Exercise 7

Konstantinos Vakalopoulos 12223236

2022-12-21

Premilinary work

First, the data Orange Juice (OJ) from the package ISLR were loaded

```
library(ISLR)
data(OJ,package="ISLR")
data <- OJ
```

and missing values were omitted.

```
data <- na.omit(data)
```

Below are presented extra information about the data.

```
str(data)
```

```
## ’data.frame':    1070 obs. of  18 variables:
##  $ Purchase      : Factor w/ 2 levels "CH","MM": 1 1 1 2 1 1 1 1 1 1 ...
##  $ WeekofPurchase: num  237 239 245 227 228 230 232 234 235 238 ...
##  $ StoreID       : num  1 1 1 1 7 7 7 7 7 7 ...
##  $ PriceCH       : num  1.75 1.75 1.86 1.69 1.69 1.69 1.69 1.75 1.75 1.75 ...
##  $ PriceMM       : num  1.99 1.99 2.09 1.69 1.69 1.99 1.99 1.99 1.99 1.99 ...
##  $ DiscCH        : num  0 0 0.17 0 0 0 0 0 0 0 ...
##  $ DiscMM        : num  0 0.3 0 0 0 0 0.4 0.4 0.4 0.4 ...
##  $ SpecialCH     : num  0 0 0 0 0 0 1 1 0 0 ...
##  $ SpecialMM     : num  0 1 0 0 0 1 1 0 0 0 ...
##  $ LoyalCH       : num  0.5 0.6 0.68 0.4 0.957 ...
##  $ SalePriceMM   : num  1.99 1.69 2.09 1.69 1.69 1.99 1.59 1.59 1.59 1.59 ...
##  $ SalePriceCH   : num  1.75 1.75 1.69 1.69 1.69 1.69 1.69 1.75 1.75 1.75 ...
##  $ PriceDiff     : num  0.24 -0.06 0.4 0 0 0.3 -0.1 -0.16 -0.16 -0.16 ...
##  $ Store7        : Factor w/ 2 levels "No","Yes": 1 1 1 1 2 2 2 2 2 2 ...
##  $ PctDiscMM     : num  0 0.151 0 0 0 ...
##  $ PctDiscCH     : num  0 0 0.0914 0 0 ...
##  $ ListPriceDiff : num  0.24 0.24 0.23 0 0 0.3 0.3 0.24 0.24 0.24 ...
##  $ STORE         : num  1 1 1 1 0 0 0 0 0 0 ...
```

The numeric variables: “STORE”, “StoreID”, “SpecialCH”, “SpecialMM” were transformed into factors.

```
catVars = c("STORE","StoreID","SpecialCH","SpecialMM")
data[catVars] <- lapply(data[catVars], as.factor)
```

The plots below present the factor and the numeric variables of the OJ data set.

```
library(ggplot2)
library(tidyverse)
```

```
data %>%
  keep(is.factor) %>%
  gather() %>%
  ggplot(aes(value, fill=value)) +
  facet_wrap(~ key, scales = "free") +
  geom_bar() +
  theme(legend.position="none")
```

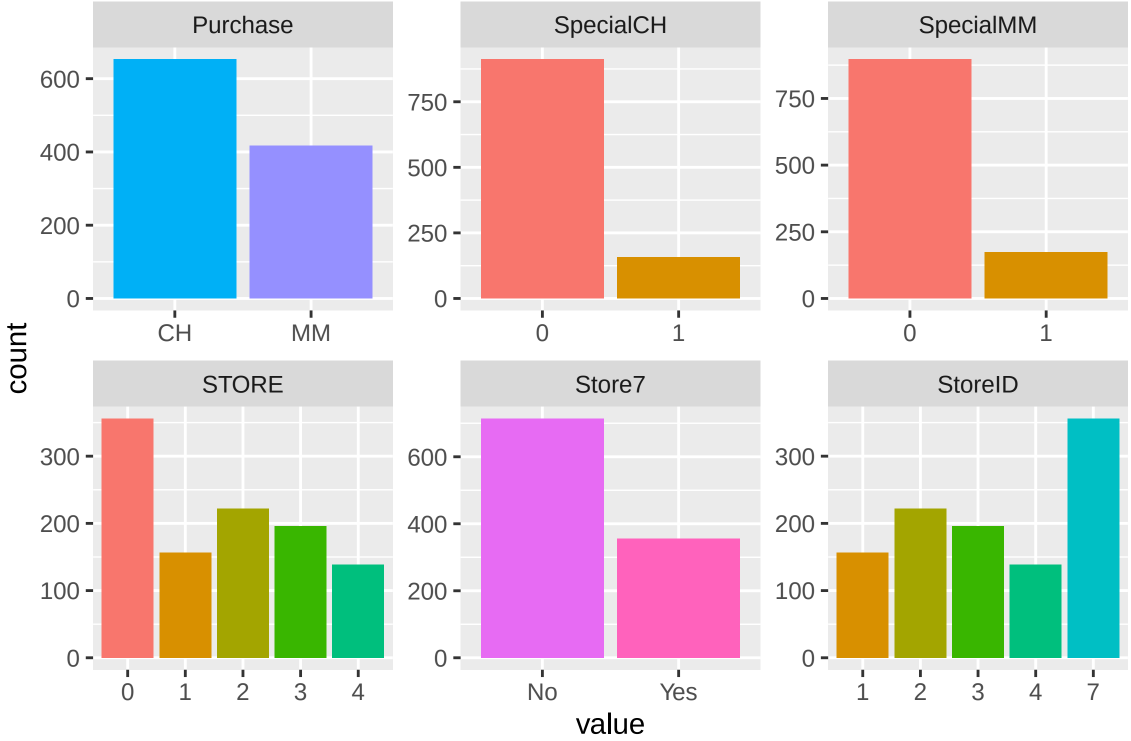

Figure 1: Factor variables

```
data %>%
  keep(is.numeric) %>%
  gather() %>%
  ggplot(aes(value,fill=key)) +
  facet_wrap(~ key, scales = "free") +
  geom_histogram(bins=sqrt(nrow(data))) +
  theme(legend.position="none")
```

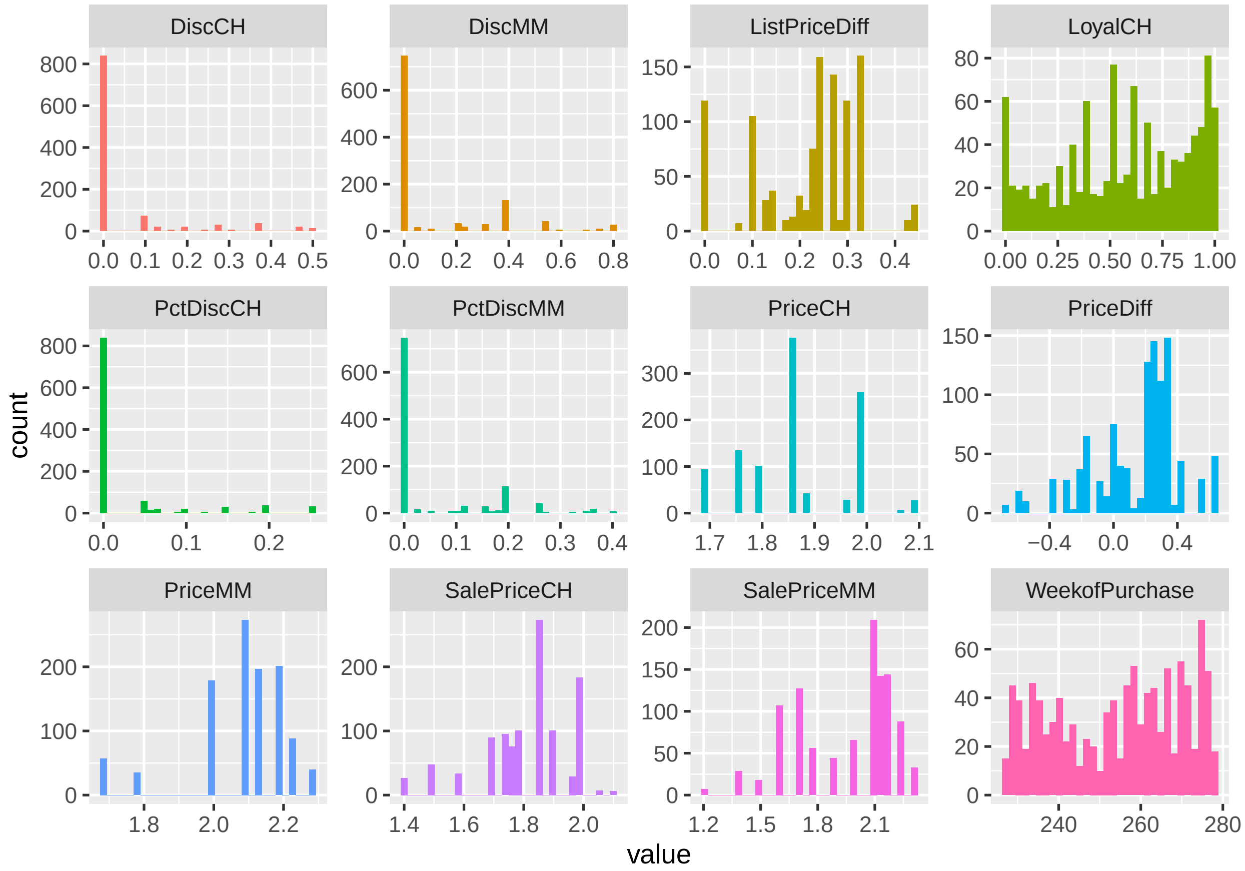

Figure 2: Numeric variables

Question 1(a)

The data were randomly split into train (2/3 of the observations) and test (1/3 of the observations) set.

```
set.seed(12223236)
n <- nrow(data)
train <- sample(1:n,round(n*2/3))
test <- (1:n)[-train]
```

From the library mgcv, the function gam() was used in order to implement Generalized Additive Models. The smooth functions in GAMs can be defined for every variable by s(variable). With the parameter k you could also set an upper bound for the degrees of freedom. In our case k=3 was used on the explanatory variable “PriceMM”. The rest of the variables were simply used as s(variable). Also, for the variables “STORE”, “StoreID”, “SpecialCH” and “SpecialMM”, smooth functions were not used because they are factors. Thus, the final GAM model with the chosen formula is:

```
library(mgcv)
```

```
mod.gam <-gam(Purchase ~ s(PriceMM, k=3)+s(WeekofPurchase)+s(PriceCH)+s(DiscCH)
              +s(DiscMM)+s(LoyalCH)+s(SalePriceMM)+s(SalePriceCH)+s(PriceDiff)
              +s(PctDiscMM)+s(PctDiscCH)+s(ListPriceDiff)+StoreID+SpecialCH
              +SpecialMM+Store7+STORE
              ,data=data, family="binomial", subset=train)
```

Question 1(b)

For interpretation of the gam model, the function summary was used.

```
summary(mod.gam)
```

```
## 
## Family: binomial 
## Link function: logit 
## 
## Formula:
## Purchase ~ s(PriceMM, k = 3) + s(WeekofPurchase) + s(PriceCH) + 
##     s(DiscCH) + s(DiscMM) + s(LoyalCH) + s(SalePriceMM) + s(SalePriceCH) + 
##     s(PriceDiff) + s(PctDiscMM) + s(PctDiscCH) + s(ListPriceDiff) + 
##     StoreID + SpecialCH + SpecialMM + Store7 + STORE
## 
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -0.40560    0.38131  -1.064   0.2875  
## StoreID2    -0.22175    0.48021  -0.462   0.6442  
## StoreID3     0.00000    0.00000     NaN      NaN  
## StoreID4     0.00000    0.00000     NaN      NaN  
## StoreID7     0.00000    0.00000     NaN      NaN  
## SpecialCH1   0.48624    0.43555   1.116   0.2643  
## SpecialMM1   0.64514    0.36506   1.767   0.0772 .
## Store7Yes   -1.22345    0.48715  -2.511   0.0120 *
## STORE1      -0.69242    0.52419  -1.321   0.1865  
## STORE2       0.00000    0.00000     NaN      NaN  
## STORE3       0.06202    0.42513   0.146   0.8840  
## STORE4       0.00000    0.00000     NaN      NaN  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Approximate significance of smooth terms:
##                         edf    Ref.df  Chi.sq p-value    
## s(PriceMM)        3.122e-03 6.170e-03   0.003  0.9535    
## s(WeekofPurchase) 1.000e+00 1.000e+00   0.002  0.9609    
## s(PriceCH)        1.260e+00 1.475e+00   0.829  0.6362    
## s(DiscCH)         1.000e+00 1.000e+00   0.262  0.6089    
## s(DiscMM)         1.000e+00 1.000e+00   0.179  0.6725    
## s(LoyalCH)        1.000e+00 1.000e+00 152.996  <2e-16 ***
## s(SalePriceMM)    1.410e-05 2.724e-05   0.000  0.5000    
## s(SalePriceCH)    5.648e-06 1.094e-05   0.000  0.5000    
## s(PriceDiff)      3.461e-06 6.821e-06   0.000  0.5000    
## s(PctDiscMM)      4.783e+00 5.798e+00  11.697  0.0562 .  
## s(PctDiscCH)      1.000e+00 1.000e+00   0.503  0.4782    
## s(ListPriceDiff)  1.000e+00 1.000e+00   9.457  0.0021 ** 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Rank: 104/113
## R-sq.(adj) =  0.504   Deviance explained = 44.4%
## UBRE = -0.19858  Scale est. = 1         n = 713
```

Based on the summary, regarding the factor variables (parametric coefficients), can be seen that some the coefficients of the variables “StoreID” and “STORE” are zero. Thus, these two variables are excluded from the model and the new GAM model is:

```
mod.gam <-gam(Purchase ~ s(PriceMM, k=3)+s(WeekofPurchase)+s(PriceCH)+s(DiscCH)
              +s(DiscMM)+s(LoyalCH)+s(SalePriceMM)+s(SalePriceCH)+s(PriceDiff)
              +s(PctDiscMM)+s(PctDiscCH)+s(ListPriceDiff)+SpecialCH
              +SpecialMM+Store7
              ,data=data, family="binomial", subset=train)
summary(mod.gam)
```

```
##
## Family: binomial
## Link function: logit
##
## Formula:
## Purchase ~ s(PriceMM, k = 3) + s(WeekofPurchase) + s(PriceCH) +
##     s(DiscCH) + s(DiscMM) + s(LoyalCH) + s(SalePriceMM) + s(SalePriceCH) +
##     s(PriceDiff) + s(PctDiscMM) + s(PctDiscCH) + s(ListPriceDiff) +
##     SpecialCH + SpecialMM + Store7
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.6567     0.1656  -3.965 7.34e-05 ***
## SpecialCH1    0.4880     0.4295   1.136  0.25596
## SpecialMM1    0.5373     0.3527   1.524  0.12761
## Store7Yes    -0.8551     0.2876  -2.973  0.00295 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Approximate significance of smooth terms:
##                         edf    Ref.df  Chi.sq p-value
## s(PriceMM)        4.718e-01 7.184e-01   1.009  0.3151
## s(WeekofPurchase) 1.000e+00 1.000e+00   0.437  0.5088
## s(PriceCH)        1.000e+00 1.000e+00   0.006  0.9410
## s(DiscCH)         1.000e+00 1.000e+00   0.355  0.5511
## s(DiscMM)         4.285e+00 5.171e+00   9.611  0.0931 .
## s(LoyalCH)        1.000e+00 1.000e+00 165.023  <2e-16 ***
## s(SalePriceMM)    1.085e-05 1.880e-05   0.000  0.9990
## s(SalePriceCH)    3.309e-05 6.519e-05   0.000  0.9987
## s(PriceDiff)      1.085e-05 2.137e-05   0.000  0.5000
## s(PctDiscMM)      1.001e+00 1.001e+00   0.711  0.3994
## s(PctDiscCH)      1.000e+00 1.000e+00   0.622  0.4302
## s(ListPriceDiff)  1.000e+00 1.000e+00   9.373  0.0022 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Rank: 101/105
## R-sq.(adj) =  0.505   Deviance explained =   44%
## UBRE = -0.2025  Scale est. = 1         n = 713
```

According to the inference table, it is observed that the significant variables are: “Store7”, “LoyalCH” and “ListPriceDiff”.

The complexity of the smooth functions can be defined from the effective degrees of freedom (i.e. the first column of the second inference table). For the variables “WeekofPurchase”, “DiscCH”, “DiscMM”. “LoyalCH”, “PctDiscCH”, “PriceCH” and “ListPriceDiff” the effective degrees of freedom are 1 or close to 1 which indicates that the smooth functions are linear. Regarding the parameter “PctDiscMM” has effective degrees of freedom 4.783. Therefore, the smooth function is more complex and wiggly (non linear). Furthermore, the effective degree of freedom for the variable “PriceMM” is 0.4718 and thus we cannot know the form of this variable’s smooth function. Finally, for the variables “SalePriceMM”, “SalePriceCH” and “PriceDiff” the effective degrees of freedom are close to zero. Thus, if the EDF is effectively equal to 0 then the term has been effectively.

Question 1(c)

In this part of the exercise, the explanatory variables against their smoothed values as they are used in the model were plotted.

```
plot(mod.gam,page=6,shade=TRUE,shade.col="yellow")
```

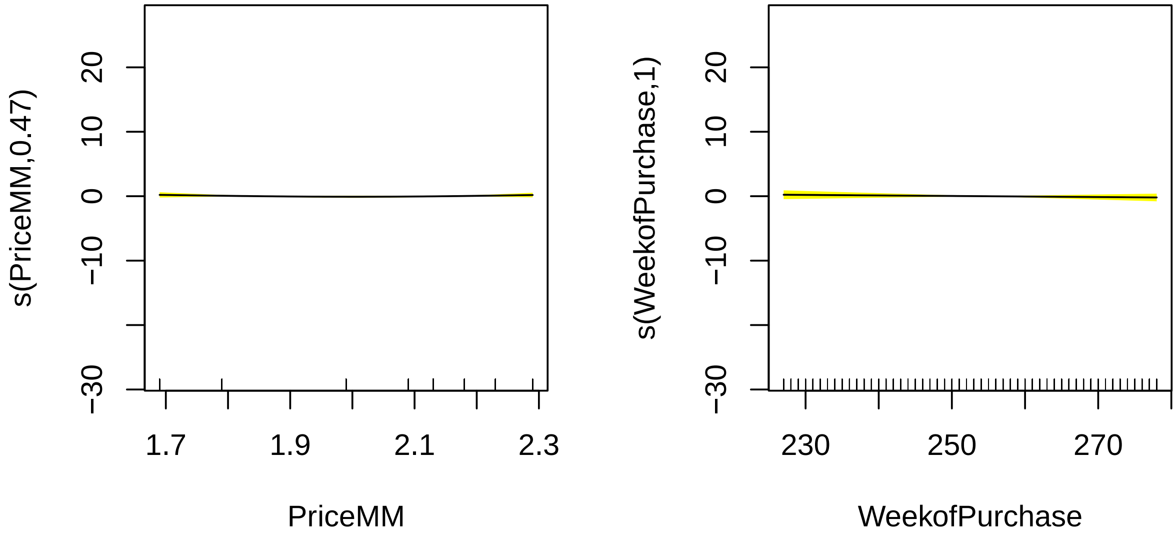

Figure 3: Explanatoty variables and Smoothed values

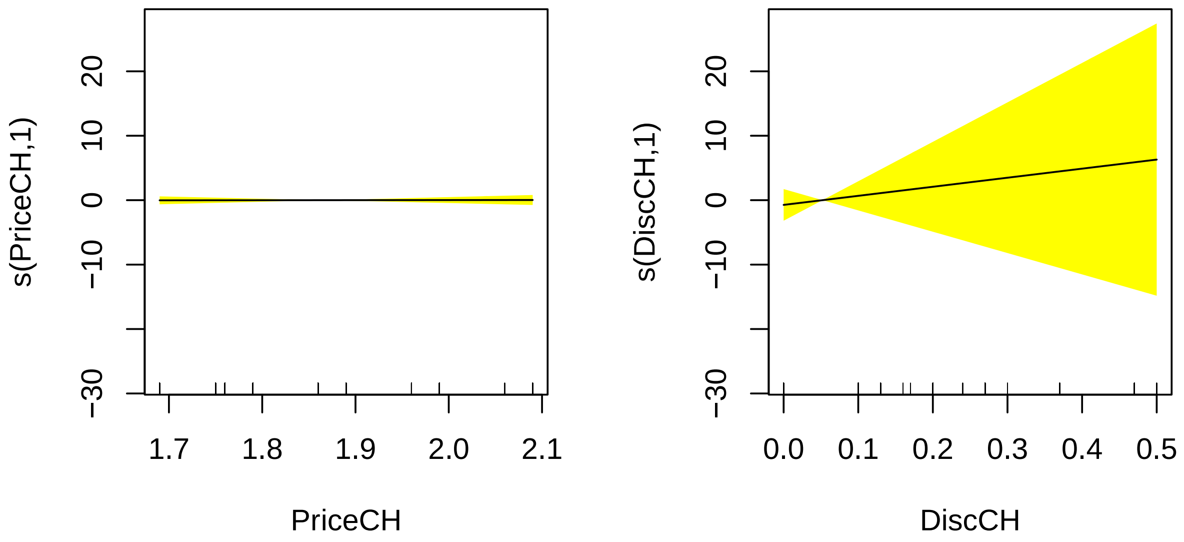

Figure 4: Explanatoty variables and Smoothed values

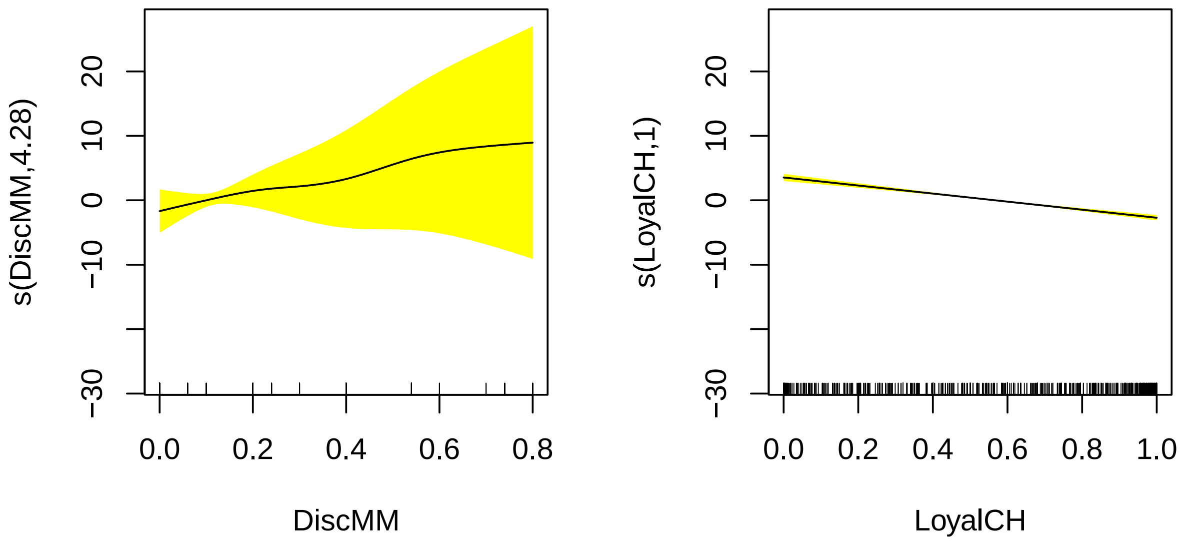

Figure 5: Explanatoty variables and Smoothed values

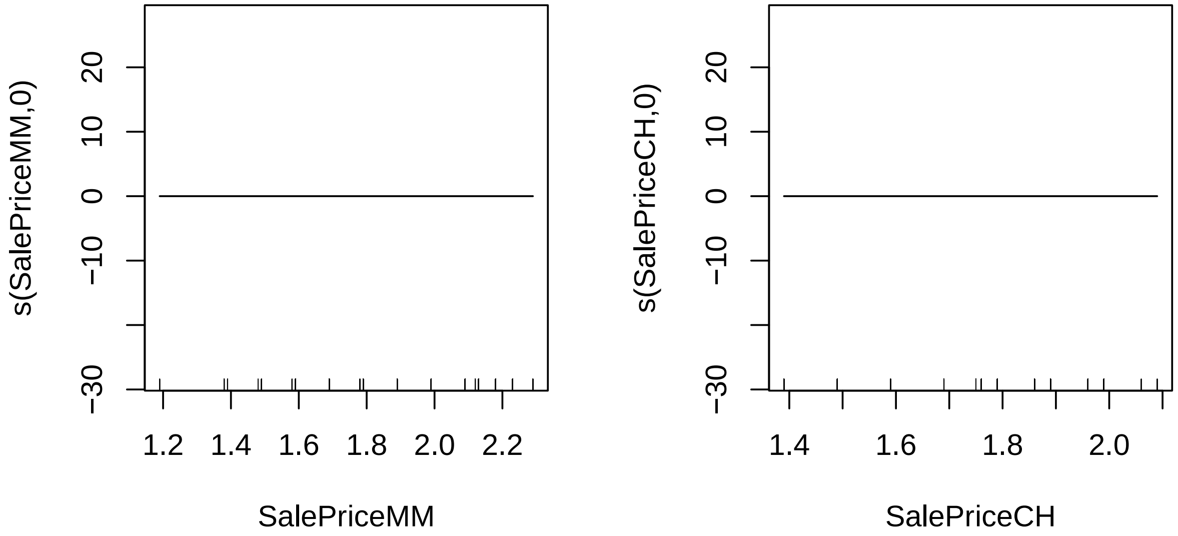

Figure 6: Explanatoty variables and Smoothed values

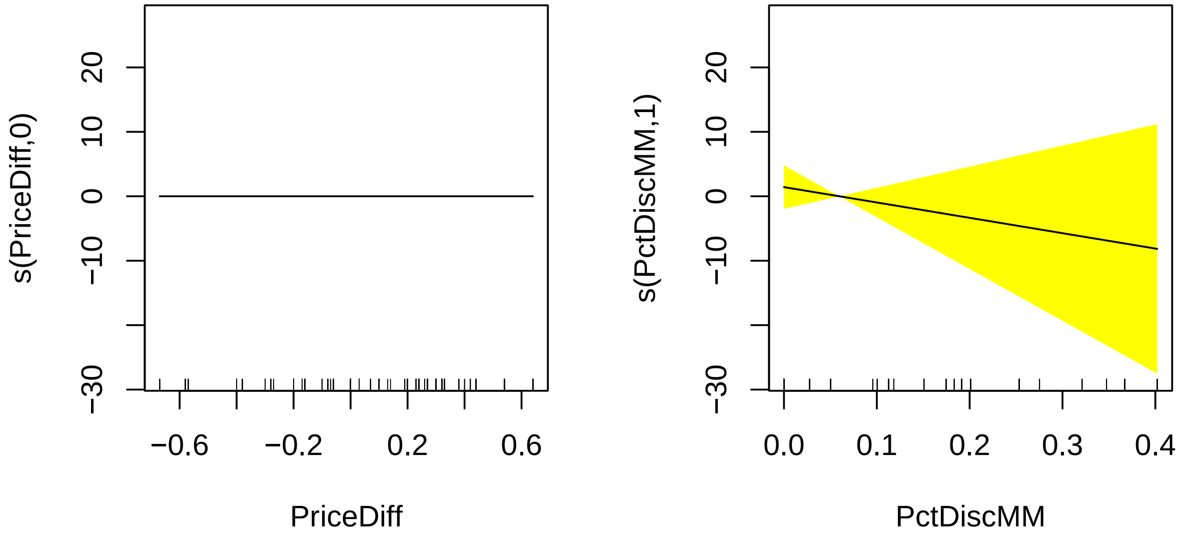

Figure 7: Explanatoty variables and Smoothed values

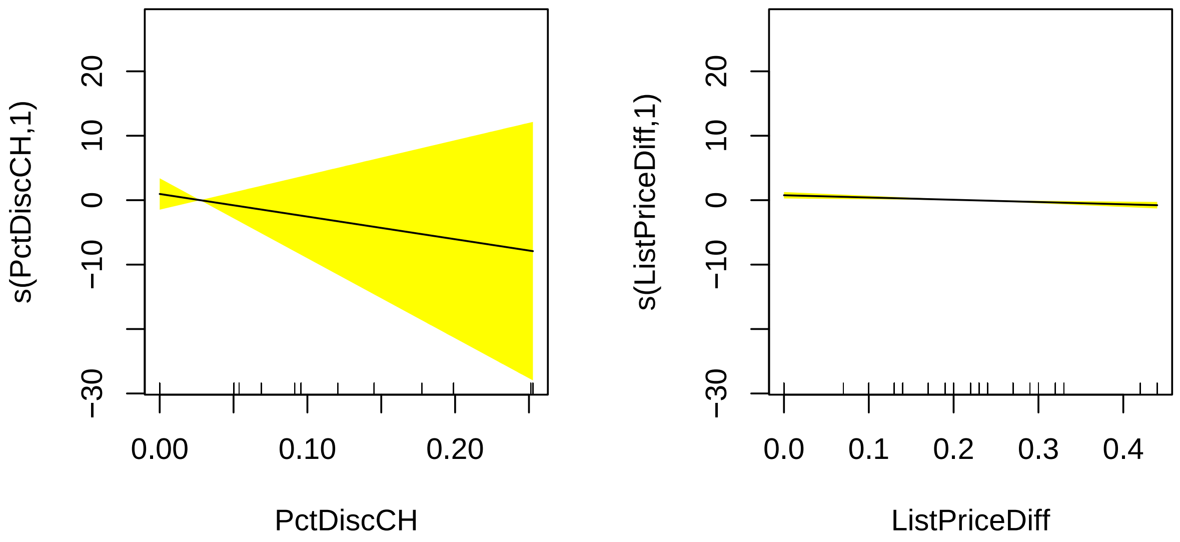

Figure 8: Explanatoty variables and Smoothed values

The linearity and non linearity, as mentioned in the question 1(b), of the smooth functions are, also, proven based on the plots. For the variables "WeekofPurchase", "DiscCH", "DiscMM". "LoyalCH", "PctDiscCH", "PriceCH", "ListPriceDiff" and also for the "PriceMM", which has edf equals to 0.4718, the smooth functions are linear. Regarding the variable "PctDiscMM", the non linearity is observed. Finally, the smooth functions for the variables "SalePriceMM", "SalePriceCH" and "PriceDiff" are zero.

Question 1(d)

The classification rate was calculated based on the above GAM model.

```
gam.res <- predict(mod.gam, data[test,])>0
gam.TAB <- table(data$Purchase[test],as.numeric(gam.res))
gam.TAB
```

```
##
##         0   1
##    CH 195  29
##    MM  33 100
```

```
mkrgam<-1-sum(diag(gam.TAB))/sum(gam.TAB)
cat("The classification rate is: ", mkrgam)
```

```
## The classification rate is:  0.1736695
```

Question 1(e)

For the variable selection, due to the fact that there is no step.gam function provided by the library mgcv, a shrinkage smoother and more specifically, the thin plate regression spline smoother was used. The shrinkage smoother was used in the s() function by adding the term bs="ts" in each s(variable). Thus the new model is:

```
new.mod.gam <-gam(Purchase ~ s(PriceMM, k=3, bs="ts")+s(WeekofPurchase, bs="ts")
                  +s(PriceCH, bs="ts")+s(DiscCH, bs="ts")
              +s(DiscMM, bs="ts")+s(LoyalCH, bs="ts")
              +s(SalePriceMM, bs="ts")+s(SalePriceCH, bs="ts")
              +s(PriceDiff, bs="ts")
              +s(PctDiscMM, bs="ts")+s(PctDiscCH, bs="ts")
              +s(ListPriceDiff, bs="ts")+SpecialCH
              +SpecialMM+Store7
              ,data=data, family="binomial", subset=train)
summary(new.mod.gam)
```

```
##
## Family: binomial
## Link function: logit
##
## Formula:
## Purchase ~ s(PriceMM, k = 3, bs = "ts") + s(WeekofPurchase, bs = "ts") +
##     s(PriceCH, bs = "ts") + s(DiscCH, bs = "ts") + s(DiscMM,
##     bs = "ts") + s(LoyalCH, bs = "ts") + s(SalePriceMM, bs = "ts") +
##     s(SalePriceCH, bs = "ts") + s(PriceDiff, bs = "ts") + s(PctDiscMM,
##     bs = "ts") + s(PctDiscCH, bs = "ts") + s(ListPriceDiff, bs = "ts") +
##     SpecialCH + SpecialMM + Store7
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.5984     0.1628  -3.676 0.000237 ***
## SpecialCH1    0.3063     0.4075   0.752 0.452292
## SpecialMM1    0.5569     0.3521   1.582 0.113725
## Store7Yes    -0.8991     0.2797  -3.214 0.001309 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Approximate significance of smooth terms:
##                         edf Ref.df Chi.sq  p-value
## s(PriceMM)        1.441e+00      2  19.22 9.01e-06 ***
## s(WeekofPurchase) 6.452e-05      9   0.00 0.401815
## s(PriceCH)        1.438e-06      9   0.00 0.598266
## s(DiscCH)         3.700e-06      9   0.00 0.397252
## s(DiscMM)         1.343e-05      9   0.00 0.572503
## s(LoyalCH)        1.191e+00      9 173.29  < 2e-16 ***
## s(SalePriceMM)    2.861e-04      9   0.00 0.265470
## s(SalePriceCH)    1.047e+00      9  13.48 0.000134 ***
## s(PriceDiff)      2.892e-05      9   0.00 0.003907 **
## s(PctDiscMM)      4.537e+00      9  17.67 0.000967 ***
## s(PctDiscCH)      3.475e-06      9   0.00 0.367019
## s(ListPriceDiff)  3.568e-06      9   0.00 0.683180
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## R-sq.(adj) =  0.507   Deviance explained = 43.8%
## UBRE = -0.21064  Scale est. = 1         n = 713
```

Based on the inference table, the significant variables are: “PriceMM”, “LoyalCH”, “SalePriceCH”, “PriceDiff” and “PctDiscMM”. Thus, those variables, plus all the factor variables from the previous model, are going to be used to our new reduced model. The final reduced model is:

```
reduced.mod.gam <-gam(Purchase ~ s(PriceMM, k=3, bs="ts")+s(LoyalCH, bs="ts")+
                        s(SalePriceCH, bs="ts")+s(PriceDiff, bs="ts")
              +s(PctDiscMM, bs="ts")++SpecialCH
              +SpecialMM+Store7
              ,data=data, family="binomial", subset=train)
summary(reduced.mod.gam)
```

```
##
## Family: binomial
## Link function: logit
##
## Formula:
## Purchase ~ s(PriceMM, k = 3, bs = "ts") + s(LoyalCH, bs = "ts") +
##     s(SalePriceCH, bs = "ts") + s(PriceDiff, bs = "ts") + s(PctDiscMM,
##     bs = "ts") + +SpecialCH + SpecialMM + Store7
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.5984     0.1628  -3.676 0.000237 ***
## SpecialCH1    0.3062     0.4075   0.752 0.452344
## SpecialMM1    0.5568     0.3521   1.582 0.113729
## Store7Yes    -0.8991     0.2797  -3.214 0.001308 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Approximate significance of smooth terms:
##                      edf Ref.df Chi.sq  p-value
## s(PriceMM)     1.4407129      2  19.42 9.03e-06 ***
## s(LoyalCH)     1.1906793      9 173.29  < 2e-16 ***
## s(SalePriceCH) 1.0464586      9  13.46 0.000134 ***
## s(PriceDiff)   0.0001279      9   0.00 0.003890 **
## s(PctDiscMM)   4.5366134      9  17.96 0.000908 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## R-sq.(adj) =  0.507   Deviance explained = 43.8%
## UBRE = -0.21064  Scale est. = 1         n = 713
```

The classification rate was calculated based on the reduced GAM model.

```
gam.res <- predict(reduced.mod.gam, data[test,])>0
gam.TAB <- table(data$Purchase[test],as.numeric(gam.res))
gam.TAB
```

```
##
##         0   1
##   CH 194  30
##   MM  35  98
```

```
mkrgam<-1-sum(diag(gam.TAB))/sum(gam.TAB)
cat("The classification rate is: ", mkrgam)
```

```
## The classification rate is:  0.1820728
```

The new classification rate is slightly higher from the classification rate of the full model. Despite removing plenty of variable from the full model, the difference is not great. Thus, the final reduced model is quite convenient.