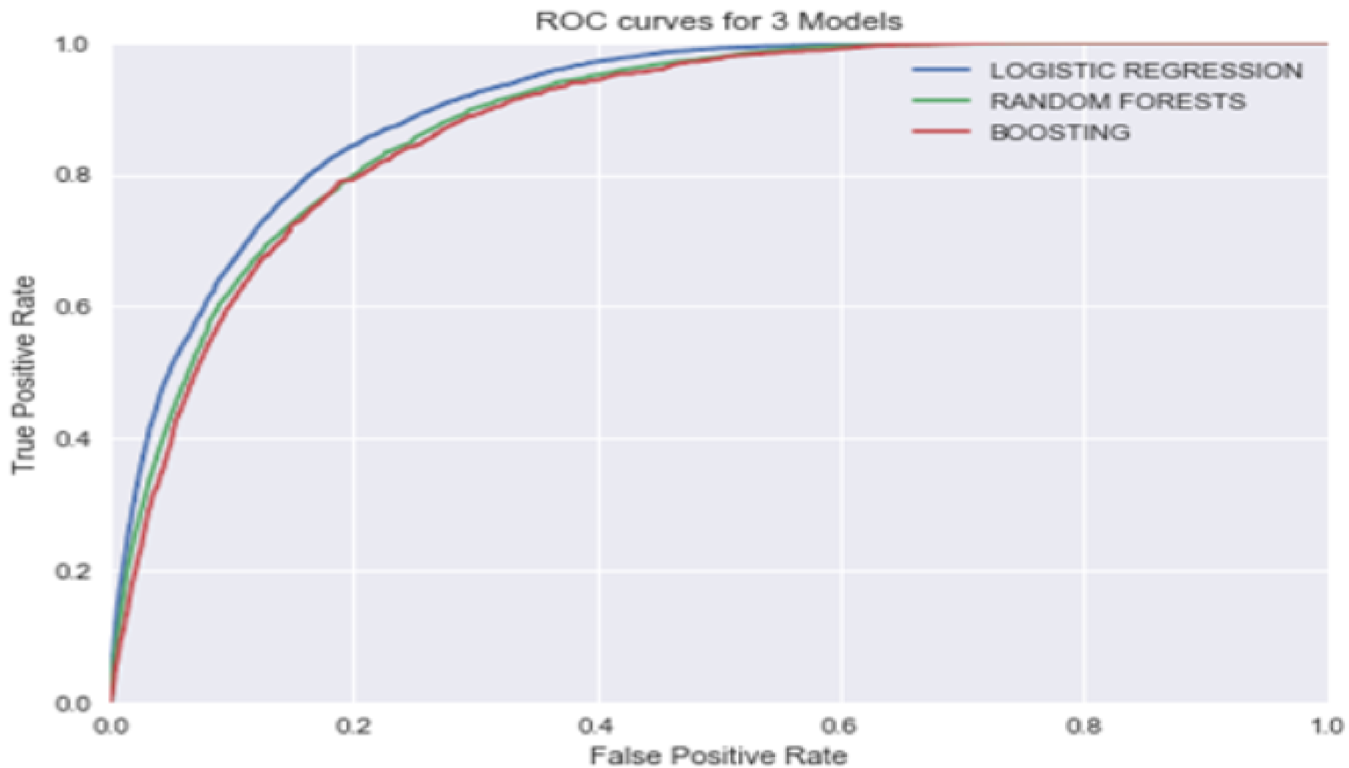
ROC Curve and AUC for Balanced Dataset

- Based on Down-Sampling:
  - Total Computational Time: 1 hour



Summary

- Based on SMOTE up-sampling:
  - Total Computational Time for Logistic Regression, Random Forests and Boosting: 2 hours



Summary

- Based on SMOTE up-sampling (for SVM):
  - Computational Time for SVM : 11+ hours

Accuracy: 0.87
Recall / TPR: 0.69996766893
Precision / FPR: 0.277332991738
AUC Score: 0.90
Classification Report:

	precision	recall	f1-score	support
0	0.98	0.88	0.93	93576
1	0.28	0.70	0.40	6186
avg / total	0.93	0.87	0.89	99762

Summary

- Balanced Dataset Comparison Results:
  - Best Model: Boosting

MODEL	ACCURACY	AUC
Logistic regression	**0.87**	**0.90**
Random forest	**0.94**	**0.89**
SVM	**0.87**	**0.90**
Boosting	**0.94**	**0.91**

Conclusion

- Work years, dividends, company size, age, education, occupation, and marital status (or relationship kind) are good for predicting income (above a certain threshold).

- Boosting has the best performance with a AUC of 0.91 and Accuracy of 0.94

Thank you