Adult Salary Prediction Analysis

Nirmal Sai Swaroop Janapaneedi

24/05/2020

1. Introduction

1.1 Dataset

The dataset which was used to create this analysis and the report was gathered online and can be accessed through the following link:

https://archive.ics.uci.edu/ml/machine-learning-databases/adult (https://archive.ics.uci.edu/ml/machine-learning-databases/adult)

This website offers a lot of different datasets to train machine learning capabilities and use datasets from various topics (https://archive.ics.uci.edu/ml/index.php (https://archive.ics.uci.edu/ml/index.php)). The dataset chosen for this paper is called “Adult Data Set” and contains different parameters from individuals like race, age, gender etc. It also contains the information if the person earns ***more or less than 50K*** per year.

This paper shows a model to predict, if a person earns more or less than 50K depending on different parameters from the dataset.

1.2 Insights in dataset

Some basic information about the dataset:

- Number of rows: 32,561
- Number of columns: 15
- Age range: 17 - 90 years

2. Methodology

In the first step, the dataset is going to be tidied, because it contains missing values which will not be used in the further analyses. After this step, the dataset will be simplified in the workclass column. This column holds different working classes which can be combined. For example the different levels where governmental employees work (local, state and federal), are combined to the more general workclass “Government”.

Then the analyses part starts and a closer look at the dataset is taken. Different parameters will be viewed at to see, if there are differences regarding the income for different characteristics of the parameters.

Then two models will be used to predict if the person earns more or less than 50K a year.

The first model is a regression analysis using the glm() function of R. Therefore the dataset will be manipulated and split into two parts, a test set containing of 10% of the data and a training set which holds 90% of the data.

The second model is an analysis usuing the k-nearest neighbors (knn) algorithm. Therefore the dataset will again be manipulted and split into two groups. This time, the test dataset and the training dataset will be equally large by splitting it into 50:50. This save computation time and, as several test runs showed, does not affect the results in a significant way.

3. The Code

3.1 Basic setup

First the neccessary libraries are loaded and a timer is started to measure how long the script needs to run. The results will be shown at the end of this report.

```
# Loading libraries

if(!require(tidyverse)) install.packages("tidyverse", repos = "http://cran.us.r-project.org")
if(!require(caret)) install.packages("caret", repos = "http://cran.us.r-project.org")
if(!require(ggplot2)) install.packages("ggplot2", repos = "http://cran.us.r-project.org")

library(tidyverse)
library(ggplot2)
library(caret)

# Starting timer to record how long the script runs
start <- proc.time()[3]
```

The following code loads the dataset into the R environment.

```
# Loading dataset
adult <- read.table('https://archive.ics.uci.edu/ml/machine-learning-databases/adult/adult.da
ta',
                    sep = ',', fill = F, strip.white = T)

colnames(adult) <- c('age', 'workclass', 'fnlwgt', 'education', 'education.num', 'marital.sta
tus',
                     'occupation', 'relationship', 'race', 'sex', 'capital.gain', 'capital.lo
ss',
                     'hours.per.week', 'native.country', 'income')
```

3.2 Data preprocessing

The dataset has missing values in the columns “workclass”, “occupation” and “native.country”. These missing values are shown as “?” in the dataset and are not useful for the prediction model. Therefore they will be taken out of the dataset. This affects 2,399 from the 32,561 rows, which means the dataset loses 7.36% of its entries.

```
# Cleaning dataset from rows with missing values containing "?"

adult_clean <- adult[!adult$workclass %in% c("?"),] # cleaning rows with "?" in workclass
adult_clean <- adult_clean[!adult_clean$occupation %in% c("?"),] # cleaning rows with "?" in
 occupation
adult_clean <- adult_clean[!adult_clean$native.country %in% c("?"),] # cleaning rows with "?"
in native.country
```

In the next step the dataset will be simplified. Some characteristics of the “workclass” parameter can be combined to make it easier understandable.

```
# Simplify workclasses to make the workclass easier understandable

adult_clean$workclass <- gsub('Self-emp-inc', 'Self-Employed/Freelancer', adult_clean$workcla
ss)
adult_clean$workclass <- gsub('Self-emp-not-inc', 'Self-Employed/Freelancer', adult_clean$wor
kclass)

adult_clean$workclass <- gsub('Federal-gov', 'Government', adult_clean$workclass)
adult_clean$workclass <- gsub('Local-gov', 'Government', adult_clean$workclass)
adult_clean$workclass <- gsub('State-gov', 'Government', adult_clean$workclass)
```

With the dataset prepared, the analysis can be started.

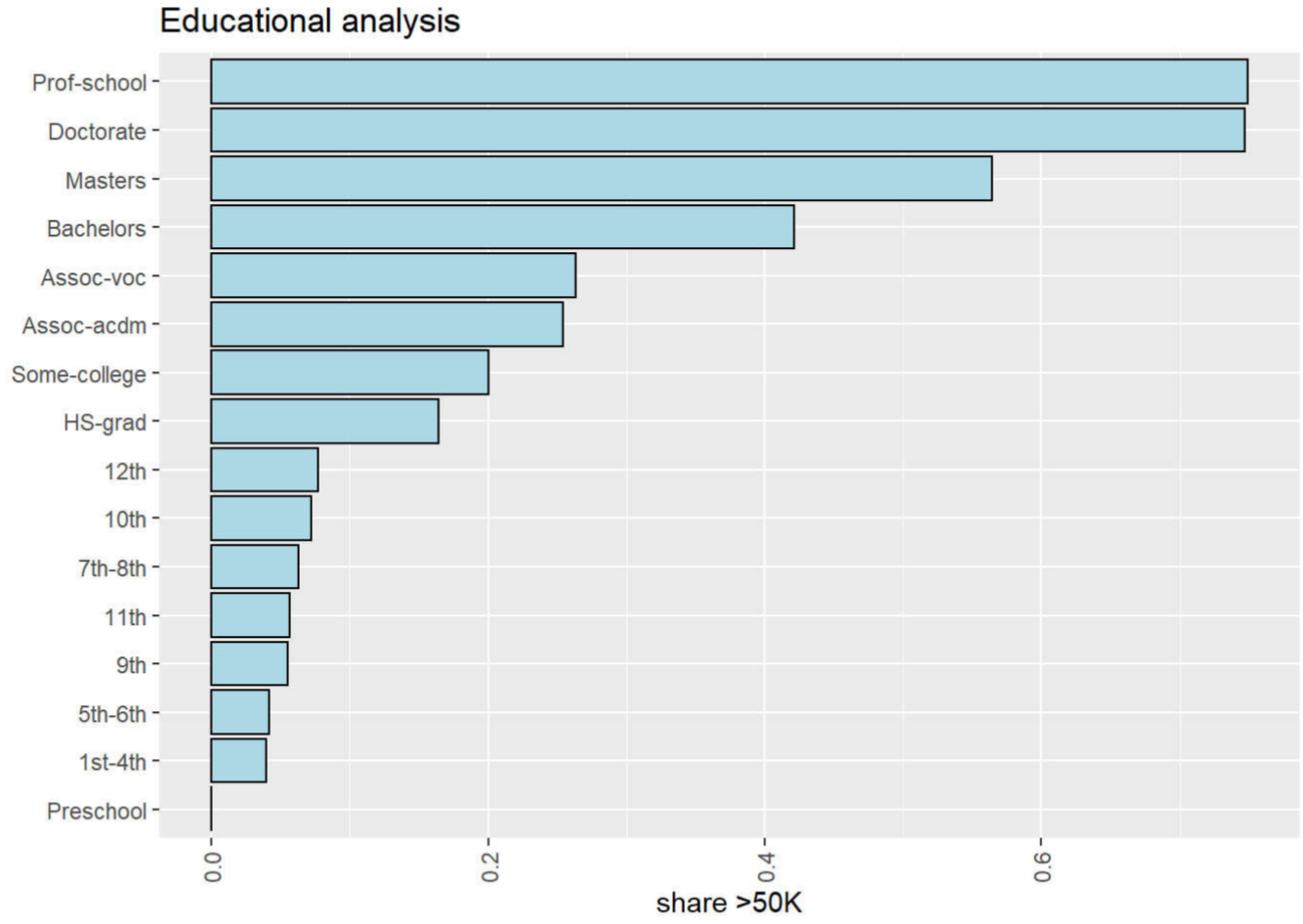
3.3 Data analysis and visualization

3.3.1 Educational analysis

The first parameter which will be analyzed more closely is the education. There is probably a high correlation between a higher education and a higher income.

```
#Education analysis
adult_education <- adult_clean %>% group_by(education) %>% summarize(share = mean(income == ">50K")) %>% arrange(desc(share))

ggplot(data=adult_education, aes(x=reorder(education, +share), y=share)) +
  geom_bar(stat="identity", color = "black", fill = "lightblue") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5)) +
  xlab(NULL) +
  ylab("share >50K") +
  ggtitle("Educational analysis") +
  coord_flip()
```



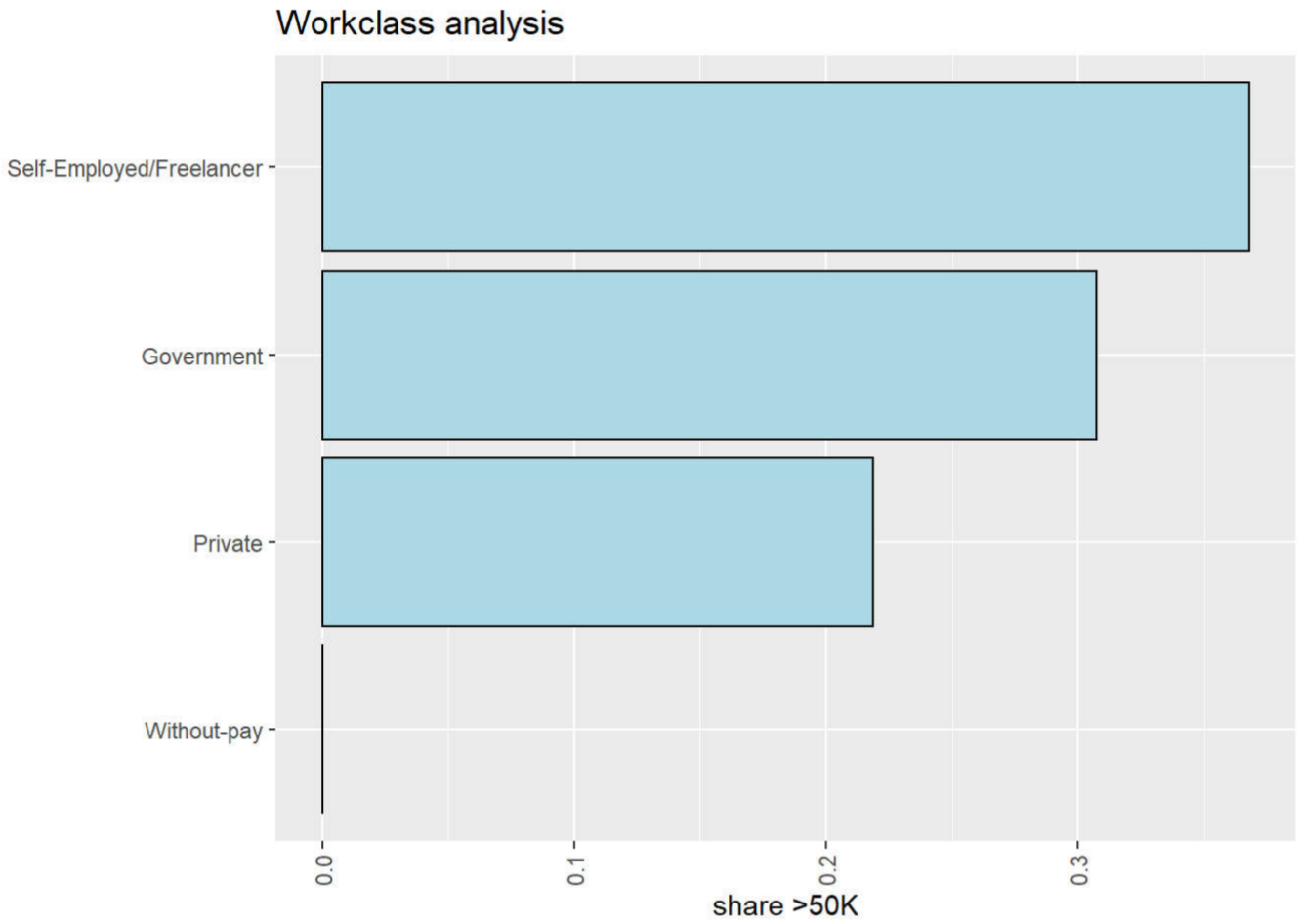
The graph confirms the initial proposition and clearly shows a strong connection between education and income.

3.3.2 Workclass analysis

In the next step the (updated) workclass will be analyzed to see if there is a connection between the workclass and the income as well.

```
# Workclass analysis
adult_workclass <- adult_clean %>% group_by(workclass) %>% summarize(share = mean(income == ">50K")) %>% arrange(desc(share))

ggplot(data=adult_workclass, aes(x=reorder(workclass, +share), y=share)) +
  geom_bar(stat="identity", color = "black", fill = "lightblue") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5)) +
  xlab(NULL) +
  ylab("share >50K") +
  ggtitle("Workclass analysis") +
  coord_flip()
```



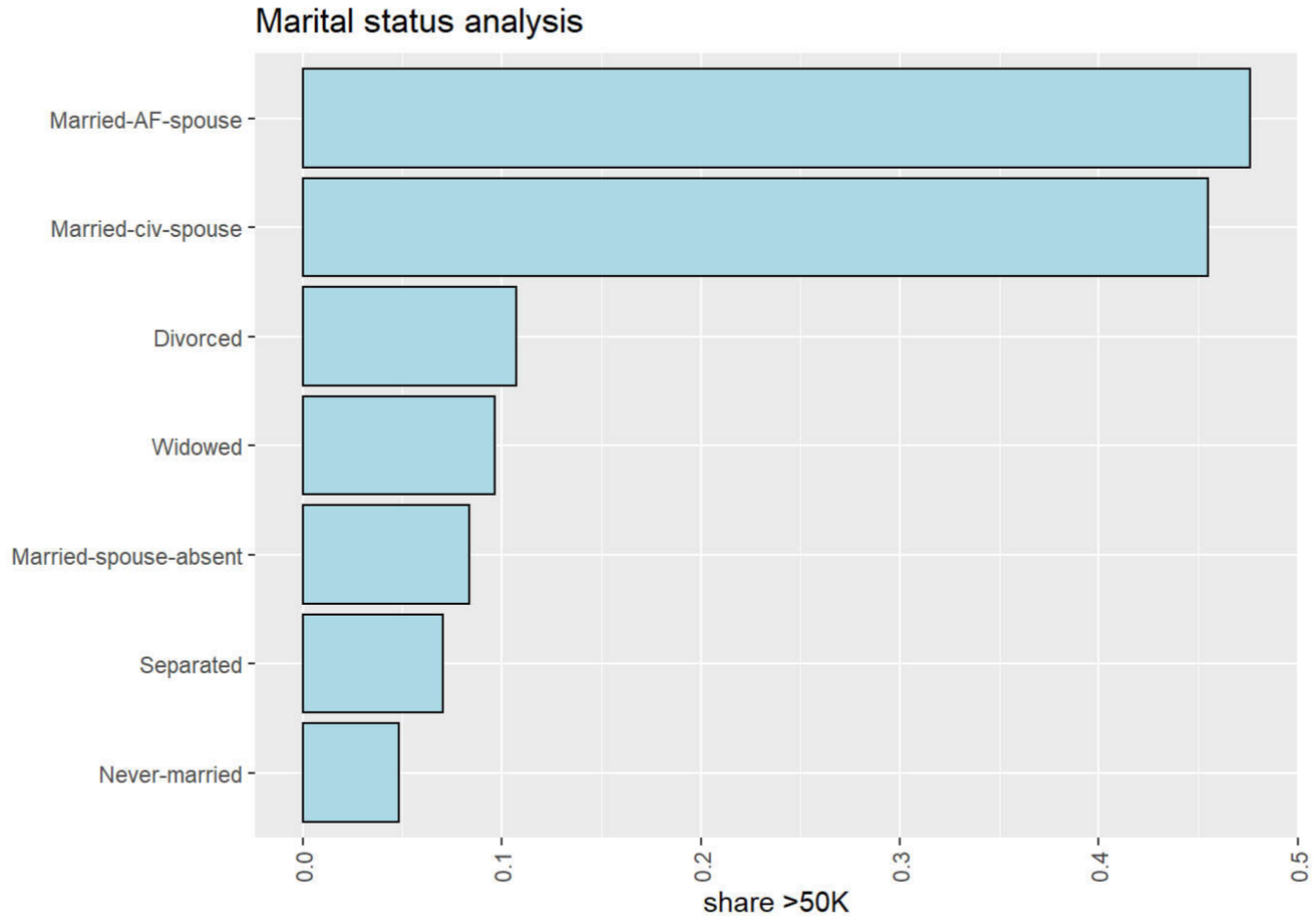
As seen in the graph, there is a solid connection between the workclass of an individual and its income.

3.3.3 Marital status analysis

Now the marital status will be analyzed to see if there is a connection as well.

```
# marital status analysis
adult_marital.status <- adult_clean %>% group_by(marital.status) %>% summarize(share = mean(i
ncome == ">50K")) %>% arrange(desc(share))

ggplot(data=adult_marital.status, aes(x=reorder(marital.status, +share), y=share)) +
  geom_bar(stat="identity", color = "black", fill = "lightblue") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5)) +
  xlab(NULL) +
  ylab("share >50K") +
  ggtitle("Marital status analysis") +
  coord_flip()
```



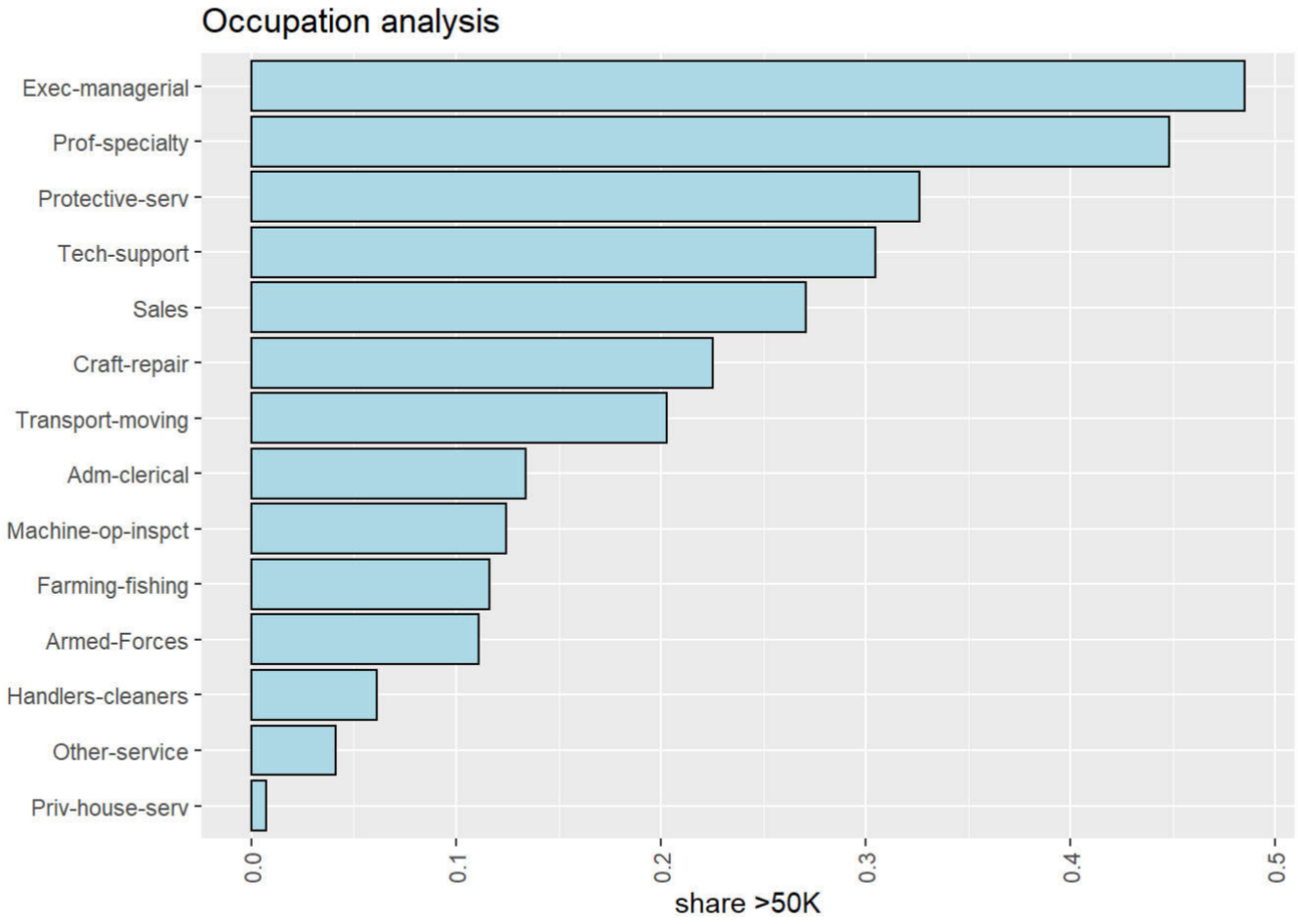
The graph shows that married people, in general, have a higher income.

3.3.4 Occupation analysis

Here the occupation of an individual will be analysed to find possible connections to the income.

```
# occupation analysis
adult_occupation <- adult_clean %>% group_by(occupation) %>% summarize(share = mean(income ==
">50K")) %>% arrange(desc(share))

ggplot(data=adult_occupation, aes(x=reorder(occupation, +share), y=share)) +
  geom_bar(stat="identity", color = "black", fill = "lightblue") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5)) +
  xlab(NULL) +
  ylab("share >50K") +
  ggtitle("Occupation analysis") +
  coord_flip()
```



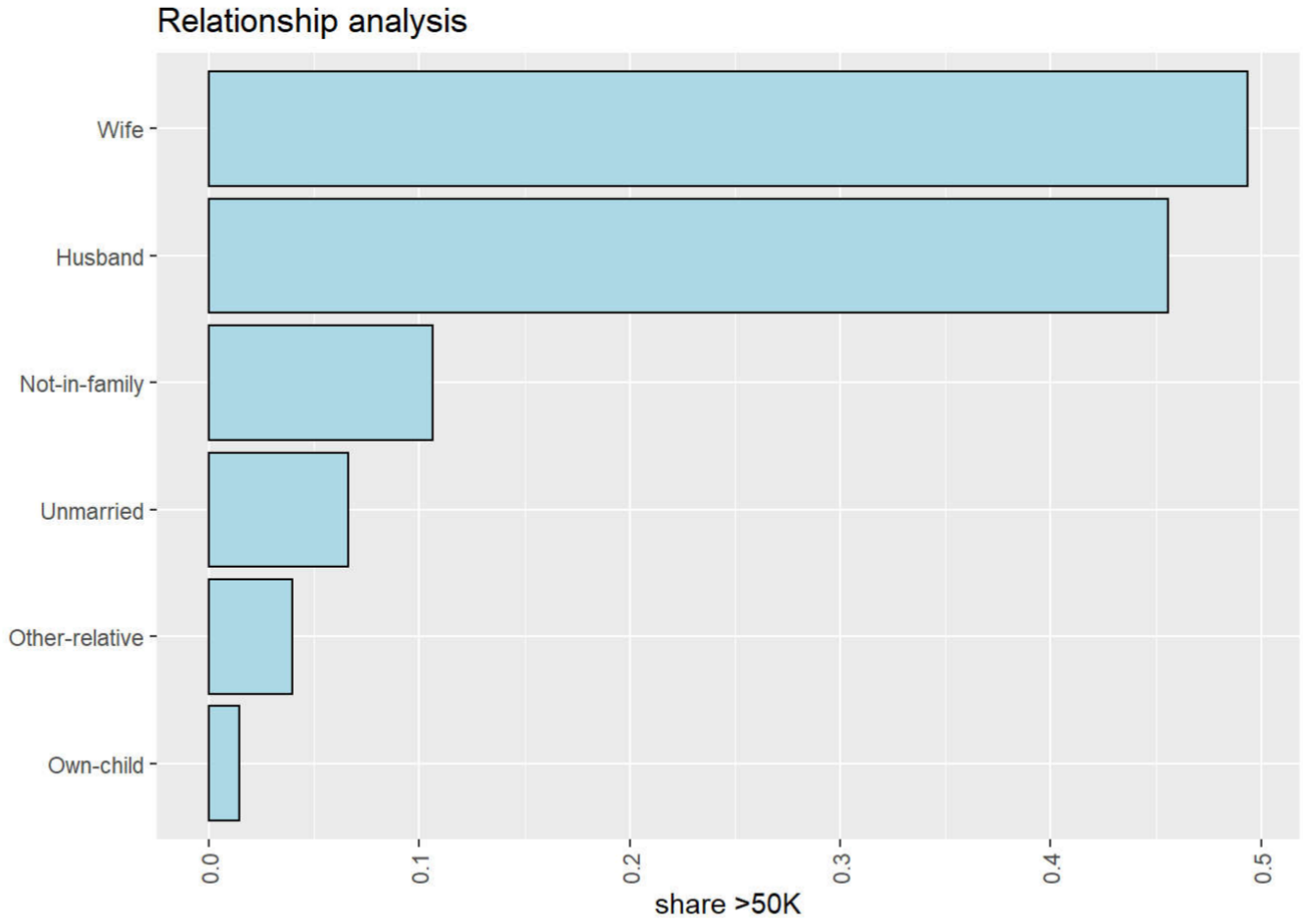
The occupation of a person has a high influence on the income, therefore this seems to be an important parameter for our later predictions.

3.3.5 Relationship analysis

In this analysis a closer look at the relationship status of an individual is taken and will be displayed in a graph.

```
# relationship analysis
adult_relationship <- adult_clean %>% group_by(relationship) %>% summarize(share = mean(incom
e == ">50K")) %>% arrange(desc(share))

ggplot(data=adult_relationship, aes(x=reorder(relationship, +share), y=share)) +
  geom_bar(stat="identity", color = "black", fill = "lightblue") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5)) +
  xlab(NULL) +
  ylab("share >50K") +
  ggtitle("Relationship analysis") +
  coord_flip()
```



In the graph it becomes obvious that the relationship also plays a role for the income.

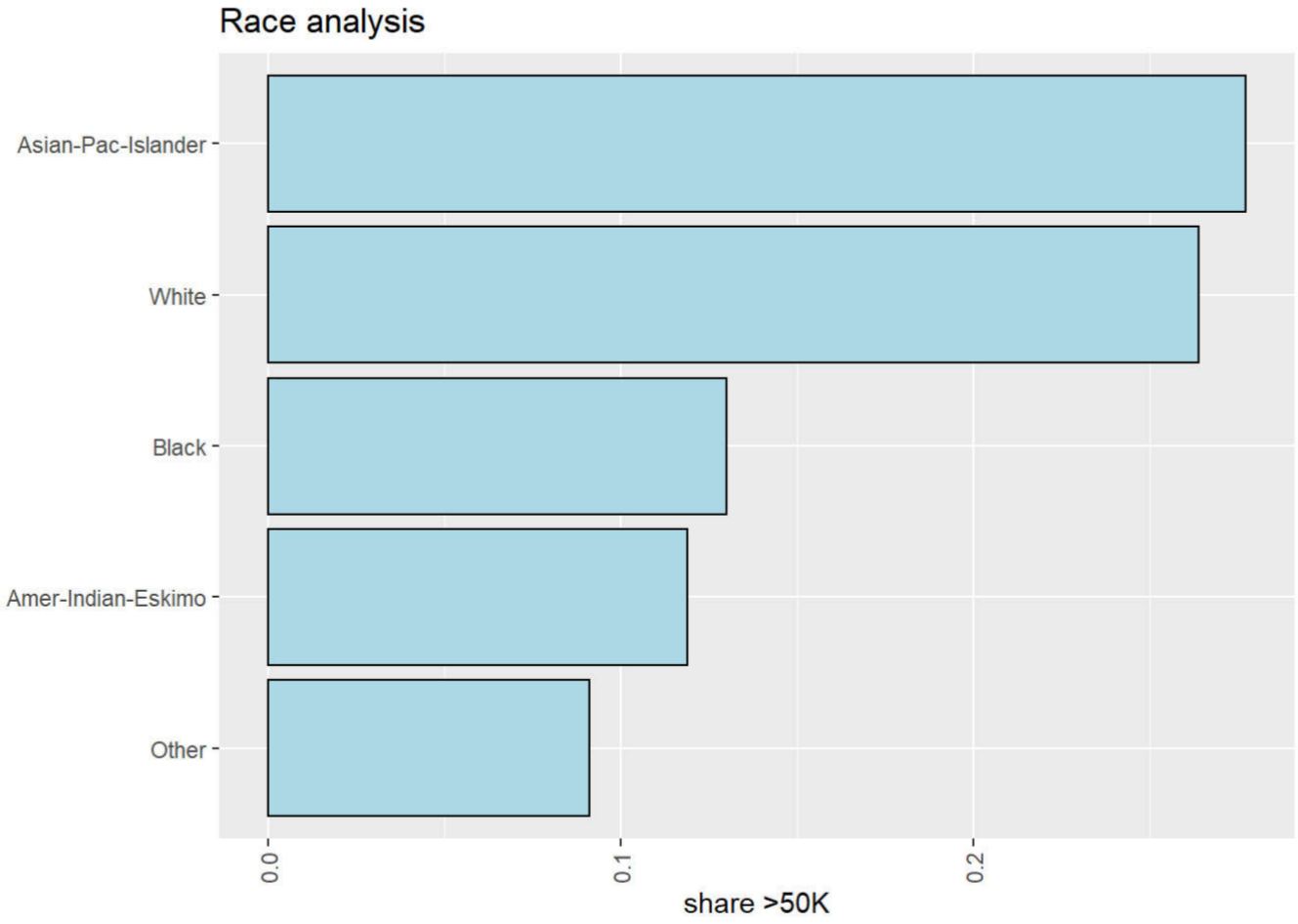
But we will not put to much focus on this parameter, since it is already included in the mariatial status and through that it will be included in the prediction. Therefore the results are not surprising and also not that relevant.

3.3.6 Race analysis

Now the race of a person will be analyzed to see if there are any connections as well.

```
# race analysis
adult_race <- adult_clean %>% group_by(race) %>% summarize(share = mean(income == ">50K")) %
>% arrange(desc(share))

ggplot(data=adult_race, aes(x=reorder(race, +share), y=share)) +
  geom_bar(stat="identity", color = "black", fill = "lightblue") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5)) +
  xlab(NULL) +
  ylab("share >50K") +
  ggtitle("Race analysis") +
  coord_flip()
```



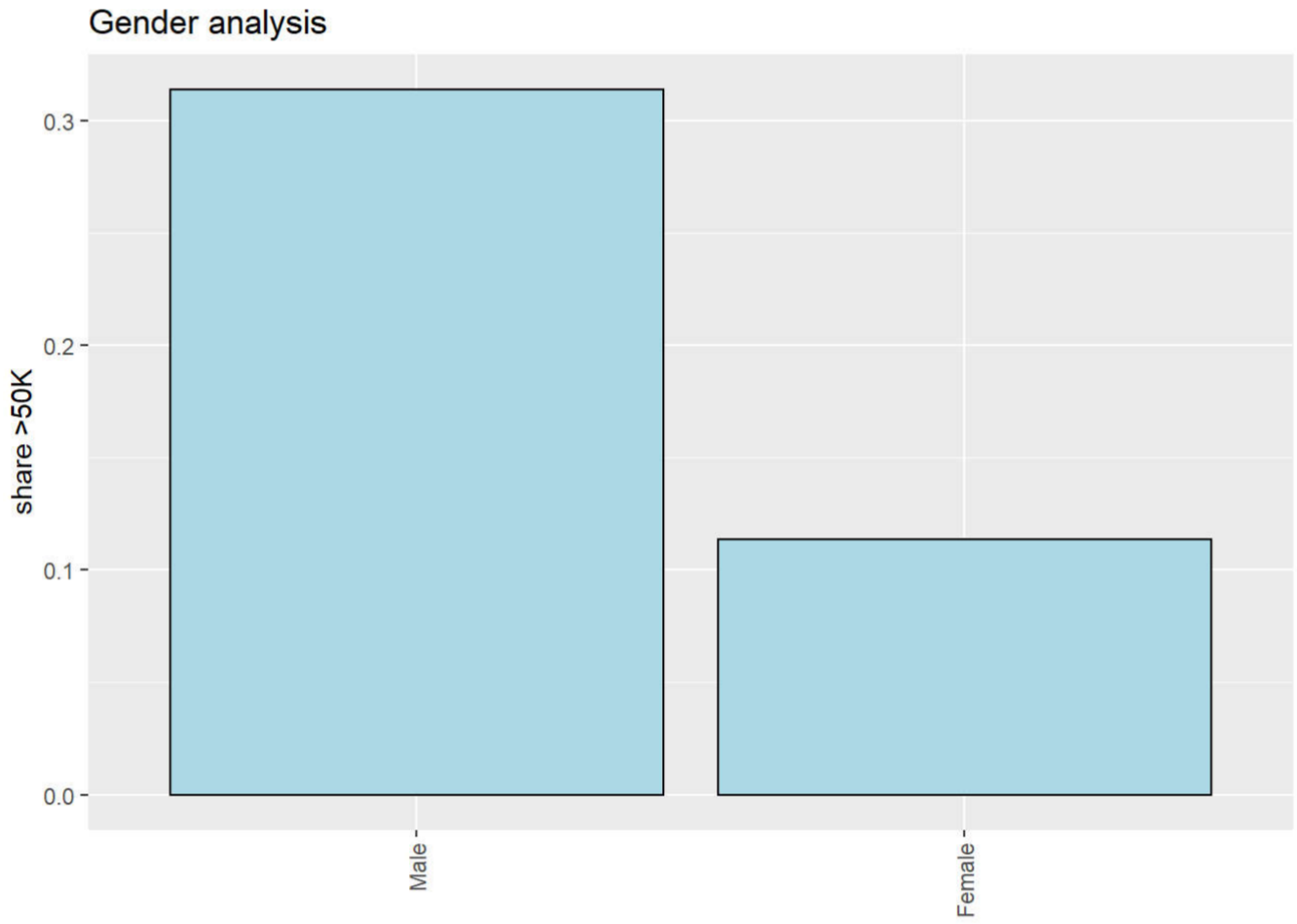
The race seems to play a role as well, but is not that significant like the maritial status for example.

3.3.7 Gender analysis

For the gender analysis two graphs will be displayed to show the income pclasses and also the count of individuals in the underlying dataset.

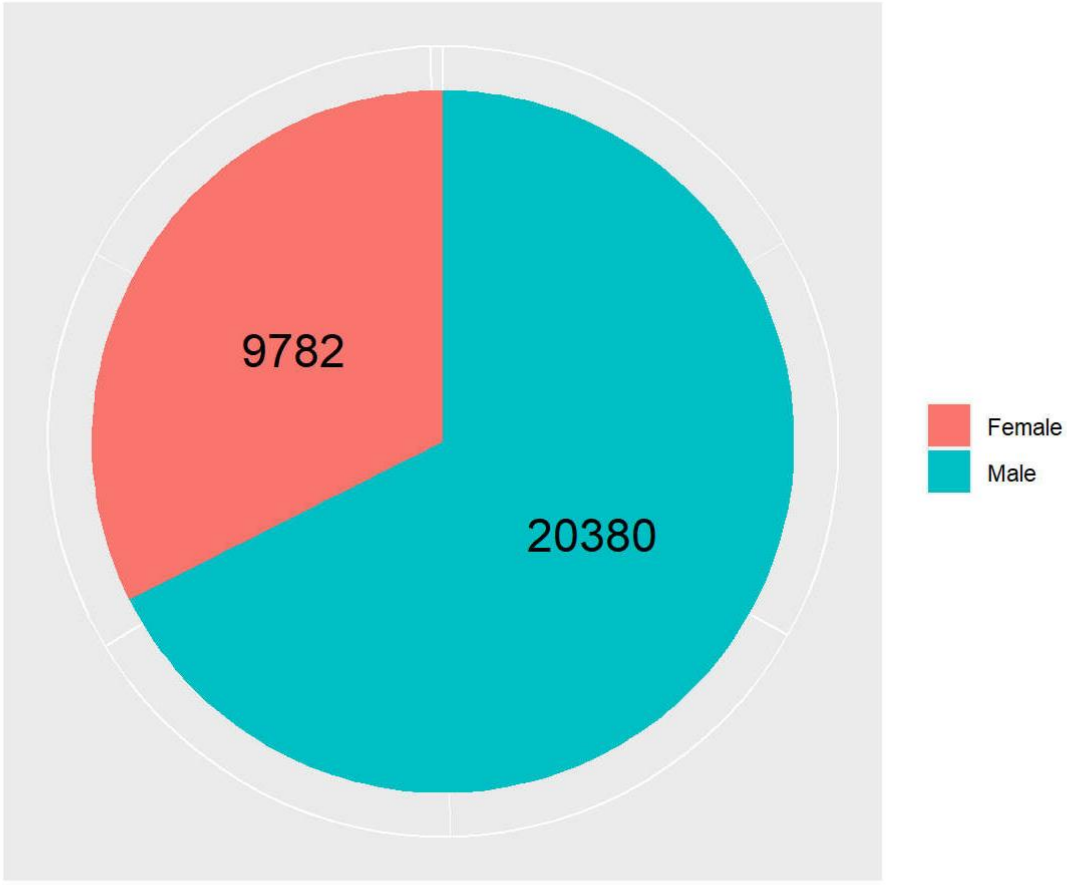
```
# Gender analysis
adult_sex <- adult_clean %>% group_by(sex) %>% summarize(share = mean(income == ">50K"), n =
 n()) %>% arrange(desc(share))

ggplot(data=adult_sex, aes(x=reorder(sex, -share), y=share)) +
  geom_bar(stat="identity", color = "black", fill = "lightblue") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5)) +
  xlab(NULL) +
  ylab("share >50K") +
  ggtitle("Gender analysis")
```




```
ggplot(data=adult_sex, aes(x="", y=n, fill=sex)) +
  geom_bar(width = 1, stat = "identity") +
  coord_polar("y") +
  geom_text(aes(label = n), position = position_stack(vjust = 0.5), check_overlap = T, size =
6) +
  labs(x = NULL, y = NULL, fill = NULL, title = "Gender analysis income >50K") + theme(axis.l
ine = element_blank(), axis.text = element_blank(), axis.ticks = element_blank(), )
```

Gender analysis income >50K



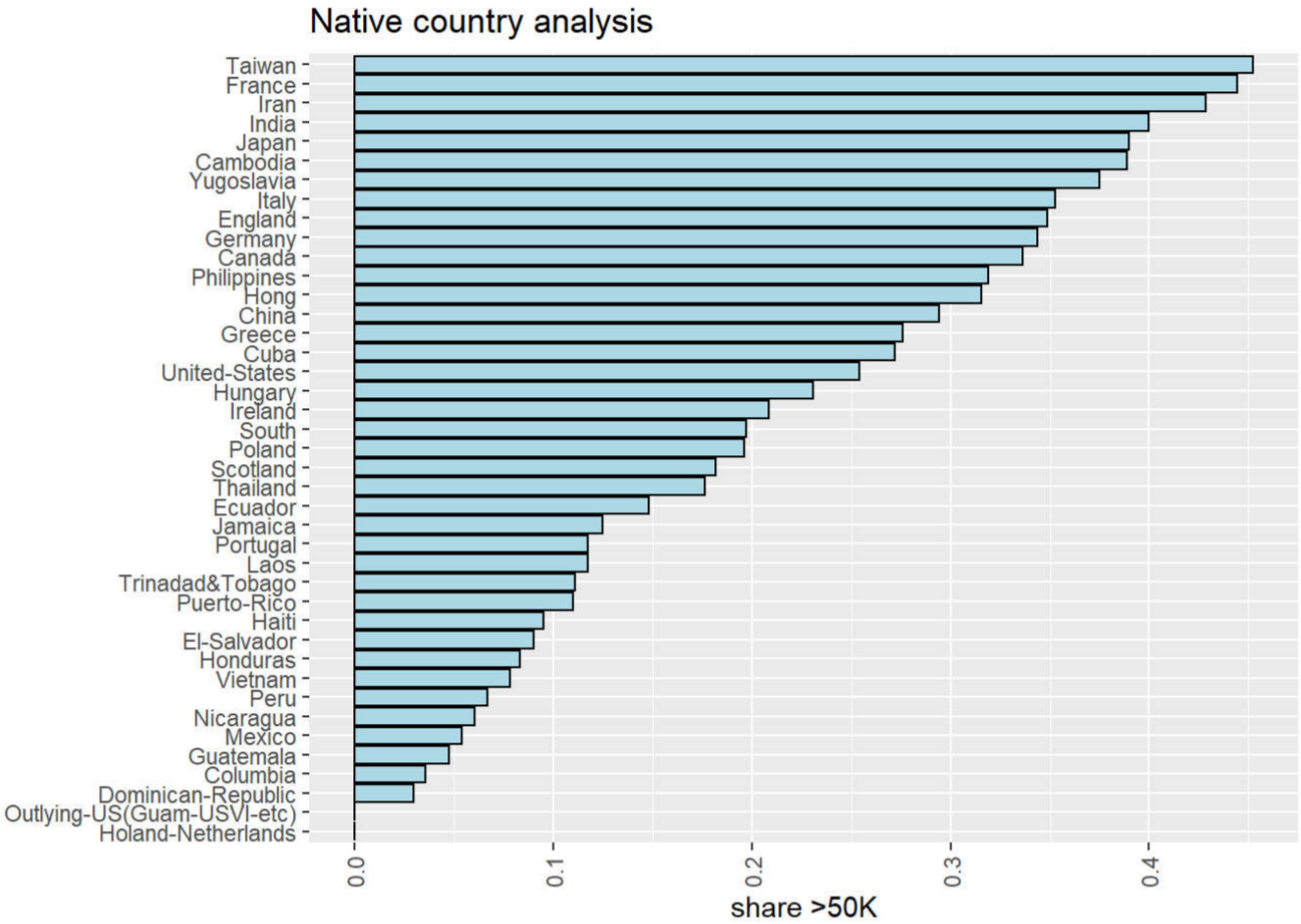
The graphs show that more men are in the dataset than women and also men seem to earn more than women.

3.3.8 Native country analysis

Another, probably important factor, is the native country of an individual. Countries with higher educational standards could be indicators for higher income.

```
# native country analysis
adult_native.country <- adult_clean %>% group_by(native.country) %>% summarize(share = mean(i
ncome == ">50K")) %>% arrange(desc(share))

ggplot(data=adult_native.country, aes(x=reorder(native.country, +share), y=share)) +
  geom_bar(stat="identity", color = "black", fill = "lightblue") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5)) +
  xlab(NULL) +
  ylab("share >50K") +
  ggtitle("Native country analysis") +
  coord_flip()
```

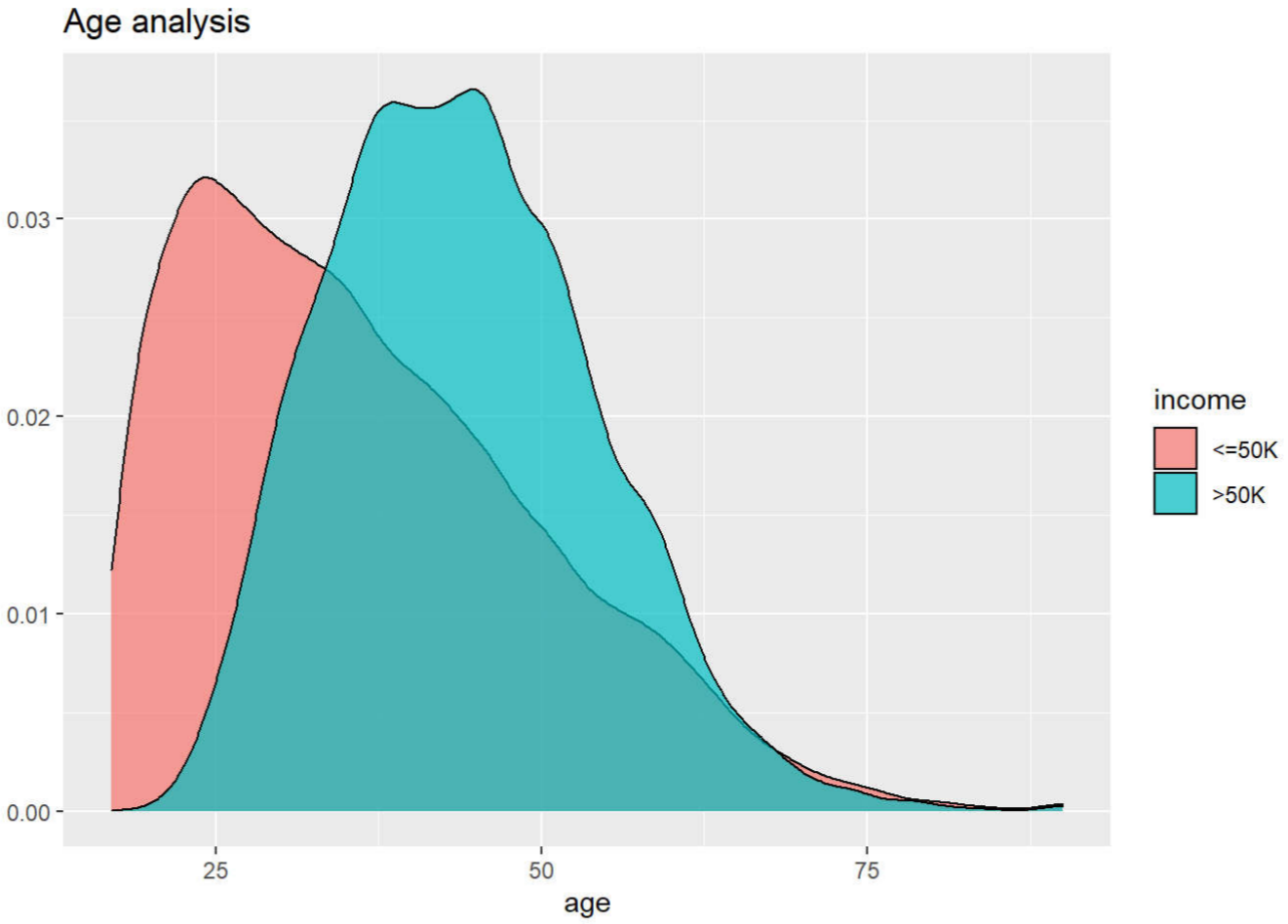


As seen in the graph the native country plays a role, but a general conclusion cannot be made. The top group consists of a mix of different continents, countries and cultures.

3.3.9 Age analysis

The last analysis inspects the age of an individual. Therefore the graph shows the two groups and their age distribution.

```
# age analysis
ggplot(adult_clean, aes(x = age, fill = income)) +
  geom_density(alpha = 0.7) +
  ylab(NULL) +
  ggtitle("Age analysis")
```



The result shows a strong connection between age and income. With increasing age the income also increases, so this factor should also be included in the further analysis.

3.4 Modelling

To predict, if someone belongs to one group or the other two models are used.

First a linear regression model is used to predict the income and in the second step the knn algorithm is used.

3.4.1 Linear regression

Before the model is built, a binary new column called “high_income” is added to the dataset. In the next step, the unwanted columns are removed from the dataset.

```
# Adding the indicator for high income
adult_clean$high_income <- ifelse(adult_clean$income == ">50K", 1, 0)

# removing not needed columns
adult_clean_lin <- adult_clean %>%
  select(-fnlwgt, -capital.gain, -capital.loss, -education.num, -income)
```

Now the dataset will be split into two groups, a training set and a test set. The test set consists of 10% of the data to test the algorithm, while the training set consists of 90% of the data and is used to train the dataset.

```
# splitting the dataset

test_index <- createDataPartition(adult_clean_lin$high_income, times = 1, p = 0.1, list = FAL
SE)
train_set <- adult_clean_lin %>% slice(-test_index)
test_set <- adult_clean_lin %>% slice(test_index)
```

Finally the model is built using the glm() function and the binary column “high_income”. The cutoff for the prediction is 0.5.

```
# Predicting to which group an individual belongs

reg_model <- glm(high_income ~ ., data = train_set, family=binomial)
reg_pred <- predict(reg_model, test_set, type="response")
# the cutoff for predicted probability lies by 0.5
reg_pred1 <- ifelse(reg_pred <= 0.5, 0, 1)
result_reg <- sum(reg_pred1==test_set$high_income)/length(reg_pred1)
```

This leads to the following results. A table which shows the matches and mismatches and the final hit rate.

```
# Shows the result as percentage
result_reg
```

```
## [1] 0.8279748
```

```
# Returns a table, to see how many matches and mismatches the algorithm produced
table(reg_pred1, test_set$high_income)
```

```
##
## reg_pred1    0    1
##         0 2065  319
##         1  200  433
```

```
print(paste("The regression model predicted the income group in ", round(result_reg*100,digit
s=2), "% of the calculations correctly"))
```

```
## [1] "The regression model predicted the income group in  82.8 % of the calculations correc
tly"
```

3.4.2 KNN model

The knn algorithm takes was longer and therefore, the dataset will be manipulated to shorten the time needed for the calculation. Side-note: On a more powerful machine the algorithm runs faster and more parameters can be used. But testing showed, the hit rate does not improve significantly enough to justify the long computation time for this dataset.

```
# Because it takes a long time, I want to use fewer prarmeters

adult_clean_knn <- adult_clean %>%
  select(-fnlwgt, -capital.gain, -capital.loss, -education.num, -high_income, -relationship,
 -race, -sex, -hours.per.week, -native.country)
```

The next step is to split the dataset. This time the dataset is split into 50:50. Again this happens to shorten the calculation time. Test runs with a bigger dataset for training showed a significantly longer calculation time with nearly no improvement in the hit rate. Therefore the dataset is chosen to be 50:50 to shorten the time in case any of you wants to run the code.

```
# The dataset is split 50:50 to save computation time. Several test runs showed, that the res
ults does not change significantly, if the training set is bigger, but the computation time i
s rising rapidely. Therefore I chose the spilt 50:50 in case you want to run the code yoursel
f.
test_index_knn <- createDataPartition(adult_clean_knn$income, times = 1, p = 0.5, list = FALS
E)
train_set_knn <- adult_clean_knn %>% slice(-test_index_knn)
test_set_knn <- adult_clean_knn %>% slice(test_index_knn)
```

Now the model is created, trained and tested on the test dataset.

```
# Predicting to which group an individual belongs

knn_model <- train(income ~ .,
                   data = train_set_knn,
                   method = "knn")
print(knn_model)
```

```
## k-Nearest Neighbors
##
## 15081 samples
##     5 predictor
##     2 classes: '<=50K', '>50K'
##
## No pre-processing
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 15081, 15081, 15081, 15081, 15081, 15081, ...
## Resampling results across tuning parameters:
##
##   k  Accuracy   Kappa
##   5  0.7912923  0.4265037
##   7  0.7976143  0.4371280
##   9  0.8015475  0.4433571
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was k = 9.
```

```
predictions_knn <- predict(knn_model, test_set_knn[,1:5])
result_knn <- sum(predictions_knn == test_set_knn[,6])/length(test_set_knn[,6])
```

The results of the knn model are as follows:

```
print(paste("The knn model predicted the income group in ", round(result_knn*100,digits=2),
"% of the calculations correctly"))
```

```
## [1] "The knn model predicted the income group in  81.58 % of the calculations correctly"
```

4. Conclusion

After both models have been run, the final results are summarized:

```
## [1] "The regression model predicted the income group in  82.8 % of the calculations correc
tly"
```

```
## [1] "The knn model predicted the income group in  81.58 % of the calculations correctly"
```

```
## [1] "This script ran for 499.32 seconds"
```

```
## [1] "The machine used runs Win 10 x64 on a Xenon E3-1231 v3 @ 3.40 GHz with 16 GB RAM."
```

The models show, that there is a difference in the hit rate of the two models but it is not very high. Still, the hit rate of ~80% is good and probably works for most applications. In sensitive fields, for example financial scenarios, in my personal opinion the hit rate should be higher and be at least >85%.

The limitations of this report lies on one side in the parameters used. More parameters would probably improve the model performance, but the limited power of the used machine makes the computation time pretty long. Therefore a better environment would speed up the process and allow more parameters for the predictions in a reasonable amount of time.

Another approach to continue working with the dataset is to use other algorithms and test their prediction hit rates.

Also decision trees and random forests could be used to predict to which group an individual belongs.