Adult Census Income Prediction & Analysis

Abstract

According to the United Nations, 71 percent of world's population live in countries where inequality has grown. While Income inequality between countries has improved, the inequalities within countries are the inequalities people feel day to day, month to month, year to year. We can use Machine Learning and Data Mining Techniques to predict a person's income based on certain attributes. Moreover, we can learn what features have a more significant impact on a person's income. This information can be used by the government to recognize focus areas and especially work on them to eradicate inequalities. In this work, we would train multiple Machine Learning models and compare them based on different evaluation metrics. Later we would use Explainable AI techniques to understand the significance of different features in the decision of the model. Explainable AI will help us know the reasoning behind model's decision. From its visualization we can understand how our target variable behaves against every feature individually and combined. This will make the model human interpretable and give us a detailed analysis that could help eradicate inequality.

Contents

List of Figures

List of tables

List of abbreviations

UCI – University of California Irvine

XAI – Explainable Artificial Intelligence

LIME - local interpretable model-agnostic explanations

SVM – Support Vector Machine

PDP – Partial dependence plot

Chapter 1 - Introduction

While income inequality between countries has decreased, income inequality within countries has increased. Today, 71 per cent of the world's population lives in countries with increasing inequality. This is especially important because inequalities within countries are the inequalities that people experience on a daily, monthly, and annual basis. This is how people compare and rank themselves to their neighbours, family members, and society. In most developed countries and some middle-income countries, including China and India, income inequality has increased since 1990.

Humans have grown increasingly reliant on data and information in society, and as a result, technologies for their storage, analysis, and processing on a large scale have evolved. Data Mining and Machine Learning have not only been used them to gain knowledge and discover new things, but also to uncover hidden patterns and concepts that have led to the prediction of

previously unattainable future events. We can try to use data mining techniques on a census dataset to know if the problem can be solved.

The data for our study were accessed from the University of California Irvine (UCI) Machine Learning Repository. It was extracted by Barry Becker using the 1994 census database. Our focus on this project is to build a Machine learning model that can predict the income class of a person (greater than or less than 50K$?). Later we use explainable AI techniques to explain the significance of each feature for the decision making model. The main goal is to make the model “Human interpretable”. By making the model interpretable we can use the obtained information to understand the income behaviour with different features.

Although we can understand which feature is important for the model training, it is difficult to get an idea on how the model behaves with different values of that feature. Let us say ‘age’ is an important factor for the prediction of income. XAI can help us observe the behaviour of the model with varying values of age, letting us know what values of age generally earn more thank 50K dollars and what values of age tend to earn less that 50K dollars. In this project we have leveraged the Skater, LIME and SHAP libraries for model interpretation. The model we have used here is random forest classifier, it performs the best among the tested models. We have done the process of data cleaning, data pre-processing and feature engineering before training and interpreting the model.

Chapter 2 - Literature review

- As per the work done in [3], a hyper parameter tuned with grid search based gradient boost classifier model performed with an accuracy of 88.16%. And a F1 score of 0.88. The most significant were selected for training, based on the Extra tree classifier scores for the attributes of the dataset. The process is further followed by handling missing categorical values by assigning a unique category. Categorical

variables were encoded in two stages, using label encoding and one-hot encoding in first and second stage respectively.

- According to [7], the dataset involves multiple different classification for similar instances. And when SVM is applied to six subsets of the datasets the highest accuracy received was 84.9%
- With development of Machine learning models the field have moved from black box models to white box models. We no longer only concern about the output and accuracy of a model. It is important to validate and understand the model to extensively apply the model to practical world. As introduced in [2] , SHAP is based on the cooperative game theory concept of Shapley values [4], and it takes additive feature importance into account. The Shapley value is defined as the mean marginal contribution of each feature value across all possible feature values in the feature space. SHAP can be used to interpret decision tree models based the calculated shapely values.
- Skater [1] is an open source unified framework that enables Model Interpretation for all types of models, allowing one to design an Interpretable machine learning system, which is typically required for real-world use-cases. Skater support both global and local interpretation of a black box model. Global interpretation is to understand the overall working of the model which includes partial dependence plots and feature importance. Local interpretation is to interpret the reasoning behind every prediction of the model. Local Interpretable Model Explanation (LIME) can be used for the local interpretation of the model. LIME uses local surrogate model to approximate the predictions of the underlying black box model.

Chapter 3 - Proposed Methodology

The dataset [5]

Attribute Name	Type	Description
age	Numerical	Represents age of the person
workclass	Categorical	Represents the nature of working class.

Education number	Categorical	Numeric representation of educational qualification. in the range (1 – 16)
marital status	Categorical	Represents the marital status of the person
fnlwgt	Numerical	It is the weight of population the sample represents
occupation	Categorical	Represents the profession of the person.
relationship	Categorical	Represents the relationship status of the person (Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried)
race	Categorical	Represents the race of the person
sex	Categorical	Represents the gender of the person – male or female
capital gain	Numerical	The total capital gain for the person
capital loss	Numerical	The total capital loss for the person
hours per week	Numerical	Total hours spent working per week
country	Categorical	The country where the person is residing
income label (Target)	Categorical	The class label column is the one we want to predict (Income <= $50K & Income > $50K)

1. In the first stage i.e. EDA & Data cleaning: it was observed that the data was not evenly distributed.

```
Total number of records: 32561


Individuals making more than $50k: 24720
Individuals making at most $50k: 7841
Percentage of individuals making more than $50k: 75.92%
```

There is an imbalance between the classes of classifications. Since the individuals making more than 50k as income represent 75% of the data. The data can be oversampled for better results.

As for the null values, there are no *NaN* values in the dataset. But certain categorical features (‘workclass’, ‘occupation’ and ‘native country’) have unknown values marked as ‘?’

Works class	**count**
Private	22673
Self-emp-not-inc	2540
Local-gov	2093
?	1836
State-gov	1298
Self-emp-inc	1116

Federal-gov	960
Without-pay	14
Never-worked	7

Table - 1

This problem can be dealt by replacing each ‘?’ values with a new class – ‘unknown’. Also there are two variables representing the education level of an individual, one in categorical form and other in numerical form. The numerical can be used to easily train the model hence we drop the categorical feature representing the education level. Further we can see there spaces in entries of the features, 'workclass', 'marital.status', 'occupation', 'relationship', 'race', 'sex', 'native.country', 'income' we should remove them for better results.

When plotting the heat map of education class against income we can observe that the majority of the dataset's population belongs to the education classes of 9 and 10.Those with
Education levels between 14 and 16 often earn more than $50,000 per year, as opposed to those with lower education levels, who typically earn less than $50,000. This is actually logical, supporting the veracity. It was also discovered that income directly increases with increase in average work hours which is logically valid.

2. Data pre-processing :
   - Scaling, All features can be converted into a range [0,1], where a feature or variable's minimum and maximum values will be 0 and 1, respectively. The function uses the following equation :

$$X_{scaled} = x - x_{min} / x_{max} - x_{min}$$

   - Encoding: Label encoding associates a categorical variable with a numerical value. One other way of encoding is one – hot encoding, which is a more practical way to deal with the problem but since we want interpret the model the dummy variables created would create

hassle. The target variable is also encoded as 1 for >50K class and 0 for <50K class.

- Oversampling: as mentioned before the data is imbalanced with 75% of the target variable belonging to greater than 50K class. We can create multiple random copies of the minority class and add to the dataset for more balanced dataset.

3. Feature Engineering: Not all features contribute to the model. We can remove irrelevant features using the feature importance metric. In this project we decide to remove feature with less than 5% importance.
4. Model training & evaluation: It is expected that the random forest classifier and XGboost classifier would provide us with good accuracy. The best model out of four models will be selected and be tuned for hyper parameters using gridSearch. The evaluation of the model will be based on accuracy and F1 score. Based on the input, or training, data, accuracy is the metric used to indicate which model is best at finding links and patterns between variables in a dataset. Whereas the harmonic mean of recall and precision is known as the F1-score. The following formula combines recall and precision into a single number:

Accuracy = Number of correct predictions/ Total Predictions

F1-score = 2 x[(precision x Recall) / (precision + Recall)]

5. XAI : The validity of the model trained in the project can be tested using methods termed as “XAI”. It is a collection of techniques used to make a model human interpretable, i.e. to make a black box model transparent. Model interpretation can help us in two ways [6] :
   - Validity: the model can be tested on sensibility of the prediction, are the predictions logical? This can be done by observing the trend of the model against specific features. Partial dependence plot helps us in interpreting the behaviour of the model.

- Knowledge discovery: Model interpretation can help us identify new patterns in data that could not be observed with the traditional methods.

Model can be interpreted in two ways: globally and locally. Global interpretation involves studying the behaviour of the complete dataset w.r.t the dependent variable against independent variables. Whereas local interpretation involves studying a particular prediction for the reason and importance of each feature in the decision.

Chapter 4 - Implementation Details

We have used the following libraries for our project:

- *Numpy& pandas* : for calculations and handling arrays
- *Matplotlib&seaborn* for plotting bar graphs, heat maps
- *Sklearn*

  Sklearn has many tools for data processing, the scaling requirement was satisfied using the *MinMaxScaler* function that scales the numerical variables from 0 – 1. Other such feature is the label encoder which was used to convert the categorical variables to numerical variables. The data is also oversampled using the ***RandomOversampler( )*** function from the *imblearn* library.

Training models: For identification of a suitable model, the project implements the following models, *Logistic regression, AdaBoost, Random forest classifier* and*XGboost classifier*. The best two were selected and tested upon processed data. All the models are a part of sklearn library. The data used to train the model is split into two parts, the train and the test dataset using *train_test_split* function.

```
LogisticRegression:
Accuracy: train:  0.8113224061811017 | test:  0.8016711708227036
F1-score: train:  0.5129439298733482 | test:  0.4468873793984828
----------------------------------------------------------
RandomForest:
Accuracy: train:  0.999969265640896 | test:  0.8022272831990588
F1-score: train:  0.9999362117765965 | test:  0.5286852948072358
```

```
------------------------------------------------------------
AdaBoost:
Accuracy: train:  0.8635244963849609 | test:  0.8054235857200354
F1-score: train:  0.6833630883833939 | test:  0.521592391903771
------------------------------------------------------------
XGboost:
Accuracy: train:  0.9106708069504968 | test:  0.8068375941223149
F1-score: train:  0.8021799955247723 | test:  0.5438673991716304
------------------------------------------------------------
```

Feature engineering is also done on the chosen best model using the inbuilt feature of feature importance. Features with importance of less than 5 percentage is dropped.

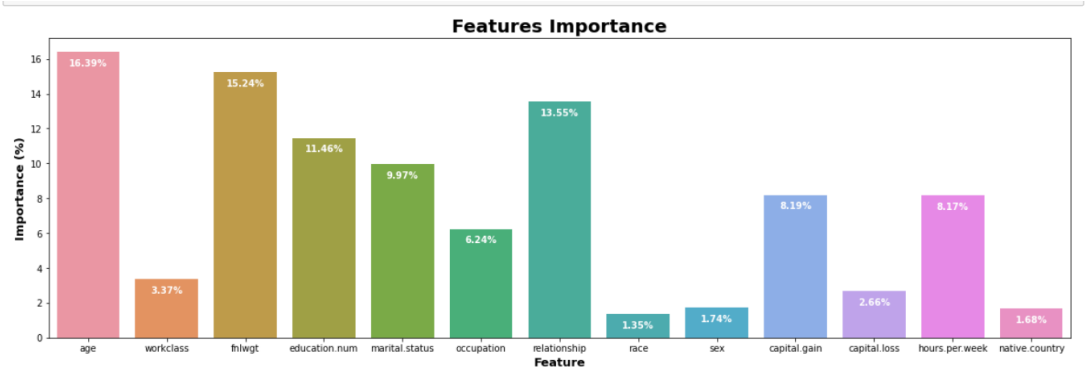


Fig - 1

Evaluation metrics: the evaluation of the trained model is done by measuring the accuracy and the f1 score. The metrics is calculated for bot training and testing datasets. The models are compared to each other based on this scores (the higher the better). The best model is retrained based best hyper parameters chosen through the *gridsearch* method available in sklearn library. We also limit the *max_depth* for random forest classifier for feasible timings while interpreting the model. Model trained with no limit on maximum depth of trees would give the maximum accuracy but will take longer time for interpretation.

XAI: random forest classifier came out to be the best model with an accuracy of 0.918 percent and and f-1 score of 0.924. We choose the same model but with the maximum depth for trees limited to 20. This is done in order to execute the *shap* model in feasible time. XAI is implemented using the following libraries:

- *Skater*
- *LIME from skater*
- *SHAP*

Skater serves the purpose of global interpretation and *LIME* (local interpretable model-agnostic explanations) as the name suggests helps us with local interpretation. Whereas *SHAP* can do both for us. Multiple libraries are used to verify if they produce similar trends.

Chapter 5 - Result and analysis

Training models: The best models observed before oversampling was done were, *Randomforest* classifier and *XGboost* classifier. After oversampling was done *RandomForest*classifier received a significant lead in terms of both accuracy and f1score.

Random Forest Classifier

```
Accuracy: train:  0.9999746944353063 | test:  0.9092864059984107
F1-score: train:  0.999974695203694 | test:  0.9176806638982342
```

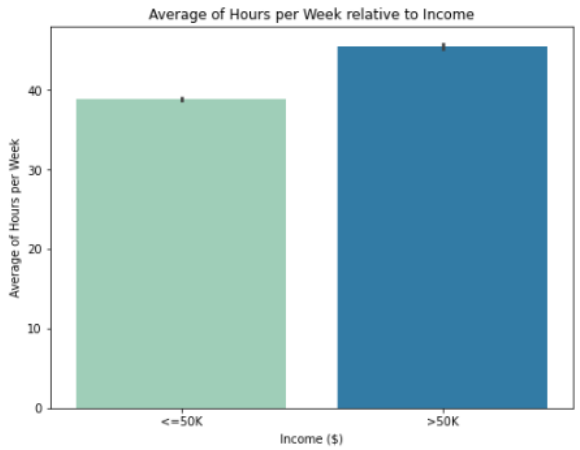
XG boost

```
Accuracy: train:  0.8966718093906281 | test:  0.8424385489524461
F1-score: train:  0.8997226566110379 | test:  0.8538633614509783
```

After application of feature engineering and tuning of hyper parameter(bootstrap='false', criterion='gini'), the results of Random forest classifier was further improved as:

```
Accuracy: train:  0.99996963319429 | test:  0.9181323180524836
F1-score: train:  0.9999696340907812 | test:  0.9239662596751919
```

Other analysis:



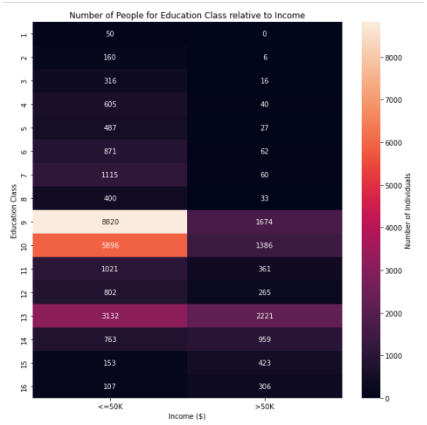


Fig - 2Fig - 3

We can see that revenue rises in direct proportion to the average number of labour hours each week, which is a sensible and logical consequence. The heat map above shows that persons with education levels of 9 and 10 make up the majority of the dataset. Also, those with an education level of 14 to 16 typically earn more than $50,000 in the numbers we have in the dataset, but those with a lower education level typically earn less than $50,000. This makes sense and confirms the authenticity of data.

For Interpretation the following model was used:

RandomForestClassifier(bootstrap='false', criterion='gini', max_depth=20)

```
Accuracy: train:  0.9675176447269983 | test:  0.902827244066156
F1-score: train:  0.968473173107965 | test:  0.9106081254136432
```

Results from *skater* global interpretation:

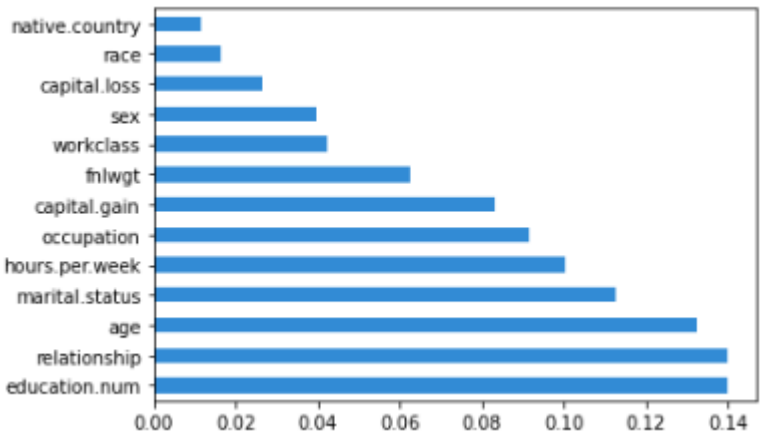
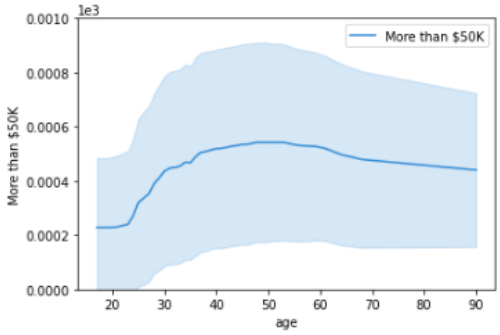


Fig - 4

The above graph shows the importance of each feature for the model, one significant change from the default feature importance metric is the importance of “fnlwgt”. As mentioned before it is the variable suggesting the number of population the row represents. This makes sense as it should not effect a person’s income as much as categories like education and age does.



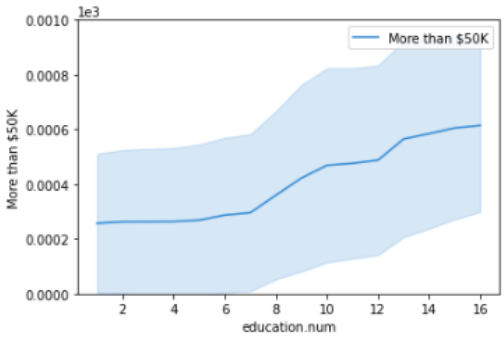


Fig - 5Fig – 6

The above graphs represent the partial dependence plot (PDP) received on interpretation through skater. The graph shows us the following patterns:

The first plot shows the relation between target variable and age. It can be observed that the middle aged people have the highest chance of earning more than 50K. This is logically valid and hence confirms the validity of our model. The second plot shows an obvious pattern that higher education results in more chances of earning >50K $.

RELATIONSHIP	
1	Not-in-family
4	Unmarried
3	Own-child
2	Other-relative
0	Husband
5	Wife

Table - 2

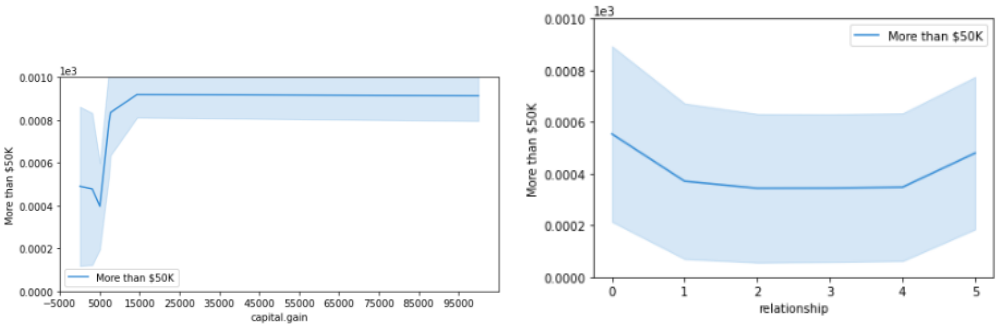


Fig - 7Fig - 8

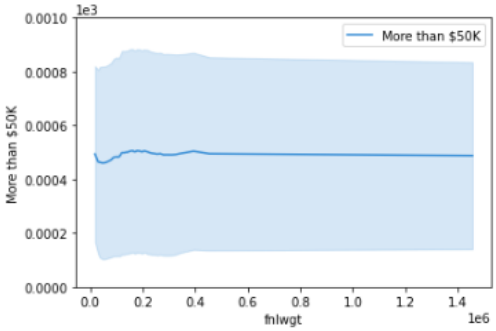


Fig – 9

On plotting PDPs for more features, we observe that for capital gain there is a sharp rise in chances of earning more than 50K, around 5000 – 8000. Where as in case of relationship it's very interesting that married people (husband and wife) have a better possibility of generating more money than others.

Partial dependence plots for two variables plotted gives us the following results:

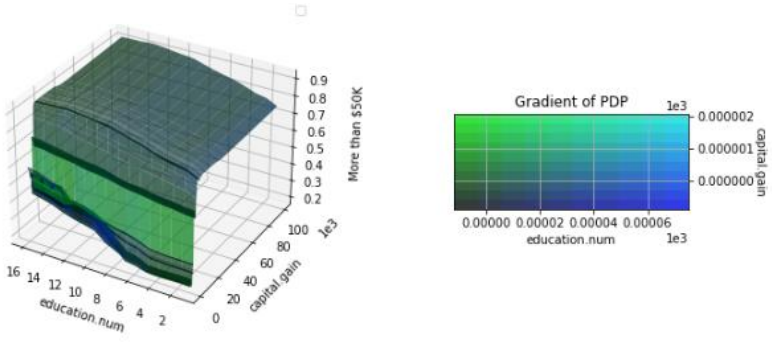


Fig - 10

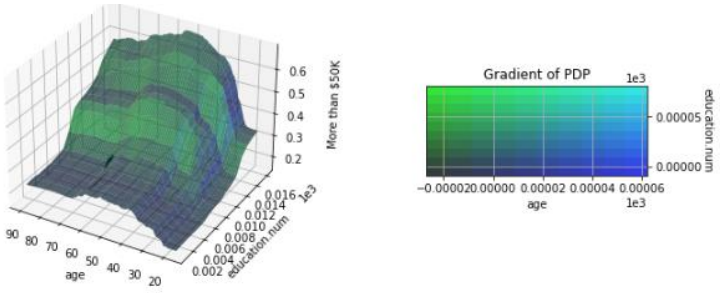


Fig – 11

We can see that higher education and more capital gain leads to higher chances of the person being in “>50K” class. Another rinteresting observation can be made that middle aged people with the highest education have the best cahnces of being in the “>50K” class.

Results from *LIME* local interpretation:

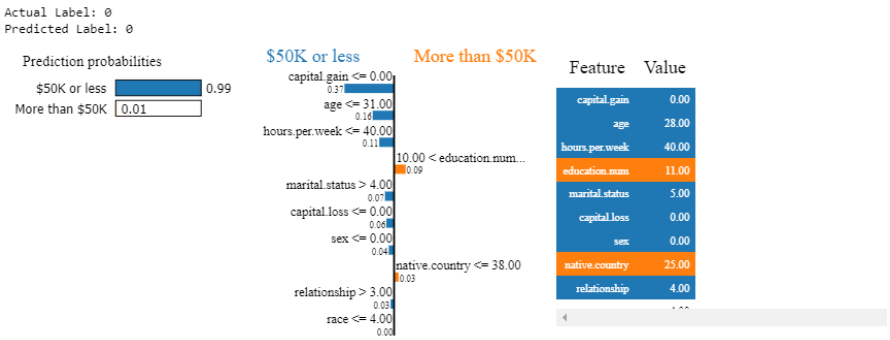


Fig- 12

The above results show us that in deciding one of the target, whose actual value was class ‘0’ and predicted was also class ‘0’ , capital gain being <=0 has the highest part followed by age and hours per week. And given below is an example of predicted value (and actual) of target class as ‘1’.

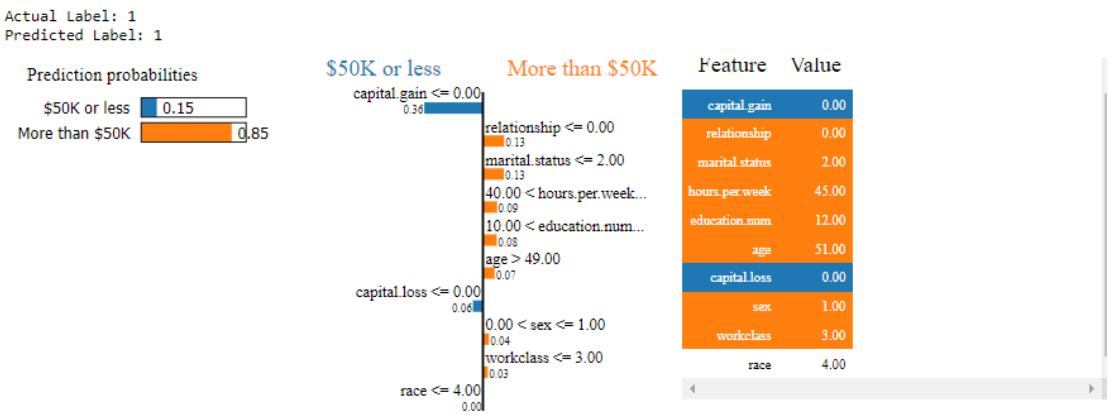


Fig - 13

Results from *SHAP* :

From the figure below, It is worth noting that the age and married status variables have a bigger overall model impact than the capital gain feature, but capital gain has a greater influence in samples where it counts, than age or marital status. In other words, capital gain has a significant influence on a few projections, whereas age or marital status have a less impact on all forecasts.

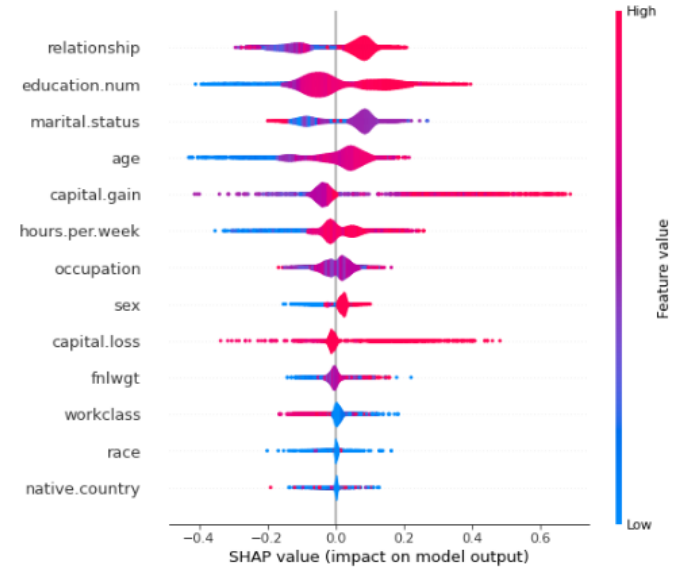
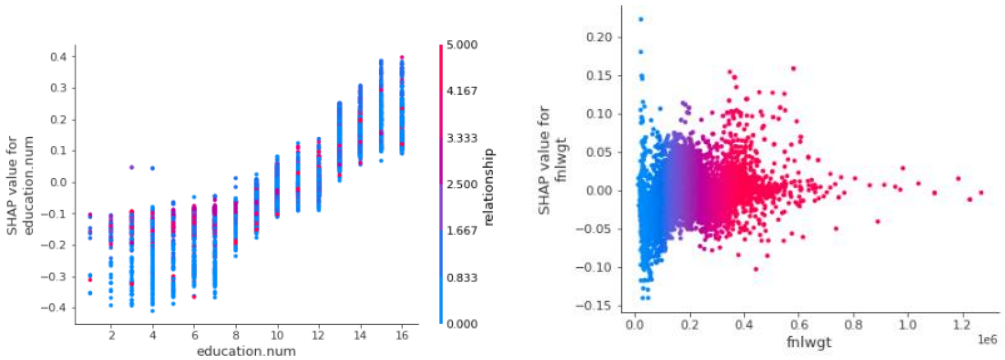


Fig - 14



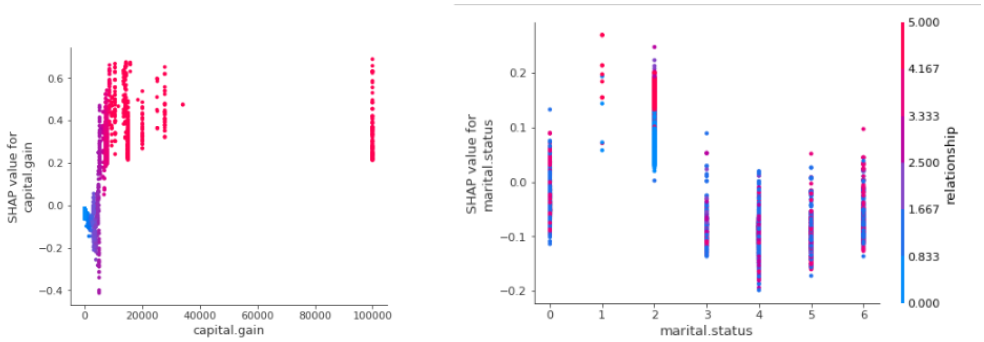


Fig - 15

When using a different mode of interpretation, SHAP values: we receive more or less the same results (from *shap* based PDPs) received from previous observations of PDPs created using *skater*.

- Higher education levels and husband or wife (married) people have the best chances of producing more money.
- Capital gain have a very different trend, a sudden rise of chances is seen at certain value.
- Final weight have no significant trend.
- Married people with relationship status of either husband or wife having the highest chance of making more money.

Shown below is a local interpretation based on *shap values.* The features shown in blue lower the chances of prediction to be ‘1’ and the features shown in red supports it. The features in red pushes the prediction from its base value (mean) to near ‘1’ leading the prediction to lie under “>50K” class.

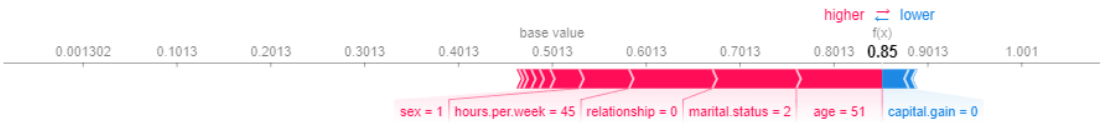


Fig – 16

Chapter 7 - Conclusion& Future work

Following the data cleaning, pre-processing, and feature engineering processes, the Random Forest classifier emerges as the best model among those evaluated, with an accuracy of 0.918 and an F1-score of 0.924. When we interpret the model, we get a lot of fresh information about the data, and it also helps us corroborate the rationale behind our trained black box model. Because the explanations are reasonable, we may argue that the model can be relied on for predictions. More work may be done to enhance the model's interpretation execution time. The final weight feature ('fnlwgt') is a perplexing feature; work may be done to harness the knowledge gathered through interpretation to better comprehend the feature and increase accuracy.

Chapter 6 - References

[1]*Overview — Skater 0 Documentation*. oracle.github.io/Skater/overview.html.

[2] Rodríguez-Pérez, R., Bajorath, J. Interpretation of machine learning models using shapley values: application to compound potency and multi-target activity predictions. *J Comput Aided Mol Des* 34, 1013–1026 (2020). https://doi.org/10.1007/s10822-020-00314-0

[3] N. Chakrabarty and S. Biswas, "A Statistical Approach to Adult Census Income Level Prediction," 2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), 2018, pp. 207-212, doi: 10.1109/ICACCCN.2018.8748528.

[4] SHAP: Shapley Additive Explanations https://towardsdatascience.com/shap-shapley-additive-explanations-5a2a271ed9c3

[5] Dua, D. and Graff, C. (2019). UCI Machine Learning Repository

[http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of

Information and Computer Science.

[6] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16). Association for Computing Machinery, New York, NY, USA, 1135–1144. https://doi.org/10.1145/2939672.2939778

[7] Lazar, Alina. (2004). Income prediction via support vector machine.. 143-149. 10.1109/ICMLA.2004.1383506.