Wage Analysis using Additive Models

Ann Joseph, Jihye Park, Meher Pooja Pranavi Punyamanthula, Ross Graham

Introduction

An additive model is a nonparametric regression method. Since the response variable in this analysis is a continuous variable, we will be using generalized additive models (GAM). Generalized additive models were originally invented by Trevor Hastie and Robert Tibshirani in 1986. It is a powerful yet, simple technique where the relationship between the predictors and the response variable follow smooth patterns that can be linear or nonlinear. In this additive modeling technique, the impact of individual predictors is determined through smooth functions which can be nonlinear. We can write the GAM structure as:

$$y = \beta_0 + {}^{p}\sum_{j=1} f_j(X_j) + \varepsilon$$

where the f_j are smooth arbitrary functions on each predictor variable, X_j.

The advantages of using a generalized linear model are:

- It is easy to interpret.
- It uses flexible predictor functions that can uncover hidden trends in the data.
- It avoids overfitting by regularizing predictors

GAM has the interpretability advantages of generalized linear models (GLM) with more flexibility because it allows a non-linear relationship between the predictors and the response variable. Because of its regularization, there is no overfitting which sometimes happens when usihng higher order polynomial terms in GLMs.

Description of the Data Set

We used the Wage dataset in the ISLR library in R for this analysis. It has 3000 rows and 11 columns describing wage and other information of a group of 3000 male workers in the Mid-Atlantic region. The response variable in this analysis is ‘logwage,’ which is a numerical variable indicating the log of a workers wage. The other 10 variables in this data set are as follows:

1. year: Year that wage information was recorded
2. age: Age of worker
3. maritl: A factor with levels 1. Never Married 2. Married 3. Widowed 4. Divorced and 5. Separated indicating marital status
4. race: A factor with levels 1. White 2. Black 3. Asian and 4. Other indicating race
5. education: A factor with levels 1. < HS Grad 2. HS Grad 3. Some College 4. College Grad and 5. Advanced Degree indicating education level

6. region: Region of the country (mid-atlantic only)
7. jobclass: A factor with levels 1. Industrial and 2. Information indicating type of job
8. health: A factor with levels 1. <=Good and 2. >=Very Good indicating health level of worker
9. health_ins: A factor with levels 1. Yes and 2. No indicating whether worker has health insurance
10. wage: Workers raw wage

Out of the above, ‘wage’ and ‘region’ were removed and the other 8 variables are used as the predictors in this analysis. The variable ‘wage’ was removed because the response variable is directly obtained from it and the variable ‘region’ was removed since there was only one region in this dataset.

Data cleaning and data pre-processing

In order to obtain high data quality we checked if the data possess the following 7 dimensions:

1. Completeness: Completeness of the data is a dimension that ensures there are no missing values in the dataset so that we do not end up drawing inaccurate inferences about the data.
2. Uniqueness: Data needs to be unique, i.e, it should not contain any duplicate values.
3. Conformity: Conformity is essential to avoid inaccurate data which gives no meaning or wrong sense.
4. Timeliness: Timeliness can be measured as the time between when data is expected and when it is readily available for use.
5. Validity: Data validity is essential for correctness and reasonableness of data.
6. Consistency: Data consistency is a crucial aspect among data dimensions as data needs to be understandable and we should be careful to avoid misinterpretation of its meaning.
7. Accuracy: Cross checking if all the variables are collected properly without multiple representation of a single variable will ensure that the data is accurate.

In our data, there were no missing values or meaningless values such as negative or zero values for wage. Further, there were three duplicates in the dataset which were removed. All variables followed standard notations but we are not sure of accuracy of the data as we did not collect the data, only retrieved it from the source, the ISLR package in R.

Exploratory Data Analysis

From Figure 1. in the Appendix, we observe that the education variable has a strong relationship with the variable, wage, whose log is the response variable. In our data set, there are more number of workers in the industrial job class than the information job class and from Figure 2. in the Appendix, we see that the industrial job class has higher wages when compared to the information job class. This can be an interesting factor to look at it to gain meaningful insights in the future. It can also be seen from this graph that most workers have a wage less than 200 but there is a small subset of workers whose wage is above 250.

We were curious about the relationship between education and wage and on looking at Figure 3. in the Appendix, we see that the education level 'Advanced Degree' falls in the age range of 30 to 50 but also that this group has some of the highest wages. This is intuitive since getting more education means you likely start working full time at a later age but also start at higher wages.

Further, we divided wages into three groups as shown in Table 1. to see which ages fall in which wage group.

cutWage		
[20.1, 92.2)	[92.2,118.9)	[118.9,318.3]
1000	1040	960

Table 1. Three wage groups

From Figure 4. in the Appendix, we can note that workers under 30 usually earn less and that only a handful of workers between the age of 39 and 51 earn above 250. This is further explained in Table 2. where we can see that for the higher wages, there are more information job classes while for the lower wages the majority of jobs is industrial job classes. From this, it can be concluded that if the 'job class' is 'information,' the wage is usually more than if it is 'industrial.'

cutWage	1. Industrial	2. Information
[20.1, 92.2)	0.6290000	0.3710000
[92.2,118.9)	0.5125000	0.4875000
[118.9,318.3]	0.3979167	0.6020833

Table 2. Wage of the two job classes

Data Analytics using GAM

After splitting the dataset into training and test sets in the ratio 7:3, we built one linear model as a reference and five GAM models with different smoothing functions and

some additive models. We then used R^2, AIC and RMSE as performance metrics to compare models and finally choose the best one.

Linear model:
The first model we built was a linear model with stepwise model selection by using the following lines of code:

```
olm = lm(logwage~., data = train)
olm = stepAIC(olm, trace = FALSE)
```

This step function removed 'jobclass' and 'race' and built a linear model with only 6 predictors. The adjusted R^2 for this model is around 38% which means that it does not fit the data very well, suggesting significant non-linearity in the relationship between the response variable, 'logwage', and the predictors. The AIC of this model was 585.03.

Model 1 (all predictors, all continuous smoothing, no factor smoothing):
The first GAM model we built included all 8 predictors with smoothing functions on 'age' and 'year' by using the following lines of code:

```
model1 <- gam(logwage ~ s(year) + s(age)+ maritl + race
        + education + jobclass + health + health_ins,data = train)
```

Fitting this model, we see a reduction in AIC from the linear model's 585 to 531. This is a significant improvement. But, the RMSE is 44.22 and the R^2 is 36%.

Next, we examined the transformations used in this model. Since we only applied smoothing to 'year' and 'age,' we examine those graphs more closely which are shown by Figure 5. and Figure 6. in the Appendix. It can be seen that while the 'age' transformation is clearly nonlinear in Figure 5. and therefore warranted, the 'year' transformation in Figure 6. is almost linear. This inspires us to build the next model.

Model 2 (all predictors, age smoothing, no factor smoothing):
Inspired by the almost linear relationship in the 'year' shown by Figure 6. in the Appendix, next, we tried a version Model 1 with smoothing on 'year' removed as shown below:

```
model2 <- gam(logwage ~ year + s(age) + maritl + race
          + education + jobclass + health
          + health_ins, data = train)
```

This model has an adjusted R^2 of around 40% and an AIC of 528.73. On predicting using the test set, we found that the RMSE is 0.281.

We then compared this model to Model 1 to test the significance of this change. The very low p-value for ‘year’ in this model indicates that we are not confident that ‘year’ requires smoothing. However, the p-value is not so large that smoothing year would be out of the question. This aligns with our intuition that wages should just increase steadily over time but there will be periods of slightly more rapid or slightly slower increases.

It should be noted that performing the same test with a model that excludes ‘age’ smoothing yields a very small p-value. This indicates high confidence that ‘age’ requires smoothing, just as we previously concluded using the graphs of Model 1.

Model 3 (all predictors, age smoothing, factor smoothing):
Next, we build on Model 2 by allowing different smoothing functions for each age level as shown below:

```
model3 <- gam(logwage ~ year + s(age, by =maritl) + maritl
          + s(age, by = race) + race + s(age, by = education)
          + education + s(age, by = jobclass) + jobclass
          +  s(age, by = health) + health
          + s(age, by = health_ins) + health_ins, data = train)
```

This is done by using the ‘by=’ parameter in the s() function of the mgcv package. The result is a much more complicated model but one that, again, reduces AIC to 484.21 from Model 2's AIC of 528.73. Further, this model has an R-squared of 42% and an RMSE of 0.284.This model represents a significant improvement which comes from the increased nuance in separating the factor effects.

On plotting the transformation of education as shown in Figure 7-11. in the Appendix, it becomes clear that the ‘education’ variable has highly non-linear and highly different transformations. The transformations shown in Figure 7. and Figure 8. for education ‘less than high school’ and ‘high school’ respectively, are almost linear. But, increasing education levels as shown in Figure 9., Figure 10. And Figure 11. have very different transformation functions. What likely drives these differences is that getting more education means you likely start working full time at a later age but also start at higher wages. These differences likely drive the improvement in the AIC and R-squared of this model.

Model 4 (fewer predictors, age smoothing, factor smoothing):
This model was motivated by taking Model 3 and noticing that some variables, namely 'maritl' and 'race,' do not have any factor levels where the smoothing component is significant. It was built as shown below:

```
model4 = gam(logwage ~ year  + maritl + race
          + s(age, by = education) + education
          + s(age, by = jobclass) + jobclass
          + s(age, by = health) + health
          + s(age, by = health_ins) + health_ins, data = train)
```

To understand this better, we eliminated variables one at a time. We find that smoothing on 'race' can be safely eliminated but the smoothing on 'maritl' is kept as we get a low p-value indicating that the smoothed term is significant.

This model has an AIC of 490.96 which is higher than the 484.21 of Model 3. The R-squared is 41% and the RMSE is 0.283.

Model 5 (classification model):
This model is motivated by the idea that we may not wish to predict an exact wage, but rather whether an individual earns a wage over a certain level. It also serves to demonstrate that generalized additive models can accommodate different response variable distributions. The model is build as shown below:

```
Y = as.factor(I(train$logwage > log(100)))
model5 = gam(Y ~ year  + s(age, by = maritl)+ maritl + race
        +  s(age, by = education) + education
        +  s(age, by = jobclass) + jobclass + s(age, by = health)
        +  health + s(age, by = health_ins) + health_ins,
           data = train, family = binomial)
```

We are interested in a wage cut off of 100 ($100,000) which translates to a 'logwage' cutoff of 4.60517. This model does not fit the data particularly well with an adjusted R-squared of 32%.

Model 6 (ACE Model)
Next, we built an ACE model using the acepack library as follows:

```
Y = train$logwage
```

```
X = model.matrix(Y ~ ., data= subset(train, select =
-c(logwage)))
acefit = ace(X,Y)
y = acefit$ty
x = acefit$tx
model6 = lm(y ~ .-1, data = data.frame(x))
```

The motivation of this model is to work with a transformed version of the response variable. This model gives the one of the strongest fits of the models so far with an R-squared of 42%. AIC comparison is not meaningful here because of the transformations to the predictors and response variable.

Model 7 (AVAS Model)
This model also uses the acepack library and was built as follows:

```
avasfit = avas(X,Y)
y = avasfit$ty
x = avasfit$tx
model7 = lm(y ~ . -1, data = data.frame(x))
```

The motivation of this model is to trade off some fit accuracy to get a constant variance. If it is more important to know how much you are right or wrong about an estimate rather than the estimate itself, this model is useful.

We duly see the lower R-squared of 40% but examining the residuals does not show a marked improvement in the shape of the residuals.

Model 8 (MARS with No Interactions)
The MARS approach without interactions is a type of additive model and was built as follows:

```
model8 = earth(logwage ~ . , data = train, degree = 1)
```
We compare this model with Model 4. The R-squared of this model is 41%.

We can also examine the cutoffs used here and the variable importance for insights as shown in Table 3.

	Overall <dbl>
health_ins2.No	100.0000000
education5.AdvancedDegree	80.3158557
education4.CollegeGrad	62.9479420
maritl2.Married	50.8752999
age.Q	36.5532690
education3.SomeCollege	28.3368615
year.L	20.2548637
health2.>=VeryGood	14.2340707
age^12	11.7067459
age^41	10.0222680

Table 3. Top 10 important variables in MARS

We see, from Table 3., that not having health insurance is the most important factor level and that it heavily reduces the income prediction. The next most important factors are having an advanced degree or college degree and your age. These are shown to be the strongest influences on the predictions and intuitively match up to what we would expect.

Conclusion/Discussion

For comparison, we look at the models and their performance as shown in Table 4.

Model	Adjusted R^2	AIC	RMSE
Linear Model	38%	585.03	0.284
Model 1	36%	531	44.22
Model 2	40%	528.73	0.281
Model 3	42%	484.21	0.284
Model 4	41%	490.96	0.283

Table 4. Different models built and their performance metrics

From Table 4., we see that Model 3 performs the best since it has a higher adjusted R^2 and the lowest AIC.

To see how the predictions of Model 2 compare with the actual data in our test set, we plotted a graph showing prediction intervals shown by Figure 12.

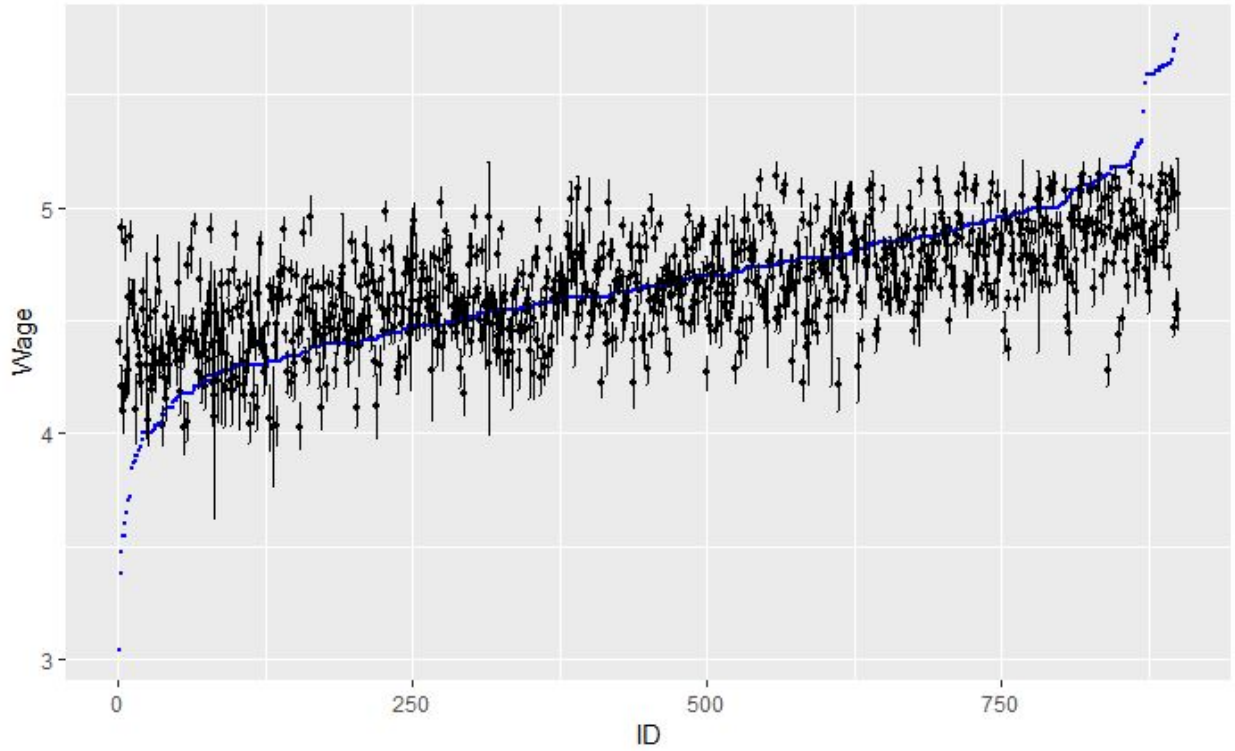

Figure 12. Actual data vs predicted data with prediction intervals

In Figure 12., the blue dots represent the actual values of wage in the test set while the black dots represent the predictions made by Model 2 along with respective prediction intervals.

Contribution Of Each Team Member

Name	**Tasks**
Ann Joseph	• Project Plan • Report
Caroline Park	• Data cleaning • Presentation
Meher Pooja Pranavi Punyamanthala	• Data Cleaning • Exploratory Data Analysis

Ross Graham	• Data modelling

References

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013) *An Introduction to Statistical Learning with applications in R*. Springer

Faraway J. J. (2009). *Extending the Linear Model with R (Second Edition).* Chapman & Hall/CRC

Appendix

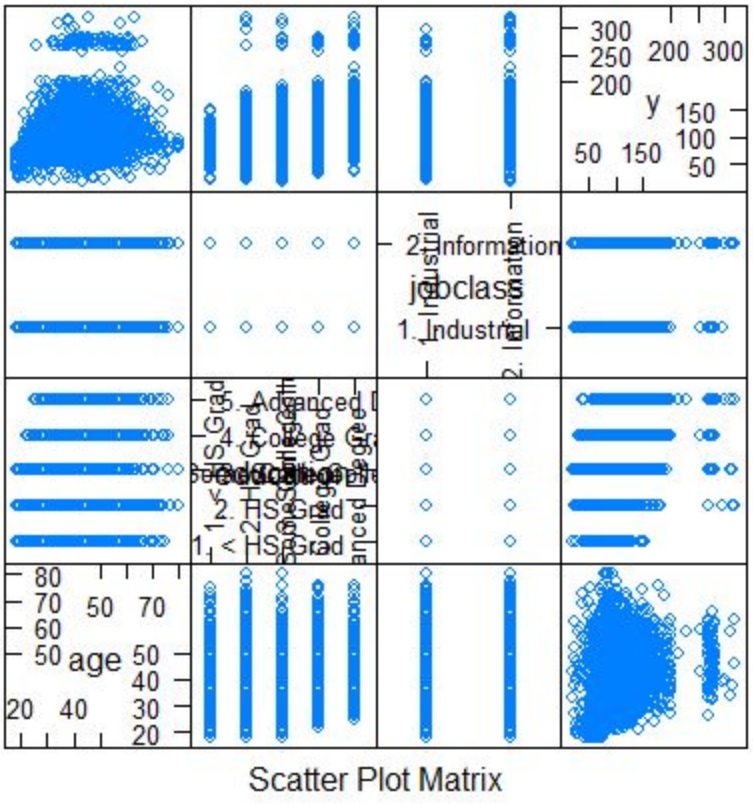

Figure 1. Relationship between all variables

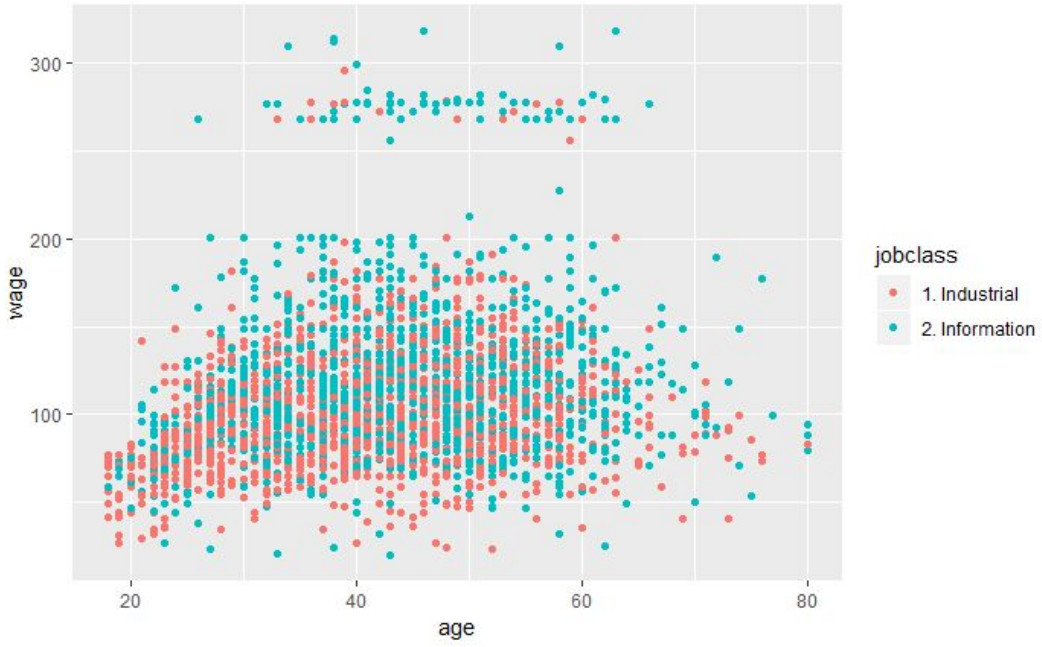

Figure 2. Relationship between age and wage factored by job class

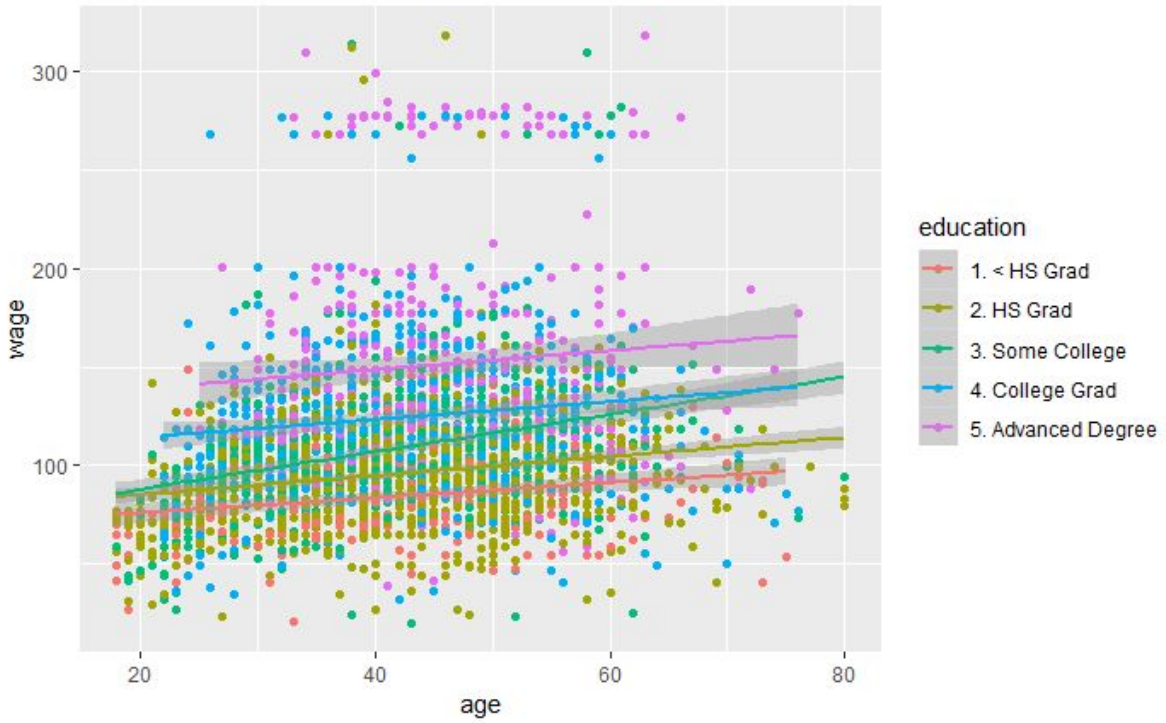

Figure 3. Relationship between age and wage factored by education

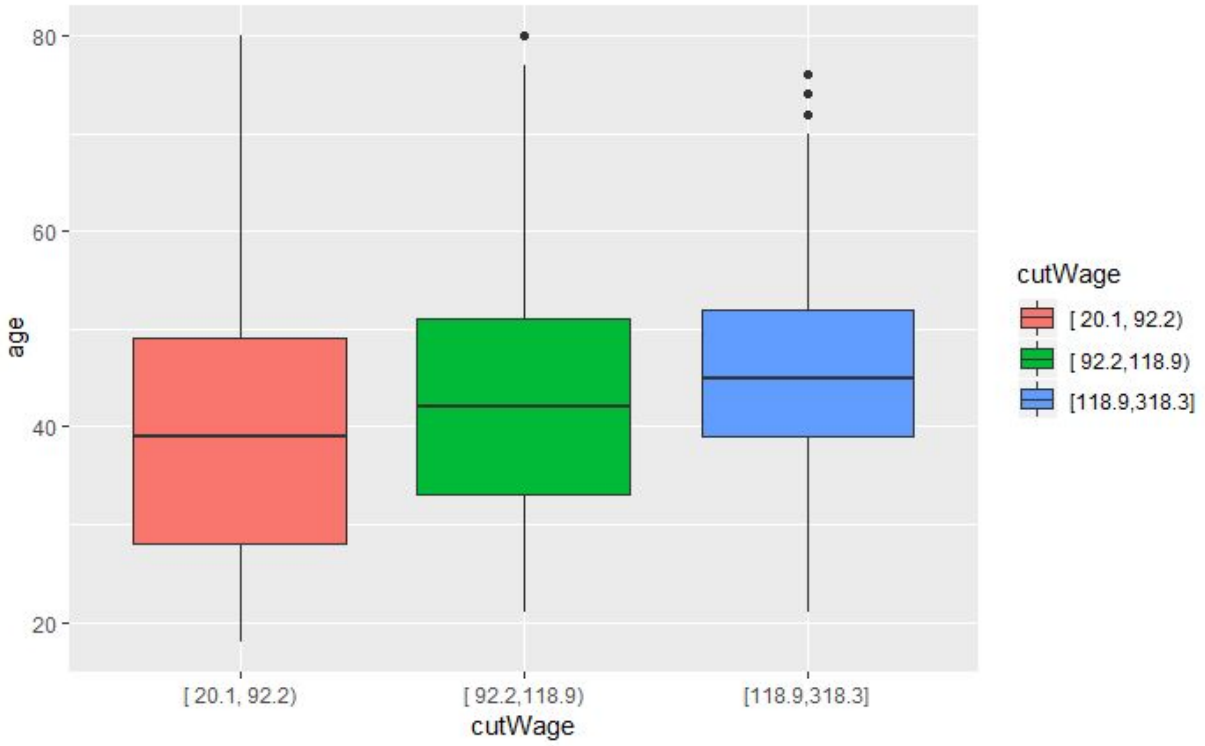

Figure 4. Relationship between the wage groups and age

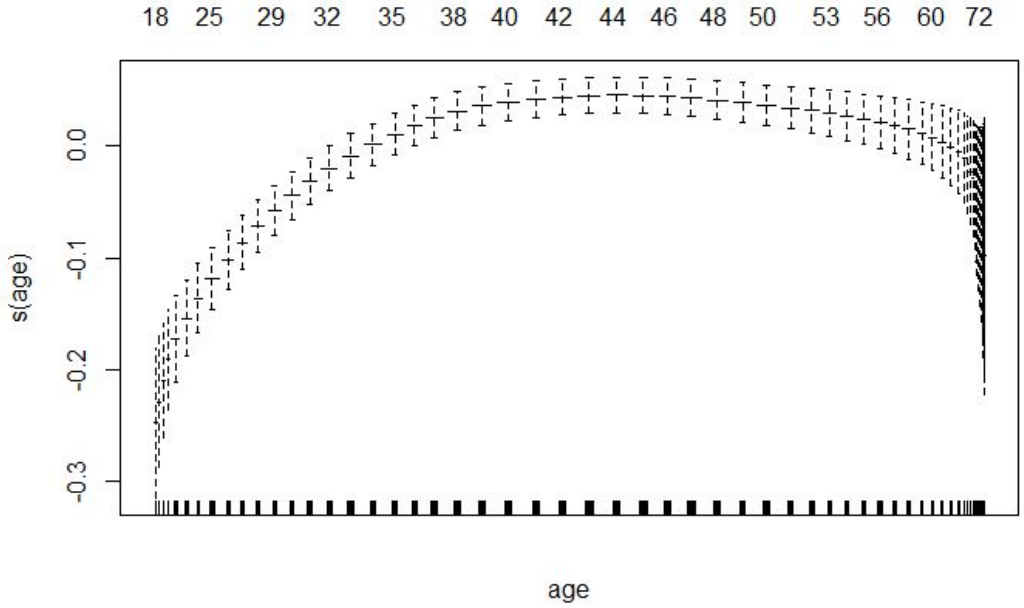

Figure 5. Transformation of age

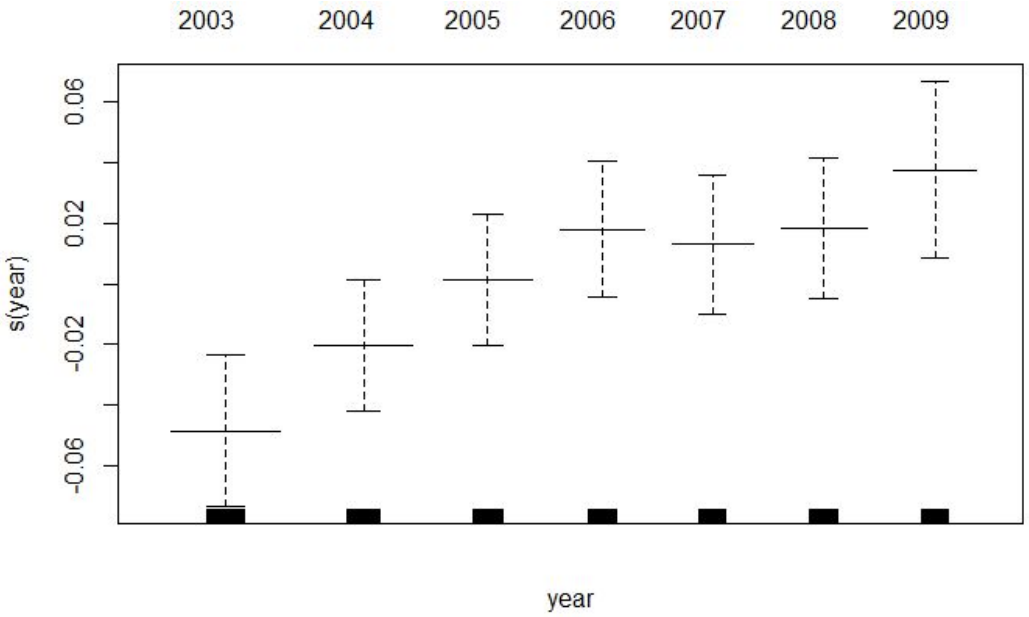

Figure 6. Transformation of year

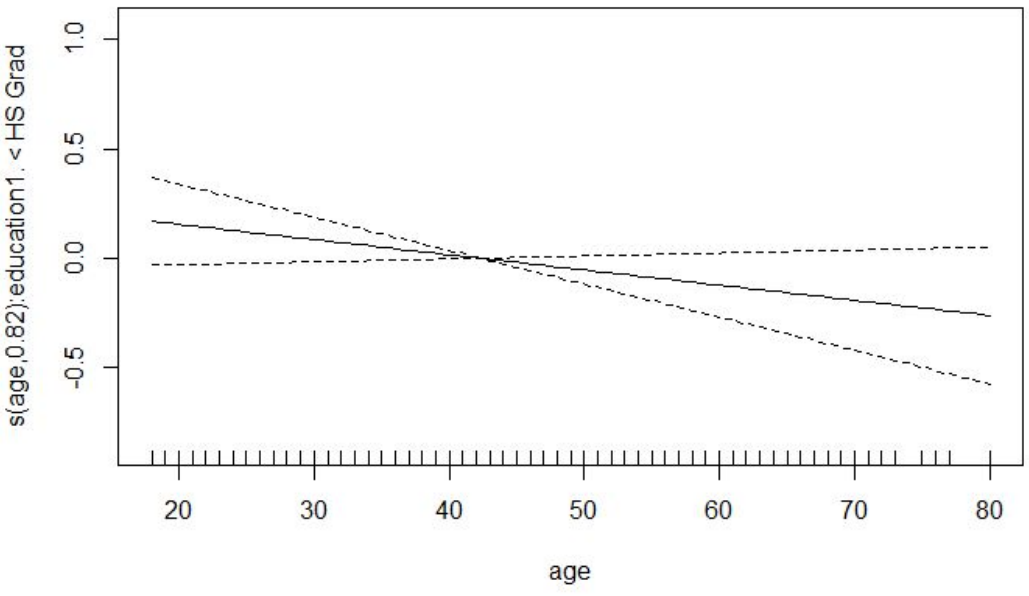

Figure 7. Transformation of age by education less than high school

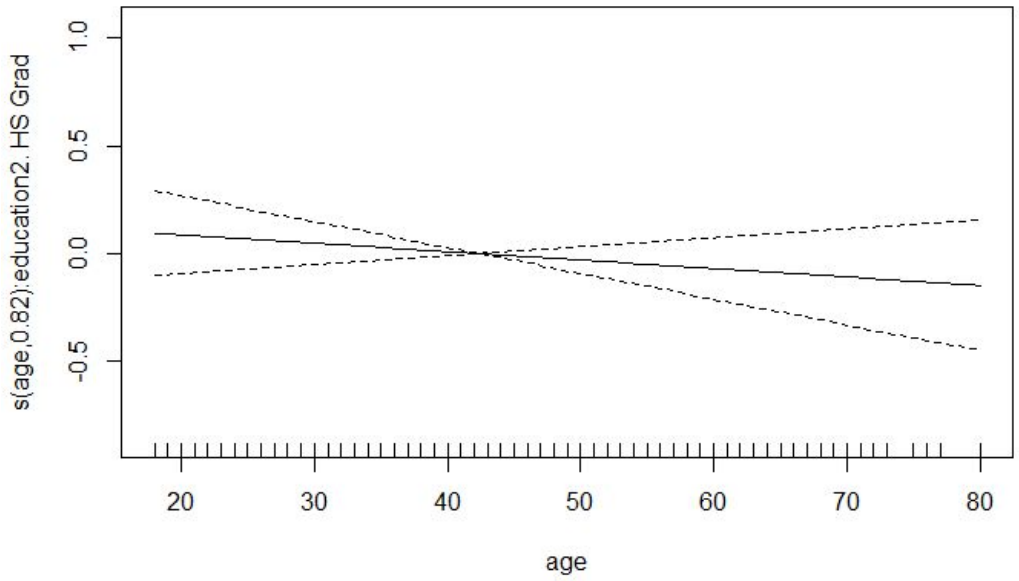

Figure 8. Transformation of age by high school education

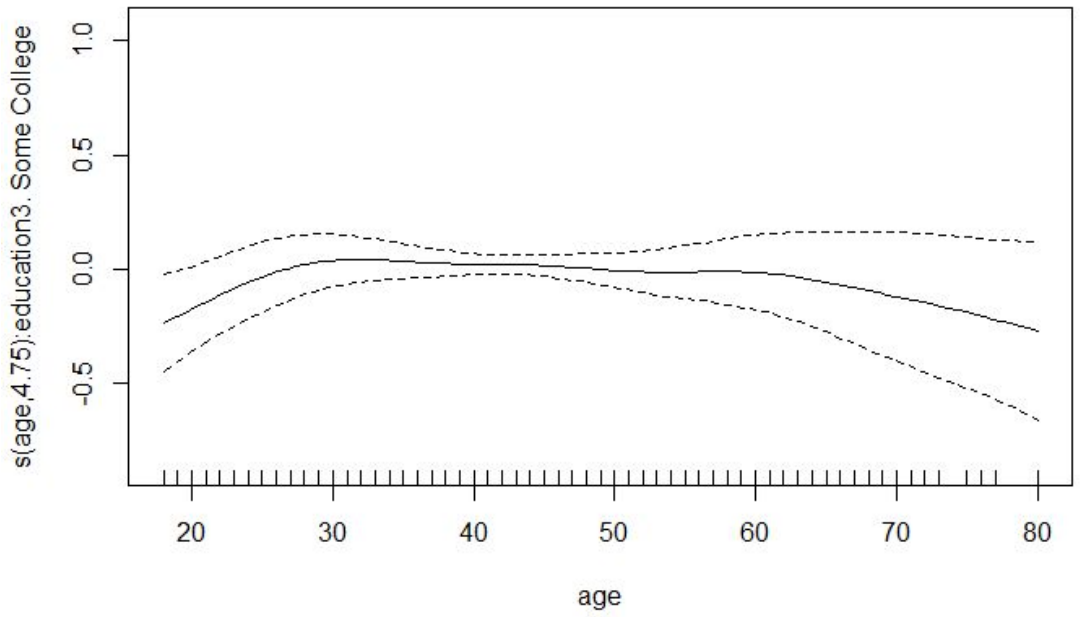

Figure 9. Transformation of age by ‘Some College’ education

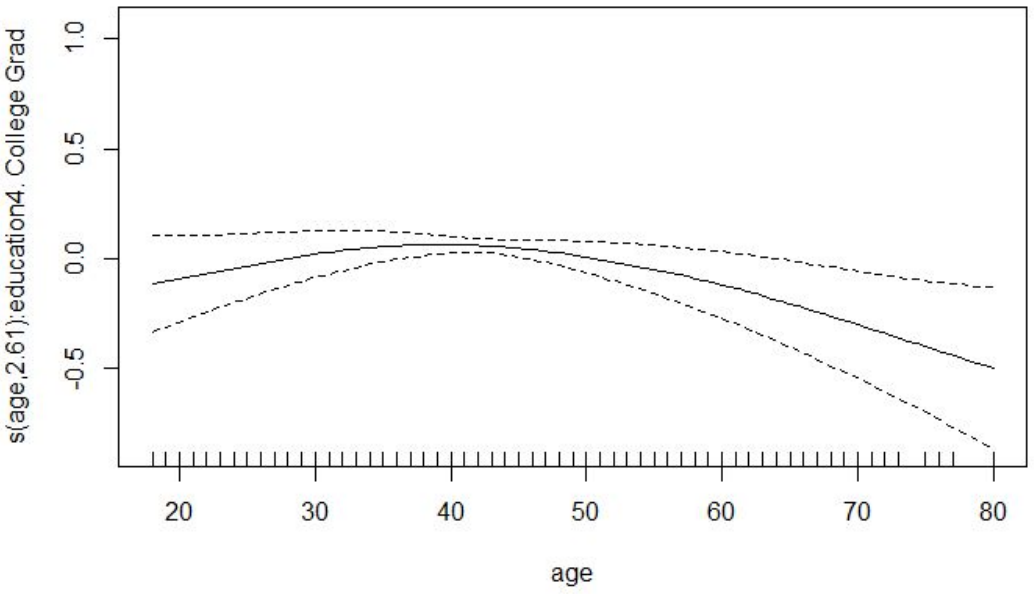

Figure 10. Transformation of age by 'College Grad' education

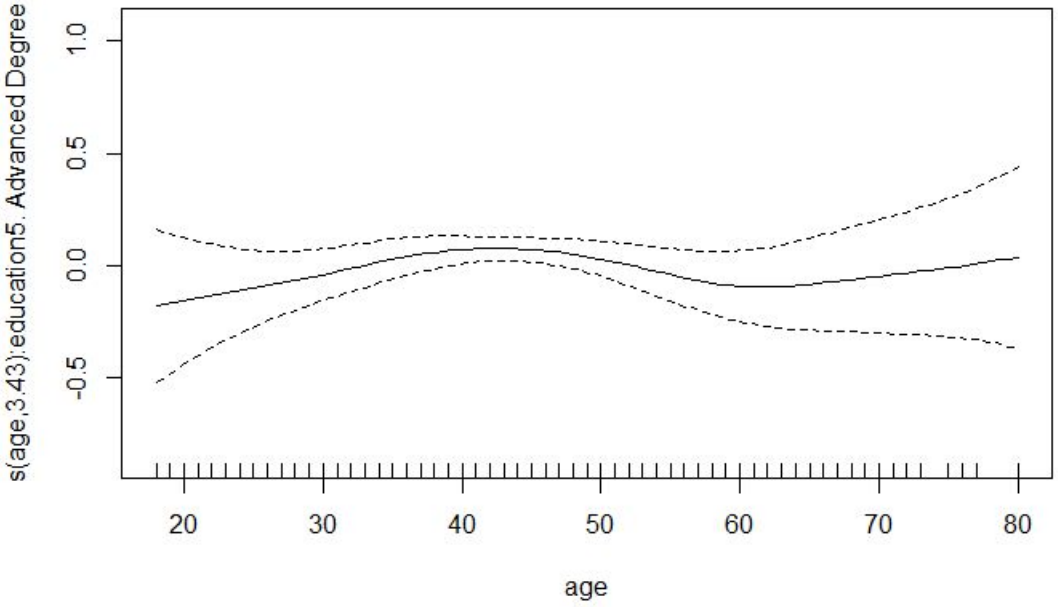

Figure 11. Transformation of age by 'Advanced degree' education

R Code for Data Modelling:

```
# Importing Data
data(Wage,package="ISLR")
data = Wage
# setup data
data = subset(data, select = -c(wage,region))

# Splitting the data into 7:3
smp_size <- floor(0.70 * nrow(data))
set.seed(123)
train_ind <- sample(seq_len(nrow(data)), size = smp_size)
train <- data[train_ind, ]
test <- data[-train_ind, ]

# Linear model
olm = lm(logwage~., data = train)
olm = stepAIC(olm, trace = FALSE)
# coefficient output
out=round(summary(olm)$coeff,4)
print(xtable(out,caption = "\\tt Linear Model Coefficient Output",digits =4),
      floating = TRUE,latex.environments = "center")
# fit output
stats = c(AIC(olm), summary(olm)$r.squared,summary(olm)$adj.r.squared,
      summary(olm)$sigma)
stats=t(stats)
colnames(stats)=c("AIC", "R-Squared","Adjusted R-Squared","Residual Standard Error")
rownames(stats)="Linear Model"
print(xtable(stats,caption = "\\tt Linear Model Fit Output",digits =4),
      floating = TRUE, latex.environments = "center")
summary(olm)
AIC(olm)
#Predicting using the test set
lm_pred<-predict(olm, test[,-9])
error <- lm_pred-test$logwage
rmse_lm <- sqrt(mean(error^2)) #RMSE
rmse_lm
```

Model 1- All Predictors, All continuous smoothing and no factor smoothing

```
require(gam)
#Ordering the factor variables in train and test
train$year<- ordered(train$year)
train$age<-ordered(train$age)
test$year<-ordered(test$year)
test$age<-ordered(test$age)
#Fitting the GAM model with continuous smoothing and no factor smoothing
ammgcv <- gam(logwage ~ s(year) + s(age)+ ordered(maritl) + ordered(race) +
ordered(education) + ordered(jobclass) +
        ordered(health) + ordered(health_ins),data = train)
#summary or gist of the model
summary(ammgcv)
#Predicting using the test set
ammgcv_pred<-predict(ammgcv, test[,-9])
#RMSE
error <- ammgcv_pred-test$logwage
rmse_ammgcv <- sqrt(mean(error^2)) #RMSE
Rmse_ammgcv
sse <- sum((ammgcv_pred - test$logwage)^2)
sst <- sum((test$logwage - mean(test$logwage))^2)
# R squared
rsq <- 1 - (sse / sst)
rsq*100

# Model 2- All Predictors, Age Smoothing, No Factor Smoothing
ammgcv2 = gam(logwage ~ year + s(age) + maritl + race + education + jobclass +
   health + health_ins, data = train)
#Plotting the Model
par(mfrow=c(1,3))
plot(ammgcv2)
summary(ammgcv2)
AIC(ammgcv2)
anova(ammgcv,ammgcv2,test="F")
#Predicting using the test set
ammgcv2_pred<-predict(ammgcv2, test[,-9])
error <- ammgcv2_pred-test$logwage
rmse_ammgcv2 <- sqrt(mean(error^2)) #RMSE
rmse_ammgcv2
#Prediction Interval graph
```

```
preds   <- predict(ammgcv2, test[,-9], se.fit = TRUE)
my_data <- data.frame(test,
                      logCOST = test$logwage,
                      mu   = preds$fit,
                      low  = preds$fit - 1.96 * preds$se.fit,
                      high = preds$fit + 1.96 * preds$se.fit)
my_data_ordered <- arrange(my_data,logCOST)
head(my_data_ordered)
#my_data_ordered <- my_data[order(logCOST),]
my_data_ordered <- tibble::rowid_to_column(my_data_ordered, "ID")
head(my_data_ordered)
PI <- ggplot(my_data_ordered, aes(x = ID, y = mu)) +
  geom_point(size = 1) +
  geom_point(aes(x = ID, y = logCOST), col = 12, size = 0.5) +
  geom_errorbar(aes(ymax = high, ymin = low))
PI + ylab("Wage")
#Model Diagnostics
par(mfrow=c(1,2))
plot(residuals(ammgcv2)~predict(ammgcv2),xlab="Predicted",ylab="Residuals")
abline(h=0)
qqnorm(residuals(ammgcv2),main="")
qqline(residuals(ammgcv2))

# Model 3- All Predictors, Age Smoothing, Factor Smoothing
ammgcv3 = gam(logwage ~ year + s(age, by =maritl) + maritl + s(age, by = race) + race
+ s(age, by = education) + education + s(age, by = jobclass) + jobclass + s(age, by =
health) + health + s(age, by = health_ins) + health_ins, data = train)
summary(ammgcv3)
AIC(ammgcv3)
#Predicting using the test set
ammgcv3_pred<-predict(ammgcv3, test[,-9])
error <- ammgcv3_pred-test$logwage
rmse_ammgcv3 <- sqrt(mean(error^2)) #RMSE
rmse_ammgcv3

# Model 4 - Fewer Predictors, Age Smoothing, Factor Smoothing
# both
```

```
ammgcv4 = gam(logwage ~ year  + maritl + race +s(age, by = education) + education +
s(age, by = jobclass) + jobclass +  s(age, by = health) + health + s(age, by = health_ins)
+ health_ins, data = train)
anova(ammgcv3,ammgcv4,test="F")
# race smoothing eliminated
ammgcv4 = gam(logwage ~ year  + s(age, by =maritl)+ maritl + race +
            s(age, by = education) + education + s(age, by = jobclass) + jobclass +
            s(age, by = health) + health + s(age, by = health_ins) + health_ins, data = train)
anova(ammgcv3,ammgcv4,test="F")
# maritl smoothing eliminated
ammgcv4 = gam(logwage ~ year  + maritl + race + s(age, by = race) +
            s(age, by = education) + education + s(age, by = jobclass) + jobclass +
            s(age, by = health) + health + s(age, by = health_ins) + health_ins, data = train)
anova(ammgcv3,ammgcv4,test="F")
# final model
ammgcv4 = gam(logwage ~ year  + s(age, by =maritl)+ maritl + race +
            s(age, by = education) + education + s(age, by = jobclass) + jobclass +
            s(age, by = health) + health + s(age, by = health_ins) + health_ins, data = train)
summary(ammgcv4)
AIC(ammgcv4)
#Predicting using the test set
ammgcv4_pred<-predict(ammgcv4, test[,-9])
error <- ammgcv4_pred-test$logwage
rmse_ammgcv4 <- sqrt(mean(error^2)) #RMSE
rmse_ammgcv4

# Model 5 - Classification Model
# new reponse
Y = as.factor(I(train$logwage > log(100)))
# model
ammgcv5 = gam(Y ~ year  + s(age, by =maritl)+ maritl + race +
            s(age, by = education) + education + s(age, by = jobclass) + jobclass +
            s(age, by = health) + health + s(age, by = health_ins) + health_ins,
            data = train, family = binomial)
summary(ammgcv5)
AIC(ammgcv5)
lr = glm(Y~., data = train, family = "binomial")
summary(lr)
anova(ammgcv5,lr,test="F")
```

```
# Model 6 - ACE Model
# new variable
Y = train$logwage
X = model.matrix(Y~., data= subset(train, select = -c(logwage)))
# model
acefit = ace(X,Y) ## problem here
y = acefit$ty
x = acefit$tx
acemod = lm(y~.-1, data = data.frame(x))
summary(acemod)
#Transformations done on the response and predictor variables
par(mfrow=c(1,3))
plot(train$logwage,acefit$ty,xlab="logwage", ylab=expression(theta(logwage)))
plot(train$year,acefit$tx[,2],xlab="year",ylab="f(year)")
plot(train$age,acefit$tx[,3],xlab="age",ylab="age")

# Model 7 - AVAS Model
# new variable
Y = train$logwage
X = model.matrix(Y~., data= subset(train, select = -c(logwage)))
# model
avasfit = avas(X,Y) ## problem here
y = avasfit$ty
x = avasfit$tx
avasmod = lm(y~.-1, data = data.frame(x))
summary(avasmod)
#Transformations done on the response and predictor variables
par(mfrow=c(1,3))
plot(train$logwage,avasfit$ty,xlab="logwage", ylab=expression(theta(logwage)))
plot(train$year,avasfit$tx[,2],xlab="year",ylab="f(year)")
plot(train$age,avasfit$tx[,3],xlab="age",ylab="age")


# Model 8 - MARS Model, No Interactions
# new variable
Y = train$logwage
X = subset(train, select = -c(logwage))
# model
mars1 = earth(Y ~ ., data = X, degree = 1)
```

```
summary(mars1)
varImp(mars1)

# Model 9 - MARS Model, Interactions
# new variable
Y = data$logwage
# model
mars2 = earth(Y ~ ., data = X, degree = 2)
summary(mars2)
varImp(mars2)
```