Anirudh Chintaluri, Dev Makwana

Dr. Yilmaz, Period 4

Machine Learning 1

8 October 2024

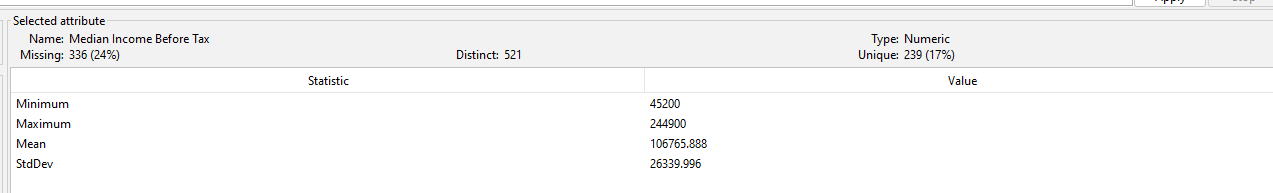
**Quarter 1 Project: Intermediate Report**

At this point in our project, we have completed the steps to preprocessing, and got started with training our 20 classifier models.

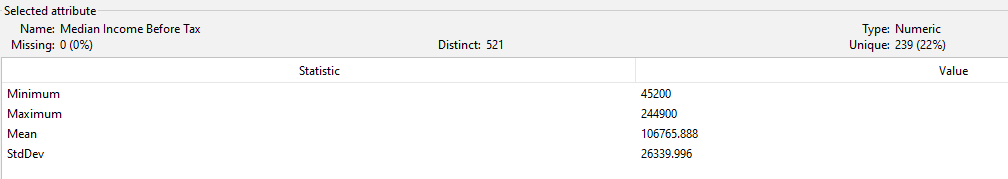
**Preprocessing:**

The first step in this project is to handle missing values and remove unnecessary attributes. First, we had to make the Median Income Before Tax the class variable in weka, by making it the last column. Since we plan to run a supervised model, we removed all instances that had a missing value for the class. To do this, we ran a RemoveWithValues filter, with matchMissingValues as true.

Before:



After:



Removing these instances brought the total instance count from 1406 down to 1070.

Now we will fill in missing values for the rest of the attributes, or remove the attribute entirely if there are too many missing values. We moved all attributes with 80% or more missing values, because replacing the missing values will not produce very accurate data. The attributes removed were:

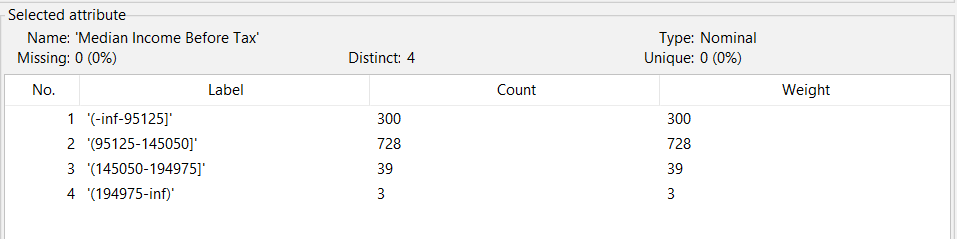
* Range\_of\_Gross\_Earnings\_from\_Self\_Employment\_£
* Range\_of\_Total\_Expenses\_from\_Self\_Employment\_£
* Range\_of\_Income\_Before\_Tax\_from\_Self\_Employment\_£
* Range\_of\_Total\_Income\_Before\_Tax\_£
* Average SE Gross Earnings
* Average SE Expenses,
* Average SE Income Before Tax
* Average Emp Gross Earnings
* Average Emp Expenses
* Average Emp Income Before Tax
* %Zero Office and General Business
* %Zero Premises
* %Zero Other
* Count\_of\_GPs
* Percentage\_of\_GPs\_%
* Cumulative\_Percentage\_of\_GPs\_%
* GE Median
* GE Q1
* GE Q3
* GE D1
* GE D9
* TE Median
* IBT Q1
* IBT Q3
* IBT D1
* IBT D9

After removing these attributes, 29 were left (Excluding the class).Weekly Working Hours was also removed because it was a nominal attribute with only one distinct value. Since there is only one possible value, it has no effect on the class variable.

Of the remaining attributes, Practice\_Type, Gender, Age, Rurality, Region, and Practice\_Registered\_Patients all had a nominal value “All,” which seems to be a disguised missing value. In the context of any of these attributes, it does not make sense, so we treated them as missing values. Rurality, Region, and Practice\_Registered\_Patients were removed because the “All” value accounted for 80% or more of the values in each attribute. The following is a list of the remaining attributes:

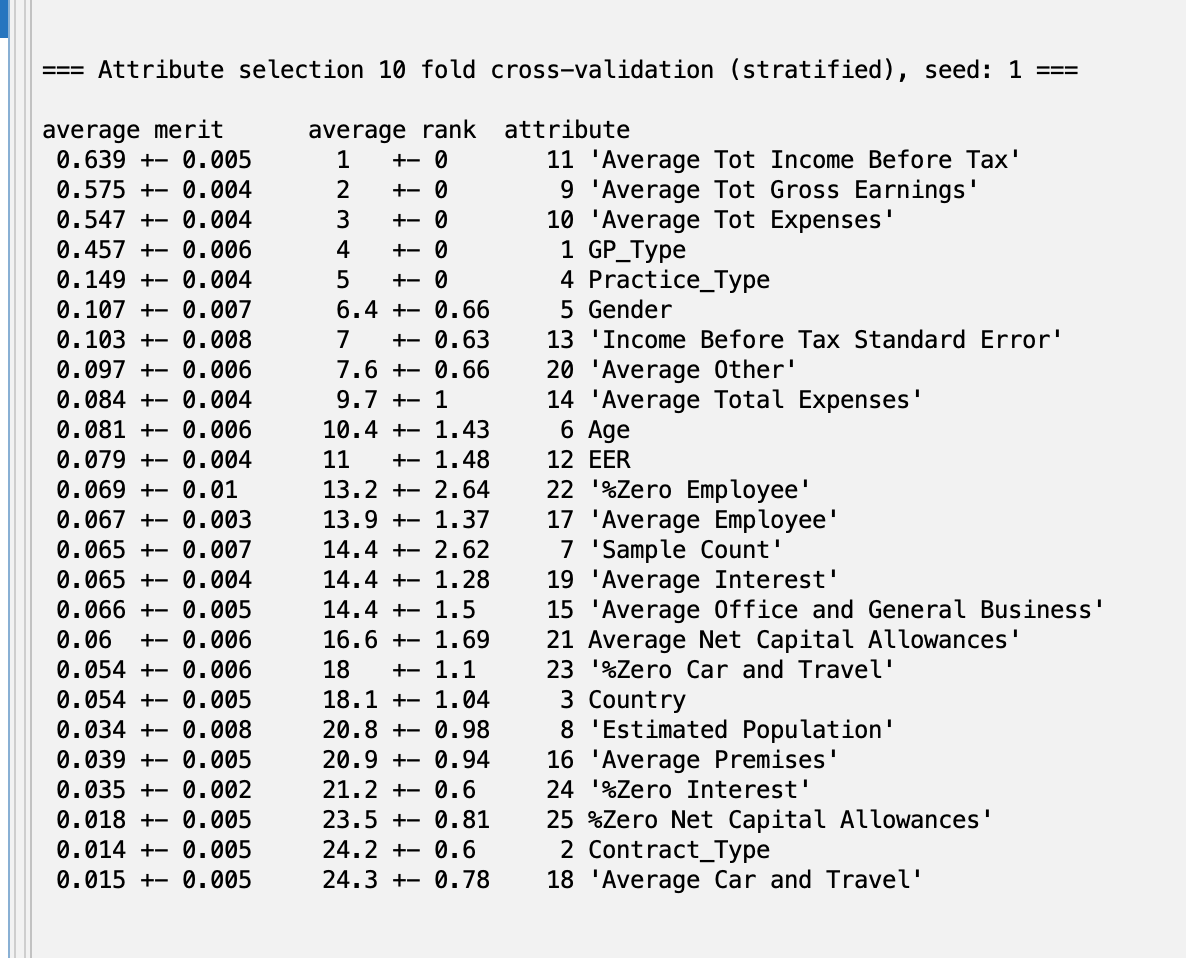
* GP\_Type
* Countract\_Type
* Country
* Practice\_Type
* Gender
* Age
* Sample Count
* Estimated Population
* Average Tot Gross Earnings
* Average Tot Expenses
* Average Tot Income Before Tax
* EER
* Income Before Tax Standard Error
* Average Total Expenses
* Average Office and General Business
* Average Premises
* Average Employee
* Average Car and Travel
* Average Interest
* Average Other
* Average Net Capital Allowances
* %Zero Employee
* %Zero Car and Travel
* %Zero Interest
* %Zero Net Capital Allowances

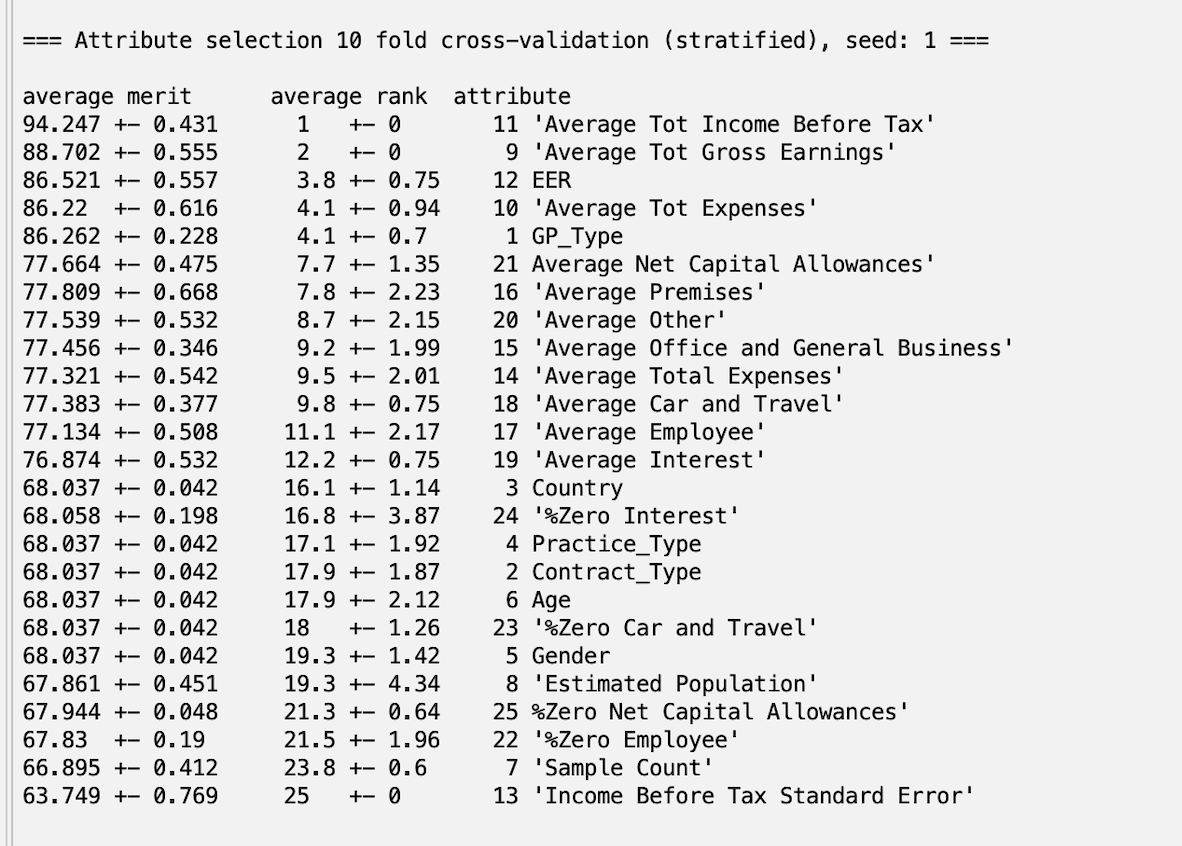
In order to run a classification model, we discretized our class variable into four bins of equal width.

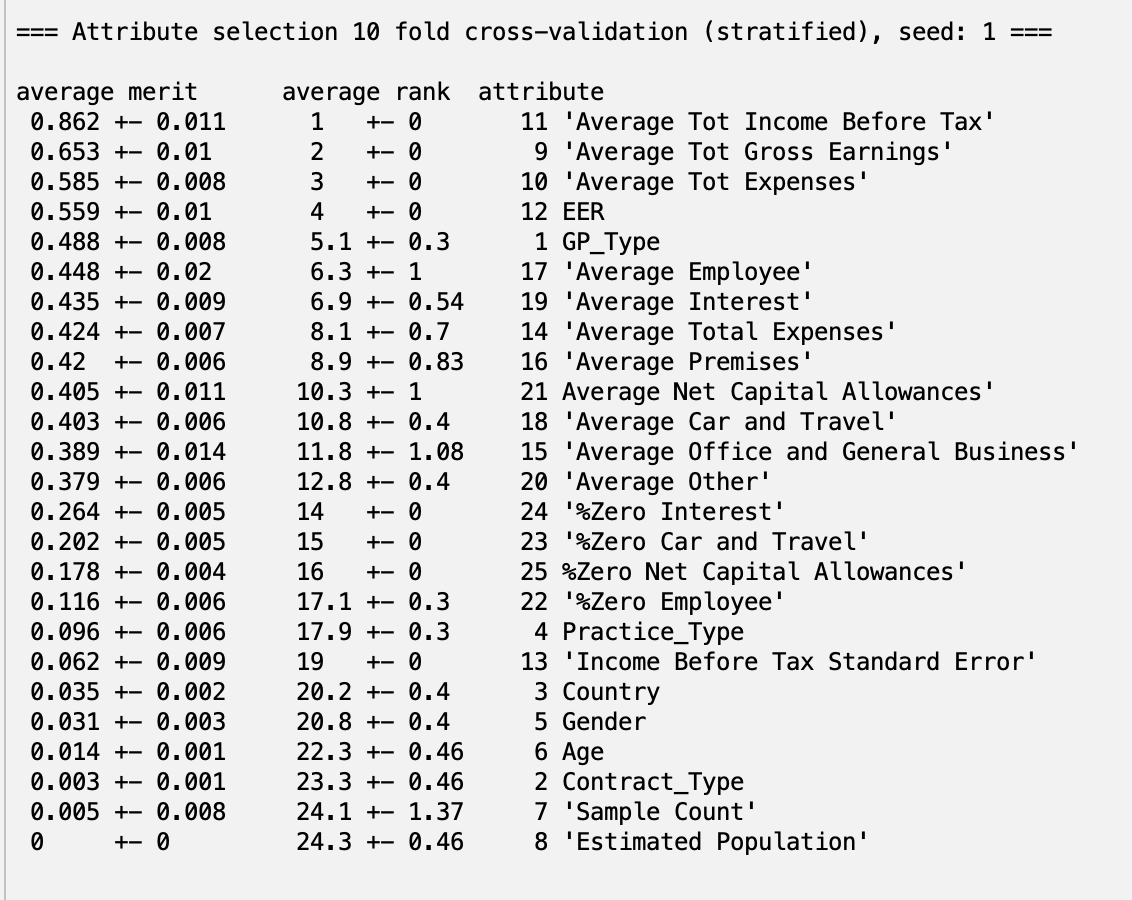


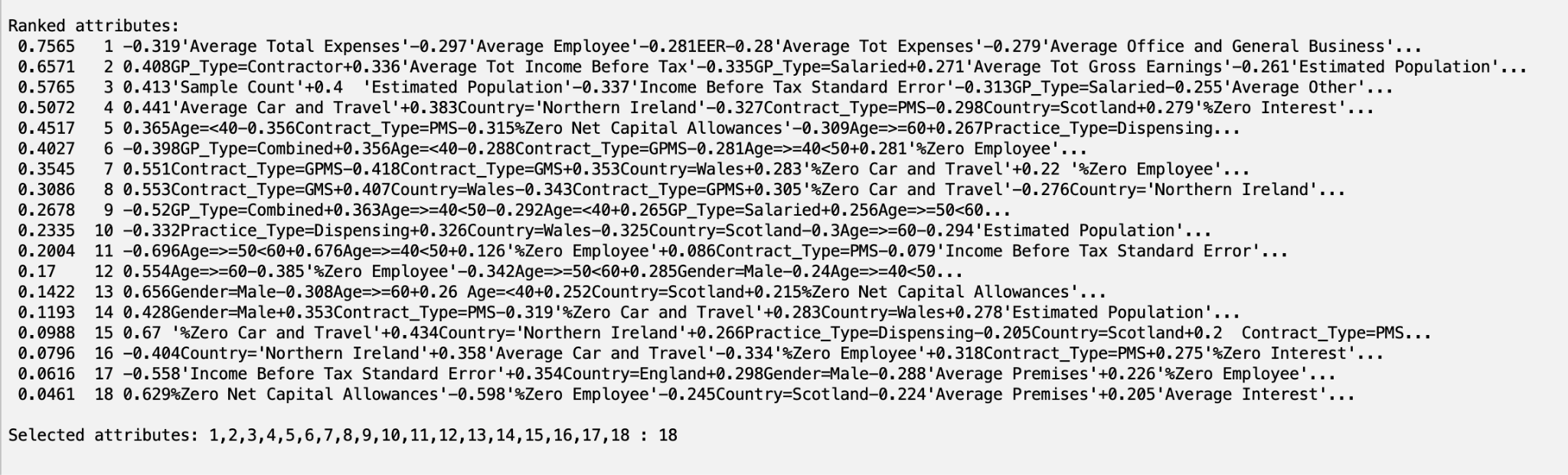
Before getting to attribute selection, we worked towards normalizing the data for each of the datasets, except for the PCA. Firstly, for all continuous data we normalized it by z-score range using the Standardize filter (filters > unsupervised > attribute > Standardize).

**Attribute Selection:**

We ran four different algorithms: CorrelationAttributeEval, OneRAttributeEval, InfoGainAttributeEval, and PrincipalComponents. We ran all of these algorithms with 10-fold cross validation, while setting **Median Income Before Tax** as the class attribute. The snapshots below reveals our results from using Correlation Analysis, OneR Evaluation, Information Gain, and PCA respectively:







For the first three attribute selection algorithms, we set our thresholds to be the following, for which all attributes with lower values have been dropped:

1. **Correlation Analysis - 0.05.**
2. **OneR - 68.**
3. **InfoGain - 0.05.**

We have also created a separate dataset for our PCA results.

For our final custom attribute selection, we thought about the dataset from an economic standpoint. What we saw was that **Average Tot Gross Earnings**, **Average Tot Income Before Tax,** and **Income Before Tax Standard Error** are too closely related to the class attribute, **Median Income Before Tax**. Because of this, these three attributes will be weighed more heavily than any other attribute as the correlation is obvious conceptually and even from a correlation standpoint. For that reason, we chose to remove these three attributes to prevent “spoiler attributes” from inflating our accuracy.

The attribute selection datasets are all organized in the folder linked here:[**Attribute Selection Datasets**](https://drive.google.com/drive/folders/16wYuCyObpCSg_HSGNrFvLLvlHx8Ggj8u?usp=share_link)

To prepare the datasets for train-validation-test splits, we began by randomizing the dataset. This is done to avoid any bias introduced from the raw dataset. We split the dataset by using the Resample filter (filters > unsupervised > instance > Resample) and setting the sample size percentage to 70.0 (for 70% of the dataset). This new dataset is our training dataset. By hitting “Undo,” we return to the original dataset so we can retrieve the other 30% for validation and testing. This time, we set invertSelection to True to avoid overlap of data.

From the rest of the 30% of the data, we repeated the process of the first split by having 50% of that set go towards validation, and 50% towards testing. Files of the train-test-splits for each type of attribute selection dataset are organized in folders, by attribute selection method, here: [**Train-Validation-Test Datasets**](https://drive.google.com/drive/folders/1iXQaNh4IKAh3VZ4VsHMuXi3Kl6CBGVuj?usp=share_link)

From this point onwards, we compared performance of each of the datasets on four different models: Naive Bayes, Random Forest, OneR, and Multilayer Perceptron (MLP). We stuck with default settings for each model. We were able to determine model results using 10-fold cross-validation on Weka. The data table below displays the performance of these models:

|  | Naive Bayes | Random Forest | OneR | MLP | Average |
| --- | --- | --- | --- | --- | --- |
| Correlation | 77.0093 % | 96.1682 % | 94.486 % | 92.8972 % | 90.2103% |
| OneR | 74.2056% | 96.6355% | 94.486% | 92.7103% | 89.1589% |
| Info Gain | 73.9252% | 95.8879% | 94.486% | 92.0561% | 85.5841% |
| PCA | 85.0467 % | 93.5514 % | 82.1495% | 92.5234 % | 88.9252% |
| Custom | 70.2804% | 94.2991% | 86.3551% | 90.0000% | 70.2804% |

Screenshots of the performance measures are organized in folders for each dataset here: [**Model Performance Data**](https://drive.google.com/drive/folders/1uMyTdkRRjSV_kRgZNPeO5bXEedzABx9m?usp=drive_link)