Predicting Wage based on Census Data

Sage Elfanbaum

Computer Science
UMSL
12/18/2020

1 Introduction

1.1 What is the census

The census is a servery given to each person currently living in the united states every ten years. This data is used to determine many things from how many representatives each state gets to how much everyone is taxed.

1.2 Why did I choose this project

I chose this project to because of the interesting data set I found comparing age, hours worked, and sex, and many more categorizes, with how much money they make.I was interested not only if these factors could predict income level but what parts of my data set would be most important.

2 Data set

My data set is census data from 1996 and it aimed to predict weather income exceeds 50k per year based on this data. Age, sex, and hours worked per week are some of the inputs with the output being if the person makes more than 50k per year or not. I found this data set at the Machine Learning Repository. The input features are the following fields:

- sex
- workclass
- fnlwgt
- education
- education-number
- marital-status
- occupation
- relationship
- race
- sex
- capital-gain
- capital-loss
- hours-per-week

3 Data Normalization

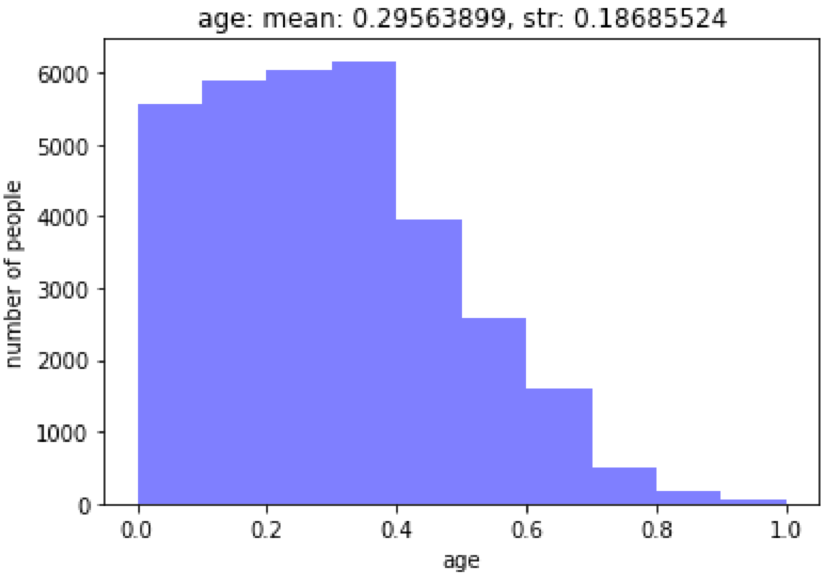
Before I could use the data I needed to normalize it. As you can see from the graphs the data was not distributed uniformly. I needed to normalize the data to use it for the network. This normalization makes the optimization of our model less sensitive to the scale of the features. The equation I used is below:

$$Xnew = \frac{X - Xmean}{Xmax - Xmin}$$

3.0.1 Data Spliting

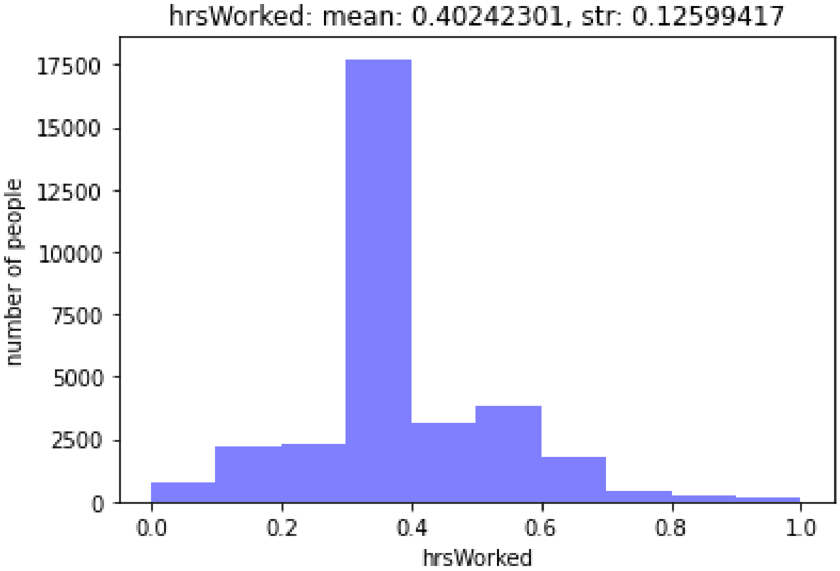
When training my Network I split my dataset doing a 30-70 split. I trained the model with a few different combinations and found that this split was the best for my project.

4 graphs



4.1 age

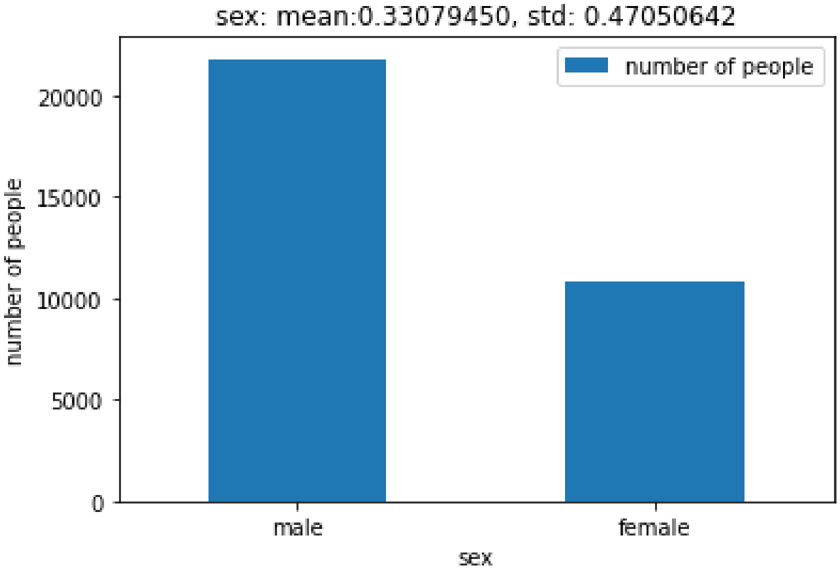
As you can see age is quite heavily left centred. This is because the census starts counting people at age 18 so and as people age they die. So we should expect this graph to look like this.



4.2 hours worked

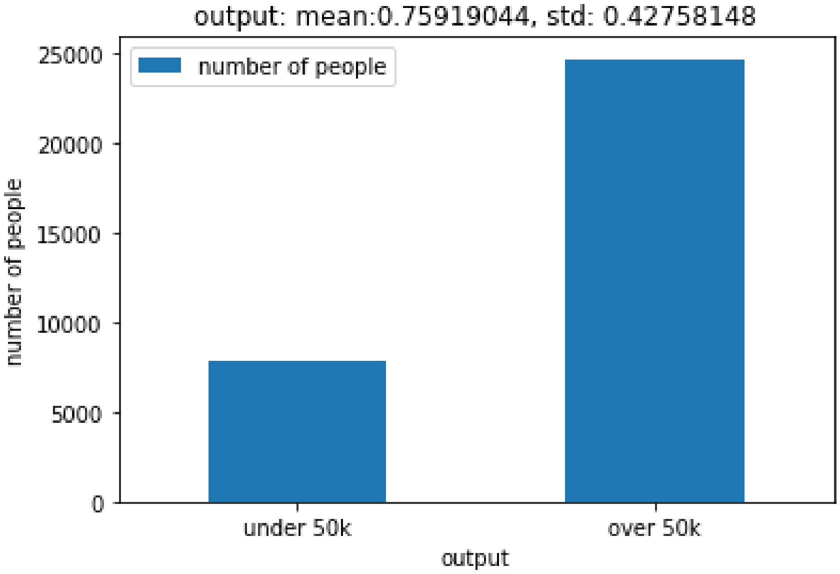
this is centered on 40 hours worked like we would expect.

4.3 sex



As you can see there is an over representing of males in the data set. I am unsure why this is the case but here it is.

4.4 output



As you can see most people surveyed were making above 50k per year.

5 Comparing logistic regression to multi-layered model

I needed to know what size network would work best to analyze my data. In order to do this I tested a one two and three level network as well as many sizes of the levels. I will be using the three level network as it creates the best model of my data. I use the 8-4-1 node setup. Below in table 1 are my test results.

Table 1: Size of network test

Logistic regression	
mean:	0.21394
STD:	0.24875
Two level network	
mean:	0.20487
STD:	0.24791
Three level network	
mean:	0.19585
STD:	0.24509

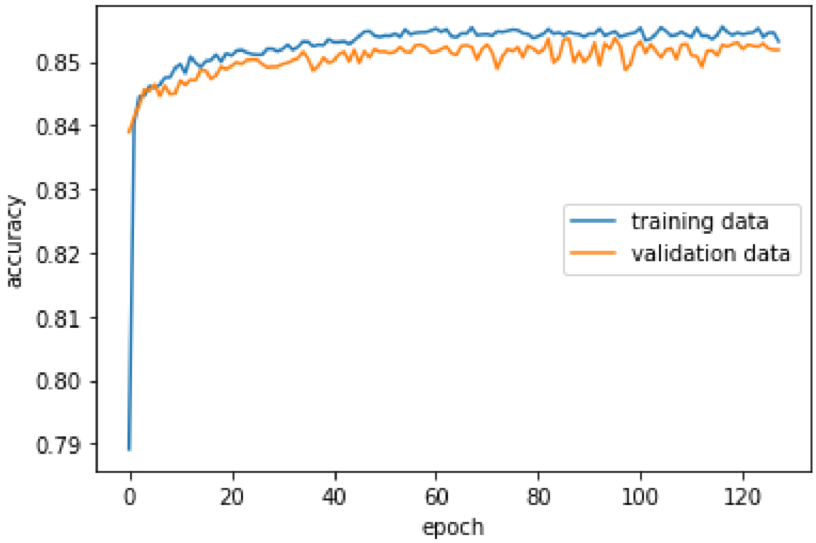
6 liner activation vs sigmoid

I tested Liner vs sigmoid both all nodes and only the last node with the rest relu.

Table 2: node testing

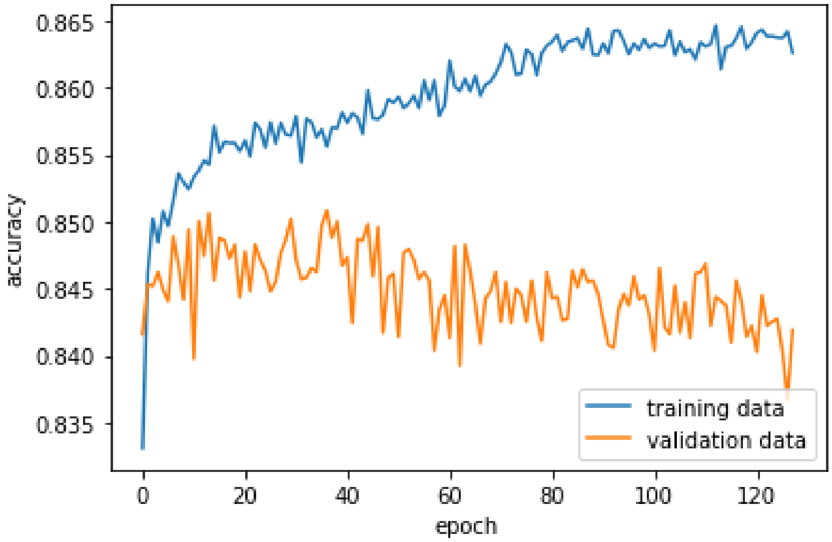
all Liner:	
accuracy:	0.84451
mean:	0.36588
std:	0.45645
Last node liner:	
accuracy:	0.84841
mean:	0.59825
std:	2.98531
Last node liner:	
accuracy:	0.84841
mean:	0.59825
std:	2.98531
last node sigmoid	
accuracy	0.85464
mean	0.21290
std	0.24846
all sigmoid:	
accuracy:	0.85061
mean:	0.21290
std:	0.24451

Based on the testing I concluded that Sigmoid needed to be the last node for my model. This makes sense because It is a classification problem. I decided to use all sigmoid nodes as it was the best preforming in my testing. The best curve I found is below



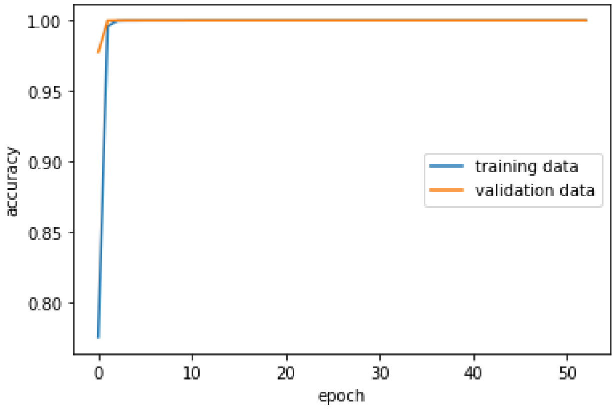
7 overfiting the data

over fitting was done by using a 34-16-8-4-1 node structure. As you can see in the curve below it was quite over fit.



8 over fitting with output as input feature

almost right away the accuracy shot to 1.00 as it should when it has the output feature as input to the system.



9 Predictions

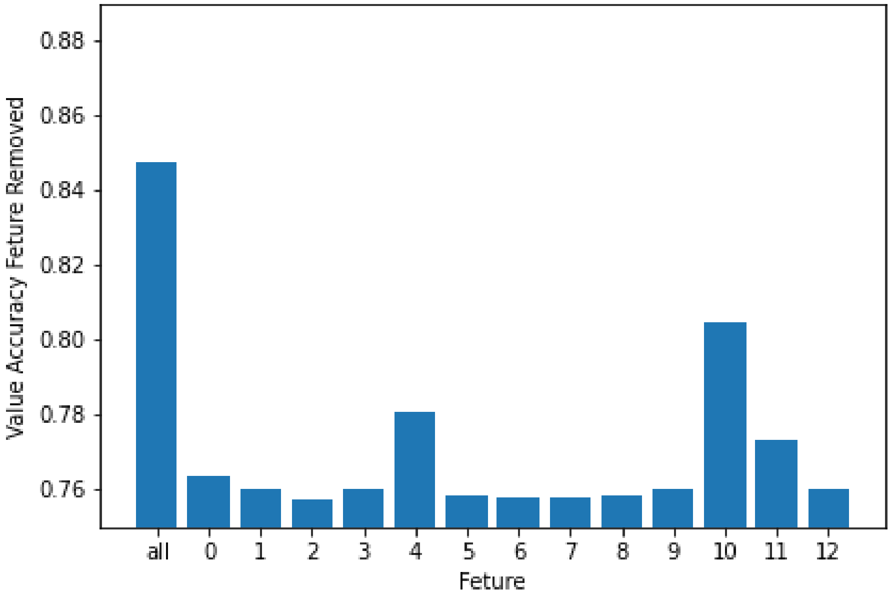
My model does an alright job of fitting to the validation set. With an accuracy of .85061 and a MAE of .2129 the neural net makes good guesses most of the time.

10 Data importance

Now that I knew how to set up my neural network to best meet my needs. I needed to investigate my data in-order to remove extraneous inputs.

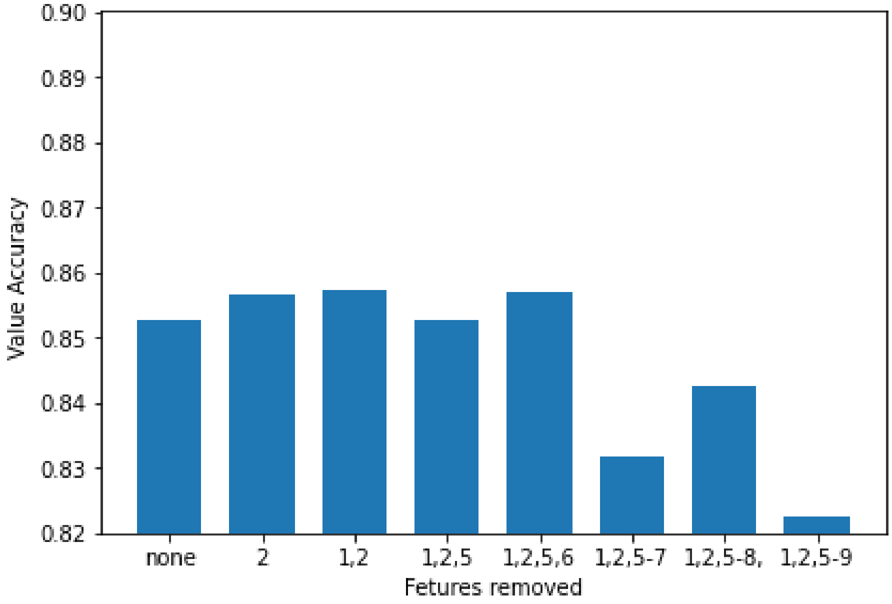
11 Study feature importance by iterativaly removing input features

I first tested my model by removing one input feature at a time, then I tested my model by only testing one input feature at a time. The resulting graph is the relative importance of each input feature.



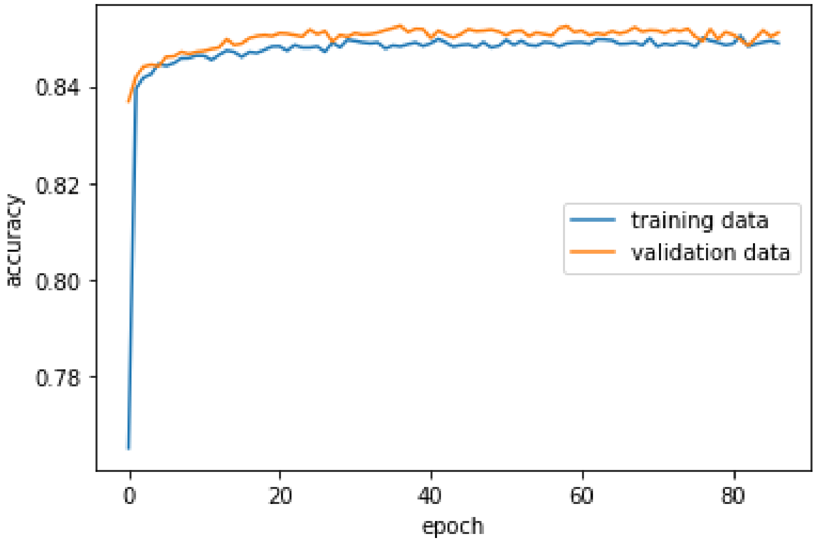
12 Identify non-informative input features and remove them

I then removed the least important features one a time resulting in the graph below.

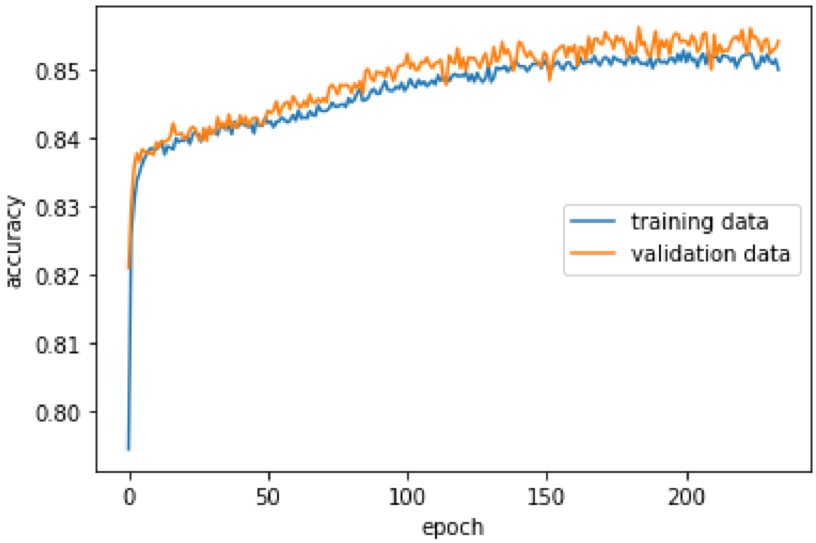


13 Compare your feature-reduced model with the original model with all input features

No features removed



After features are removed



As you might be able to see from these curves removing the variables actually made the model better at predicting.

14 Conclusion

This problem was very interesting for me to solve. There were non surprising things like capital loss/gain being so informative. There were also some surprises like hours worked not mattering as much as I would think It would have.

Neural network models are hard to fine tune and I learned that a lot of that is done by hand. It can seem more like an art than a science when you are slowly testing and fine tuning your model. The fast stopping and save features make this much more reliable and interesting. I learned a lot about how these networks work and I feel like given a new data set I could make a good attempt at creating a Neural network that would do a good job with it.One cool thing I want to do in my free time is find the data set from another year and test my network on that year. This would be especially Interesting if I make my model a function.