Modellierung des Weihnachtsgeschenkbudgets

Vincent Schmalor, Hendrick Kaiser, Lukas Küppers

3.1.2020

knitr::opts\_chunk$set(echo = TRUE)  
library(mosaic)

## Loading required package: dplyr

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

## Loading required package: lattice

## Loading required package: ggformula

## Loading required package: ggplot2

## Loading required package: ggstance

##   
## Attaching package: 'ggstance'

## The following objects are masked from 'package:ggplot2':  
##   
## geom\_errorbarh, GeomErrorbarh

##   
## New to ggformula? Try the tutorials:   
## learnr::run\_tutorial("introduction", package = "ggformula")  
## learnr::run\_tutorial("refining", package = "ggformula")

## Loading required package: mosaicData

## Loading required package: Matrix

## Registered S3 method overwritten by 'mosaic':  
## method from   
## fortify.SpatialPolygonsDataFrame ggplot2

##   
## The 'mosaic' package masks several functions from core packages in order to add   
## additional features. The original behavior of these functions should not be affected by this.  
##   
## Note: If you use the Matrix package, be sure to load it BEFORE loading mosaic.

##   
## Attaching package: 'mosaic'

## The following object is masked from 'package:Matrix':  
##   
## mean

## The following object is masked from 'package:ggplot2':  
##   
## stat

## The following objects are masked from 'package:dplyr':  
##   
## count, do, tally

## The following objects are masked from 'package:stats':  
##   
## binom.test, cor, cor.test, cov, fivenum, IQR, median,  
## prop.test, quantile, sd, t.test, var

## The following objects are masked from 'package:base':  
##   
## max, mean, min, prod, range, sample, sum

library(corrplot)

## corrplot 0.84 loaded

library(kableExtra)

##   
## Attaching package: 'kableExtra'

## The following object is masked from 'package:dplyr':  
##   
## group\_rows

library(reshape2)  
source("http://www.sthda.com/upload/rquery\_cormat.r")  
#Funktion zum Löschen aller Ausreißer, die nach IQR-1.5 Methode als Ausreißer gelten  
remove\_outliers <- function(x, na.rm = TRUE, ...) {  
 qnt <- quantile(x, probs=c(.25, .75), na.rm = na.rm, ...) # Vektor mit Quartilen 25% und 75% erstellen  
 H <- 1.3 \* IQR(x, na.rm = na.rm) #Antennenberechnung  
 y <- x  
 y[x < (qnt[1] - H)] <- NA #Untere Ausreißer entfernen  
 y[x > (qnt[2] + H)] <- NA #Obere Ausreißer  
 y  
}  
  
#Funktion für die Darstellung der Favstats für jede Spalte  
multi.fun <- function(x) {   
 c(min = min(x), mean = mean(x), max = max(x), sd = sd(x), q1 = quantile(x, 0.25), med = median(x), q3 = quantile(x,0.75))  
}  
raw.training <- read.csv2("sources/Trainingsdaten.csv")

# Einleitung

Dieses Dokument stellt sowohl die Durchführung, als auch die Protokollierung des Vorhersagewettbewerbs dar. Die Aufgabenstellung besteht darin, ein Modell aus einem Grunddatensatz zu entwickeln und dieses Modell auf einen Anwendungsdatensatz anzuwenden. Dabei wird aus mehreren Einflussvariablen eine abhängige Variable geschätzt. Der konkrete Anwendungsfall ist ein Auszug aus einem Face-to-Face Interview über das Konsum- und Schenkverhalten zu Weihnachten. Das Grunddatenset beinhaltet 400 Beobachtungen mit 17 erklärenden Variablen und die abhängige Variable des geschätzten Budgets, die es im Anwendungsdatenset zu modellieren gilt. In dem folgenden Kapitel wird diese Aufgabe bearbeitet, indem zunächst ein Überblick über die Daten in einer explorativen Datenanalyse gegeben wird. Anhand verschiedener Analysen und der Untersuchung der semantischen Zusammenhänge wird dann eine Vorgehensweise erarbeitet und beschrieben. Das Ergebnis ist eine csv-Datei, die das Anwendungsdatenset mit der modellierten, abhängigen Variable beinhaltet. So wird eine Prognose über diese Variable für jede Beobachtung gegeben, für die das tatsächliche Budget nicht vorliegt. Eine Zusammenfassung der Ergebnisse findet sich auch auf dem Plakat der genannten Autoren.

# Explorative Datenanalyse

## Rohdatenanalyse und -strukturierung

Kurzer Überblick über die vorhandenen Daten in ihrer Rohform.

#Visualisierung der Rohdaten in schönen Tabellen  
head(raw.training, n= 10)

## X7.1 X7.2 X7.3 X7.4 X7.5 X7.6 X7.7 X9.1 X9.2 X9.3 X9.4 X9.5 X9.6 X9.7  
## 1 5 5 5 1 5 3 2 1 1 1 0 0 0 0  
## 2 3 1 1 1 1 3 1 0 1 1 0 0 1 0  
## 3 1 7 7 1 1 1 1 1 0 0 0 0 1 0  
## 4 5 5 5 1 1 5 2 0 1 1 0 1 0 0  
## 5 6 7 7 3 2 4 5 1 1 0 0 0 0 0  
## 6 5 1 5 4 6 6 4 1 1 1 0 0 1 0  
## 7 7 7 7 7 5 6 7 1 1 1 0 1 0 0  
## 8 7 1 7 7 2 4 7 1 0 0 0 1 1 0  
## 9 4 7 2 3 1 4 7 0 0 1 0 1 0 0  
## 10 1 1 7 1 1 6 6 1 1 1 0 0 1 0  
## X9.8 D1 D2 X10.1  
## 1 0 27 1 300  
## 2 0 18 2 150  
## 3 0 25 1 50  
## 4 0 36 2 50  
## 5 0 30 1 100  
## 6 0 51 1 1000  
## 7 0 56 2 700  
## 8 1 57 2 600  
## 9 0 70 1 500  
## 10 0 38 2 400

apply(raw.training, 2, multi.fun)

## X7.1 X7.2 X7.3 X7.4 X7.5 X7.6 X7.7  
## min 1.000000 1.000000 1.000000 1.000000 1.000000 1.000000 1.000000  
## mean 5.145000 4.515000 4.680000 3.585000 3.635000 4.417500 3.862500  
## max 7.000000 7.000000 7.000000 7.000000 7.000000 7.000000 7.000000  
## sd 1.872897 1.953026 2.099672 2.039159 1.915973 1.859257 2.010896  
## q1.25% 4.000000 3.000000 3.000000 2.000000 2.000000 3.000000 2.000000  
## med 6.000000 5.000000 5.000000 4.000000 4.000000 5.000000 4.000000  
## q3.75% 7.000000 6.000000 7.000000 5.000000 5.000000 6.000000 6.000000  
## X9.1 X9.2 X9.3 X9.4 X9.5 X9.6  
## min 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000  
## mean 0.7425000 0.6250000 0.6300000 0.0200000 0.5800000 0.4675000  
## max 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000  
## sd 0.4378047 0.4847292 0.4834089 0.1401753 0.4941766 0.4995675  
## q1.25% 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000  
## med 1.0000000 1.0000000 1.0000000 0.0000000 1.0000000 0.0000000  
## q3.75% 1.0000000 1.0000000 1.0000000 0.0000000 1.0000000 1.0000000  
## X9.7 X9.8 D1 D2 X10.1  
## min 0.0000000 0.0000000 18.00000 1.0000000 1.0000  
## mean 0.1175000 0.0625000 47.45750 1.4825000 486.6900  
## max 1.0000000 1.0000000 88.00000 2.0000000 3500.0000  
## sd 0.3224184 0.2423646 18.67901 0.5003194 427.1325  
## q1.25% 0.0000000 0.0000000 31.00000 1.0000000 200.0000  
## med 0.0000000 0.0000000 47.50000 1.0000000 400.0000  
## q3.75% 0.0000000 0.0000000 61.00000 2.0000000 600.0000

Kurzer Überblick über die Daten mit richtigen Spaltenbeschriftungen, um sie interpretieren zu können

#Auswahl der Spalten, die als Faktor interpretiert werden sollen. Nicht ausgeführt, um Regressionsanalysen durchführen zu können  
# cols <- c("X7.1","X7.2","X7.3","X7.4","X7.5","X7.6","X7.7","X9.1","X9.2","X9.3","X9.4","X9.5","X9.6","X9.7","X9.8","D2")   
#raw.training[cols] <- lapply(raw.training[cols], factor) #Faktorspalten setzen  
#Spalten mit sprechenden Überschriften versehen  
colnames(raw.training) <- c("Beratung", "Angebote","Bequemlichkeit","Einkaufsatmosphaere","Marken","GPM","Naehe","Partner","Eltern","Verwandte","Kommilitonen","Kinder","Freunde","Arbeitskollegen","Soziale.Institutionen","Alter","Geschlecht","Budget")   
head(raw.training, n= 10) #Dataset erneut anschauen

## Beratung Angebote Bequemlichkeit Einkaufsatmosphaere Marken GPM Naehe  
## 1 5 5 5 1 5 3 2  
## 2 3 1 1 1 1 3 1  
## 3 1 7 7 1 1 1 1  
## 4 5 5 5 1 1 5 2  
## 5 6 7 7 3 2 4 5  
## 6 5 1 5 4 6 6 4  
## 7 7 7 7 7 5 6 7  
## 8 7 1 7 7 2 4 7  
## 9 4 7 2 3 1 4 7  
## 10 1 1 7 1 1 6 6  
## Partner Eltern Verwandte Kommilitonen Kinder Freunde Arbeitskollegen  
## 1 1 1 1 0 0 0 0  
## 2 0 1 1 0 0 1 0  
## 3 1 0 0 0 0 1 0  
## 4 0 1 1 0 1 0 0  
## 5 1 1 0 0 0 0 0  
## 6 1 1 1 0 0 1 0  
## 7 1 1 1 0 1 0 0  
## 8 1 0 0 0 1 1 0  
## 9 0 0 1 0 1 0 0  
## 10 1 1 1 0 0 1 0  
## Soziale.Institutionen Alter Geschlecht Budget  
## 1 0 27 1 300  
## 2 0 18 2 150  
## 3 0 25 1 50  
## 4 0 36 2 50  
## 5 0 30 1 100  
## 6 0 51 1 1000  
## 7 0 56 2 700  
## 8 1 57 2 600  
## 9 0 70 1 500  
## 10 0 38 2 400

apply(raw.training, 2, multi.fun)

## Beratung Angebote Bequemlichkeit Einkaufsatmosphaere Marken  
## min 1.000000 1.000000 1.000000 1.000000 1.000000  
## mean 5.145000 4.515000 4.680000 3.585000 3.635000  
## max 7.000000 7.000000 7.000000 7.000000 7.000000  
## sd 1.872897 1.953026 2.099672 2.039159 1.915973  
## q1.25% 4.000000 3.000000 3.000000 2.000000 2.000000  
## med 6.000000 5.000000 5.000000 4.000000 4.000000  
## q3.75% 7.000000 6.000000 7.000000 5.000000 5.000000  
## GPM Naehe Partner Eltern Verwandte Kommilitonen  
## min 1.000000 1.000000 0.0000000 0.0000000 0.0000000 0.0000000  
## mean 4.417500 3.862500 0.7425000 0.6250000 0.6300000 0.0200000  
## max 7.000000 7.000000 1.0000000 1.0000000 1.0000000 1.0000000  
## sd 1.859257 2.010896 0.4378047 0.4847292 0.4834089 0.1401753  
## q1.25% 3.000000 2.000000 0.0000000 0.0000000 0.0000000 0.0000000  
## med 5.000000 4.000000 1.0000000 1.0000000 1.0000000 0.0000000  
## q3.75% 6.000000 6.000000 1.0000000 1.0000000 1.0000000 0.0000000  
## Kinder Freunde Arbeitskollegen Soziale.Institutionen Alter  
## min 0.0000000 0.0000000 0.0000000 0.0000000 18.00000  
## mean 0.5800000 0.4675000 0.1175000 0.0625000 47.45750  
## max 1.0000000 1.0000000 1.0000000 1.0000000 88.00000  
## sd 0.4941766 0.4995675 0.3224184 0.2423646 18.67901  
## q1.25% 0.0000000 0.0000000 0.0000000 0.0000000 31.00000  
## med 1.0000000 0.0000000 0.0000000 0.0000000 47.50000  
## q3.75% 1.0000000 1.0000000 0.0000000 0.0000000 61.00000  
## Geschlecht Budget  
## min 1.0000000 1.0000  
## mean 1.4825000 486.6900  
## max 2.0000000 3500.0000  
## sd 0.5003194 427.1325  
## q1.25% 1.0000000 200.0000  
## med 1.0000000 400.0000  
## q3.75% 2.0000000 600.0000

## Erweiterung um abgeleitete Metriken

Fügt die folgenden Metriken hinzu: - Anzahl der bedachten Gruppen

#Anzahl der bedachten Personen als zusätzliche Spalte hinzufügen  
raw.training <- raw.training %>% mutate(gift.count = (Partner\*1) + (Eltern\*1) +(Verwandte\*1) + (Kommilitonen\*1) + (Kinder\*1) + (Freunde\*1) + (Arbeitskollegen\*1) + (Soziale.Institutionen\*1))

#Test mit gewichteten Gruppen war nicht erfolgreich  
#raw.training <- raw.training %>% mutate(gift.count = (X9.1\*2.5) + (X9.2\*2) + (X9.3\*2) + (X9.4\*0.5) + (X9.5\*1) + (X9.6\*1.5) + (X9.7\*0.75) + (X9.8\*0.5))

## Bereinigung der Datenmenge von Ausreißern

Ausreißer identifizieren und bereinigtes Dataset in “training” speichern.

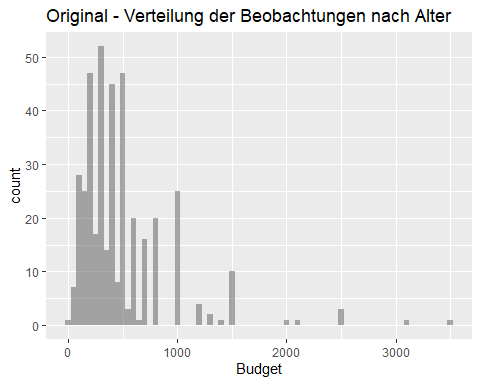
col.Budget <- pull(raw.training, Budget) #Spalte extrahieren, um Ausreißer zu identifizieren  
training <- raw.training %>% mutate("cleaned.Budget" = remove\_outliers(col.Budget)) #Ausreißer nullen und in neues Datenset speichern  
training <- training %>% filter(!is.na(cleaned.Budget)) #Datenset um Ausreißer bereinigen  
training <- subset(training, select = -(cleaned.Budget))  
head(training, n= 10)

## Beratung Angebote Bequemlichkeit Einkaufsatmosphaere Marken GPM Naehe  
## 1 5 5 5 1 5 3 2  
## 2 3 1 1 1 1 3 1  
## 3 1 7 7 1 1 1 1  
## 4 5 5 5 1 1 5 2  
## 5 6 7 7 3 2 4 5  
## 6 5 1 5 4 6 6 4  
## 7 7 7 7 7 5 6 7  
## 8 7 1 7 7 2 4 7  
## 9 4 7 2 3 1 4 7  
## 10 1 1 7 1 1 6 6  
## Partner Eltern Verwandte Kommilitonen Kinder Freunde Arbeitskollegen  
## 1 1 1 1 0 0 0 0  
## 2 0 1 1 0 0 1 0  
## 3 1 0 0 0 0 1 0  
## 4 0 1 1 0 1 0 0  
## 5 1 1 0 0 0 0 0  
## 6 1 1 1 0 0 1 0  
## 7 1 1 1 0 1 0 0  
## 8 1 0 0 0 1 1 0  
## 9 0 0 1 0 1 0 0  
## 10 1 1 1 0 0 1 0  
## Soziale.Institutionen Alter Geschlecht Budget gift.count  
## 1 0 27 1 300 3  
## 2 0 18 2 150 3  
## 3 0 25 1 50 2  
## 4 0 36 2 50 3  
## 5 0 30 1 100 2  
## 6 0 51 1 1000 4  
## 7 0 56 2 700 4  
## 8 1 57 2 600 4  
## 9 0 70 1 500 2  
## 10 0 38 2 400 4

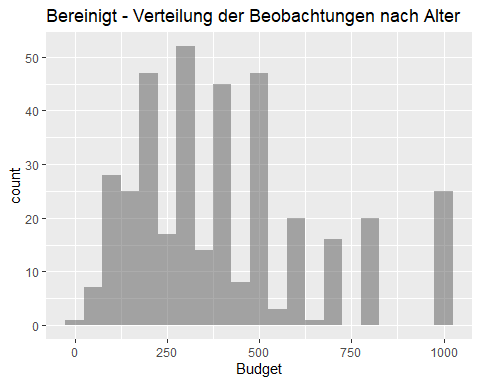
## Visualisierungen der Datenmenge

### Histogramme

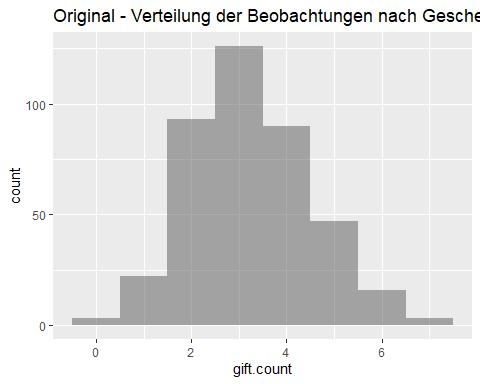
gf\_histogram(~ Budget, data = raw.training, binwidth = 50, center = 50, title = "Original - Verteilung der Beobachtungen nach Alter")



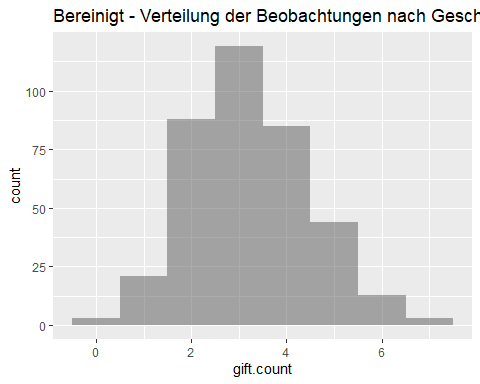
gf\_histogram(~ Budget, data = training, binwidth = 50, center = 50, title = "Bereinigt - Verteilung der Beobachtungen nach Alter")



gf\_histogram(~ gift.count, data = raw.training, binwidth = 1, center = 1, title = "Original - Verteilung der Beobachtungen nach Geschenkanzahl")

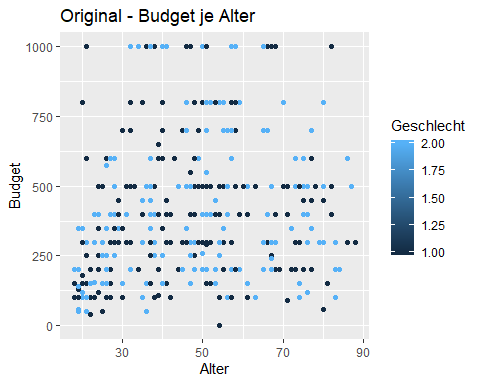


gf\_histogram(~ gift.count, data = training, binwidth = 1, center = 1, title = "Bereinigt - Verteilung der Beobachtungen nach Geschenkanzahl")

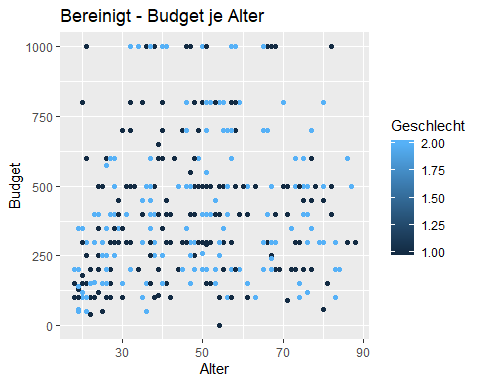


### Punktwolken

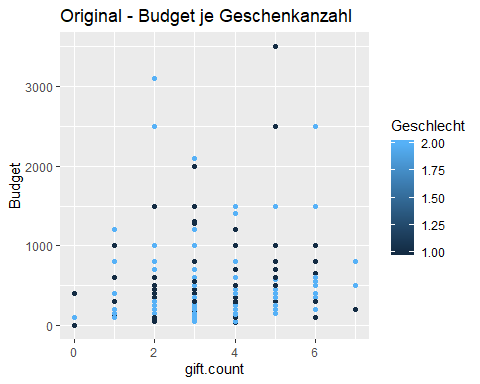
gf\_point(Budget ~ Alter, data= training, colour = ~ Geschlecht, title = "Original - Budget je Alter")



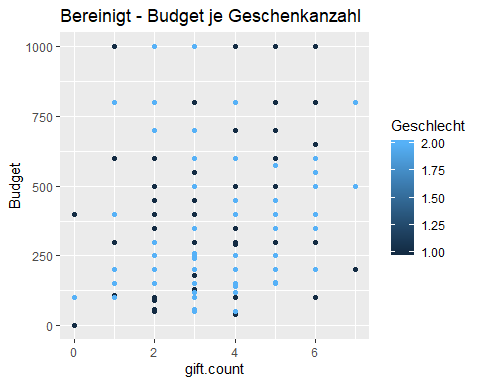
gf\_point(Budget ~ Alter, data= training, colour = ~ Geschlecht , title = "Bereinigt - Budget je Alter")



gf\_point(Budget ~ gift.count, data = raw.training, colour = ~ Geschlecht, title = "Original - Budget je Geschenkanzahl")



gf\_point(Budget ~ gift.count, data = training, colour = ~ Geschlecht, title = "Bereinigt - Budget je Geschenkanzahl")

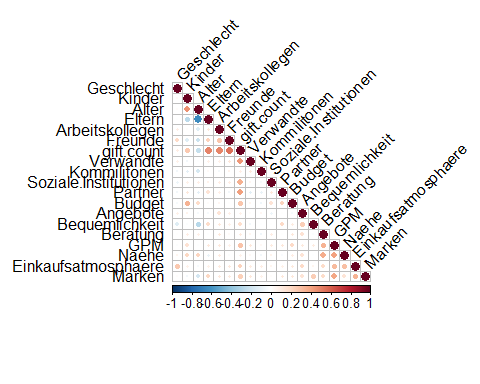


## Lineare Modelle

### Übersicht

Korrelationsmatrix erstellen, um Abhängigkeiten zu identifizieren. Hier ist zu erkennen, dass das Budget kaum Abhängigkeiten aufweist.

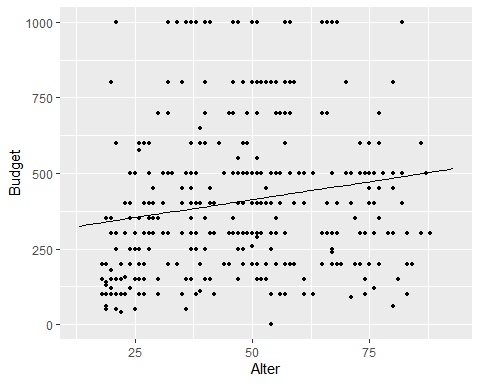
#Teil eines extrapakets aber sehr praktisch  
invisible(rquery.cormat(training))



### Detailanalyse

Lineare Modelle erstellen, um statistisch relevante Einflussvariablen zu identifizeren.

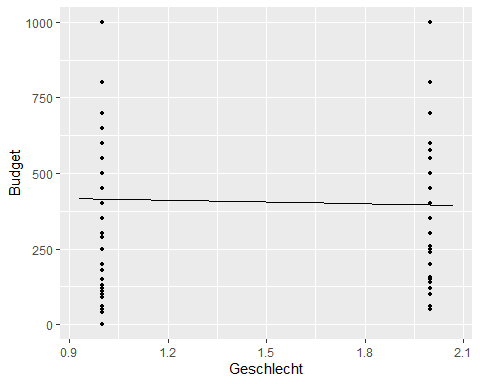
#Demografische Variablen  
lm.budget.alter <- lm(Budget ~ Alter, data = training)  
plotModel(lm.budget.alter, title = "Weihnachtsbudget in Relation zur Variable Alter")



summary(lm.budget.alter)

##   
## Call:  
## lm(formula = Budget ~ Alter, data = training)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -424.81 -188.63 -47.03 115.55 655.34   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 294.7738 34.3637 8.578 2.6e-16 \*\*\*  
## Alter 2.3755 0.6824 3.481 0.000559 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 243.8 on 374 degrees of freedom  
## Multiple R-squared: 0.03138, Adjusted R-squared: 0.02879   
## F-statistic: 12.12 on 1 and 374 DF, p-value: 0.0005589

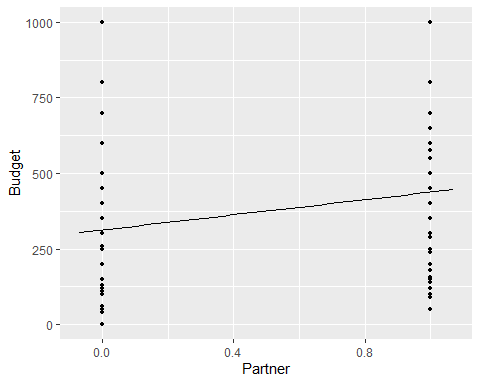
lm.budget.Geschlecht <- lm(Budget ~ Geschlecht, data = training)  
plotModel(lm.budget.Geschlecht, title = "Weihnachtsbudget in Relation zur Variable Geschlecht")



summary(lm.budget.Geschlecht)

##   
## Call:  
## lm(formula = Budget ~ Geschlecht, data = training)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -414.41 -195.94 -45.94 104.06 604.06   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 434.88 39.89 10.903 <2e-16 \*\*\*  
## Geschlecht -19.47 25.56 -0.762 0.447   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 247.5 on 374 degrees of freedom  
## Multiple R-squared: 0.001549, Adjusted R-squared: -0.00112   
## F-statistic: 0.5804 on 1 and 374 DF, p-value: 0.4466

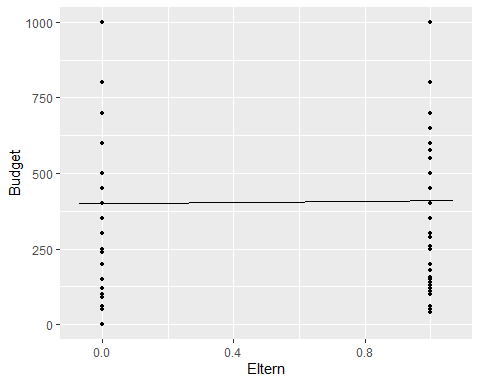
#Wer wird beschenkt?  
lm.budget.Partner <- lm(Budget ~ Partner, data = training)  
plotModel(lm.budget.Partner, title = "Weihnachtsbudget in Relation zur Variable Partner")



summary(lm.budget.Partner)

##   
## Call:  
## lm(formula = Budget ~ Partner, data = training)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -388.49 -188.49 -38.49 136.65 687.10   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 312.90 24.52 12.759 < 2e-16 \*\*\*  
## Partner 125.60 28.47 4.412 1.34e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 241.5 on 374 degrees of freedom  
## Multiple R-squared: 0.04947, Adjusted R-squared: 0.04692   
## F-statistic: 19.46 on 1 and 374 DF, p-value: 1.343e-05

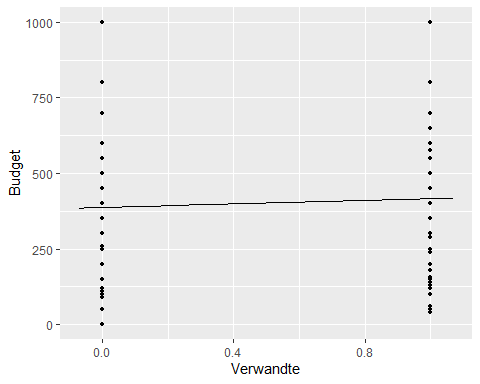
lm.budget.Eltern <- lm(Budget ~ Eltern, data = training)  
plotModel(lm.budget.Eltern, title = "Weihnachtsbudget in Relation zur Variable Eltern")



summary(lm.budget.Eltern)

##   
## Call:  
## lm(formula = Budget ~ Eltern, data = training)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -399.08 -200.08 -59.62 99.92 599.92   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 400.079 21.009 19.043 <2e-16 \*\*\*  
## Eltern 9.541 26.462 0.361 0.719   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 247.7 on 374 degrees of freedom  
## Multiple R-squared: 0.0003475, Adjusted R-squared: -0.002325   
## F-statistic: 0.13 on 1 and 374 DF, p-value: 0.7186

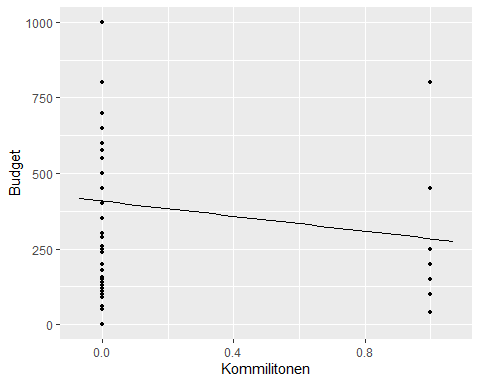
lm.budget.Verwandte <- lm(Budget ~ Verwandte, data = training)  
plotModel(lm.budget.Verwandte, title = "Weihnachtsbudget in Relation zur Variable Verwandte")



summary(lm.budget.Verwandte)

##   
## Call:  
## lm(formula = Budget ~ Verwandte, data = training)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -386.72 -187.72 -37.72 112.28 612.28   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 387.72 20.90 18.549 <2e-16 \*\*\*  
## Verwandte 29.27 26.38 1.109 0.268   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 247.3 on 374 degrees of freedom  
## Multiple R-squared: 0.00328, Adjusted R-squared: 0.0006148   
## F-statistic: 1.231 on 1 and 374 DF, p-value: 0.268

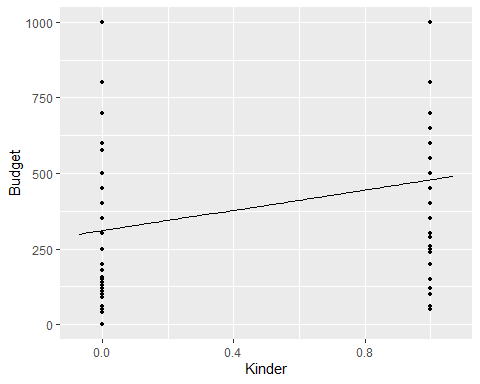
lm.budget.Kommilitonen <- lm(Budget ~ Kommilitonen, data = training)  
plotModel(lm.budget.Kommilitonen, title = "Weihnachtsbudget in Relation zur Variable Kommilitonen")



summary(lm.budget.Kommilitonen)

##   
## Call:  
## lm(formula = Budget ~ Kommilitonen, data = training)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -407.4 -208.4 -58.4 91.6 591.6   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 408.40 12.87 31.741 <2e-16 \*\*\*  
## Kommilitonen -124.12 94.30 -1.316 0.189   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 247.2 on 374 degrees of freedom  
## Multiple R-squared: 0.004611, Adjusted R-squared: 0.001949   
## F-statistic: 1.732 on 1 and 374 DF, p-value: 0.1889

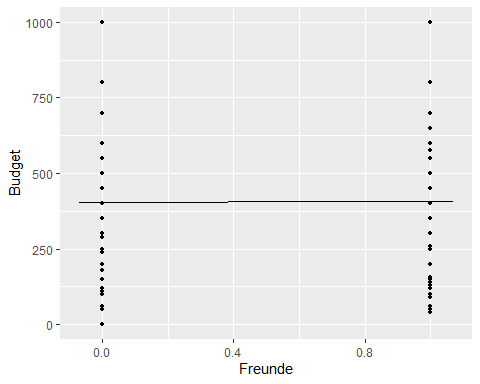
lm.budget.Kinder <- lm(Budget ~ Kinder, data = training)  
plotModel(lm.budget.Kinder, title = "Weihnachtsbudget in Relation zur Variable Kinder")



summary(lm.budget.Kinder)

##   
## Call:  
## lm(formula = Budget ~ Kinder, data = training)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -429.20 -179.20 -60.56 120.80 689.44   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 310.56 18.26 17.007 < 2e-16 \*\*\*  
## Kinder 168.64 24.26 6.951 1.62e-11 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 233.1 on 374 degrees of freedom  
## Multiple R-squared: 0.1144, Adjusted R-squared: 0.112   
## F-statistic: 48.32 on 1 and 374 DF, p-value: 1.621e-11

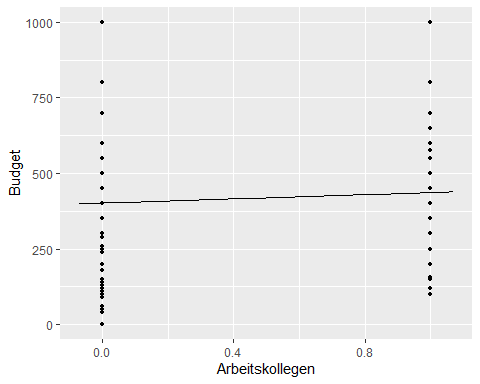
lm.budget.Freunde <- lm(Budget ~ Freunde, data = training)  
plotModel(lm.budget.Freunde, title = "Weihnachtsbudget in Relation zur Variable Freunde")



summary(lm.budget.Freunde)

##   
## Call:  
## lm(formula = Budget ~ Freunde, data = training)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -403.5 -204.5 -54.5 95.5 595.5   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 404.505 17.517 23.092 <2e-16 \*\*\*  
## Freunde 3.393 25.603 0.133 0.895   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 247.7 on 374 degrees of freedom  
## Multiple R-squared: 4.695e-05, Adjusted R-squared: -0.002627   
## F-statistic: 0.01756 on 1 and 374 DF, p-value: 0.8947

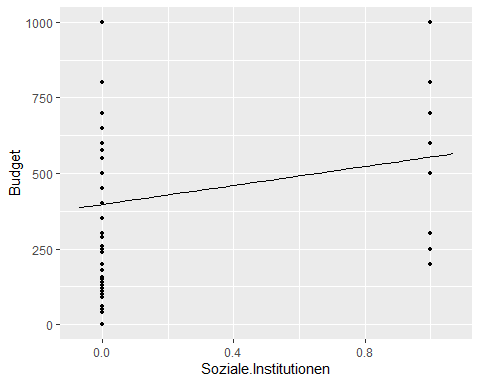
lm.budget.Arbeitskollegen <- lm(Budget ~ Arbeitskollegen, data = training)  
plotModel(lm.budget.Arbeitskollegen, title = "Weihnachtsbudget in Relation zur Variable Arbeitskollegen")



summary(lm.budget.Arbeitskollegen)

##   
## Call:  
## lm(formula = Budget ~ Arbeitskollegen, data = training)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -401.02 -202.02 -52.02 97.98 597.98   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 402.02 13.58 29.599 <2e-16 \*\*\*  
## Arbeitskollegen 34.80 39.70 0.876 0.381   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 247.5 on 374 degrees of freedom  
## Multiple R-squared: 0.002049, Adjusted R-squared: -0.0006188   
## F-statistic: 0.7681 on 1 and 374 DF, p-value: 0.3814

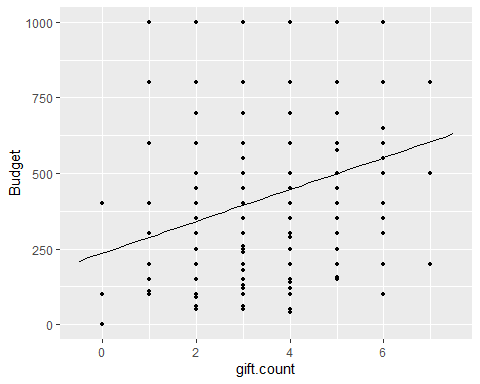
lm.budget.Soziale.Institutionen <- lm(Budget ~ Soziale.Institutionen, data = training)  
plotModel(lm.budget.Soziale.Institutionen, title = "Weihnachtsbudget in Relation zur Variable Soziale.Institutionen")



summary(lm.budget.Soziale.Institutionen)

##   
## Call:  
## lm(formula = Budget ~ Soziale.Institutionen, data = training)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -396.3 -197.3 -47.3 102.7 602.7   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 397.30 13.01 30.545 < 2e-16 \*\*\*  
## Soziale.Institutionen 157.46 55.04 2.861 0.00446 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 245.1 on 374 degrees of freedom  
## Multiple R-squared: 0.02142, Adjusted R-squared: 0.0188   
## F-statistic: 8.186 on 1 and 374 DF, p-value: 0.00446

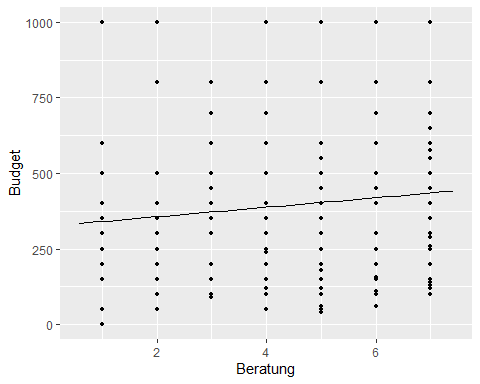
lm.budget.gift.count <- lm(Budget ~ gift.count, data = training)  
plotModel(lm.budget.gift.count, title = "Weihnachtsbudget in Relation zur Variable gift.count")



summary(lm.budget.gift.count)

##   
## Call:  
## lm(formula = Budget ~ gift.count, data = training)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -452.44 -147.61 -44.17 105.83 711.35   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 235.897 33.588 7.023 1.03e-11 \*\*\*  
## gift.count 52.757 9.689 5.445 9.39e-08 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 238.5 on 374 degrees of freedom  
## Multiple R-squared: 0.07346, Adjusted R-squared: 0.07098   
## F-statistic: 29.65 on 1 and 374 DF, p-value: 9.39e-08

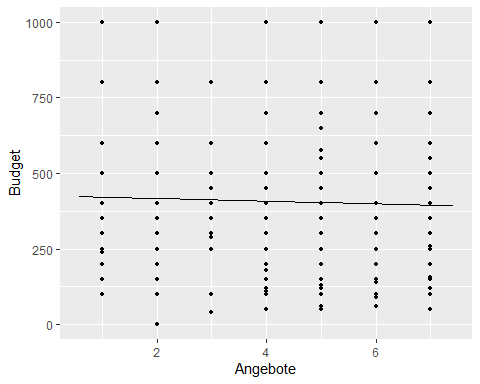
#Was ist wichtig bei der Auswahl?  
lm.budget.Beratung <- lm(Budget ~ Beratung, data = training)  
plotModel(lm.budget.Beratung, title = "Weihnachtsbudget in Relation zur Variable Beratung")



summary(lm.budget.Beratung)

##   
## Call:  
## lm(formula = Budget ~ Beratung, data = training)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -363.99 -186.22 -36.89 111.79 659.14   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 325.074 36.974 8.792 <2e-16 \*\*\*  
## Beratung 15.784 6.766 2.333 0.0202 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 246 on 374 degrees of freedom  
## Multiple R-squared: 0.01434, Adjusted R-squared: 0.01171   
## F-statistic: 5.442 on 1 and 374 DF, p-value: 0.02019

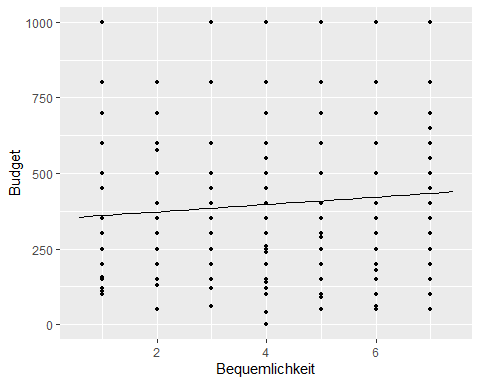
lm.budget.Angebote <- lm(Budget ~ Angebote, data = training)  
plotModel(lm.budget.Angebote, title = "Weihnachtsbudget in Relation zur Variable Angebote")



summary(lm.budget.Angebote)

##   
## Call:  
## lm(formula = Budget ~ Angebote, data = training)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -416.41 -199.49 -49.49 104.99 604.99   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 426.376 32.310 13.196 <2e-16 \*\*\*  
## Angebote -4.481 6.557 -0.683 0.495   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 247.6 on 374 degrees of freedom  
## Multiple R-squared: 0.001247, Adjusted R-squared: -0.001423   
## F-statistic: 0.467 on 1 and 374 DF, p-value: 0.4948

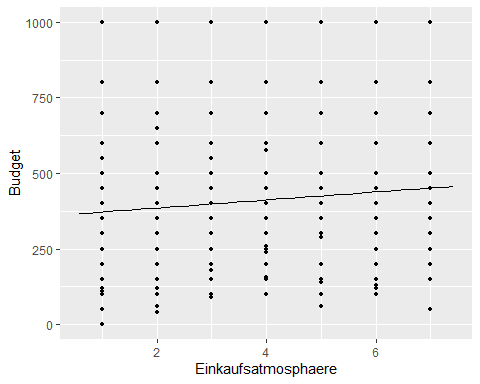
lm.budget.Bequemlichkeit <- lm(Budget ~ Bequemlichkeit, data = training)  
plotModel(lm.budget.Bequemlichkeit, title = "Weihnachtsbudget in Relation zur Variable Bequemlichkeit")



summary(lm.budget.Bequemlichkeit)

##   
## Call:  
## lm(formula = Budget ~ Bequemlichkeit, data = training)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -396.30 -197.30 -34.54 127.17 639.41   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 348.347 31.264 11.142 <2e-16 \*\*\*  
## Bequemlichkeit 12.239 6.055 2.022 0.0439 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 246.4 on 374 degrees of freedom  
## Multiple R-squared: 0.01081, Adjusted R-squared: 0.008164   
## F-statistic: 4.087 on 1 and 374 DF, p-value: 0.04394

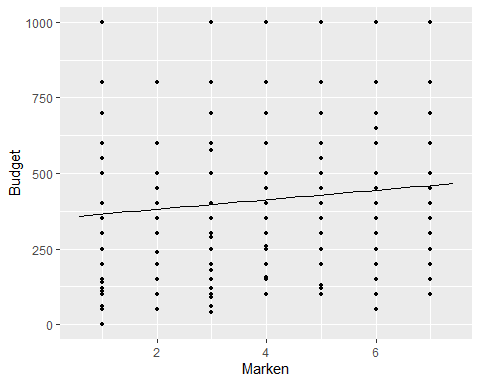
lm.budget.Einkaufsatmosphaere <- lm(Budget ~ Einkaufsatmosphaere, data = training)  
plotModel(lm.budget.Einkaufsatmosphaere, title = "Weihnachtsbudget in Relation zur Variable Einkaufsatmosphaere")



summary(lm.budget.Einkaufsatmosphaere)

##   
## Call:  
## lm(formula = Budget ~ Einkaufsatmosphaere, data = training)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -402.03 -185.85 -52.03 114.15 627.39   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 359.378 25.445 14.123 <2e-16 \*\*\*  
## Einkaufsatmosphaere 13.237 6.248 2.119 0.0348 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 246.3 on 374 degrees of freedom  
## Multiple R-squared: 0.01186, Adjusted R-squared: 0.009218   
## F-statistic: 4.489 on 1 and 374 DF, p-value: 0.03478

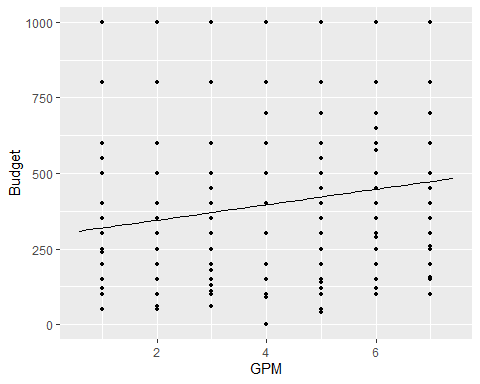
lm.budget.Marken <- lm(Budget ~ Marken, data = training)  
plotModel(lm.budget.Marken, title = "Weihnachtsbudget in Relation zur Variable Marken")



summary(lm.budget.Marken)

##   
## Call:  
## lm(formula = Budget ~ Marken, data = training)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -394.46 -195.20 -53.78 133.94 633.94   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 350.378 26.915 13.018 <2e-16 \*\*\*  
## Marken 15.680 6.681 2.347 0.0195 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 245.9 on 374 degrees of freedom  
## Multiple R-squared: 0.01451, Adjusted R-squared: 0.01188   
## F-statistic: 5.508 on 1 and 374 DF, p-value: 0.01945

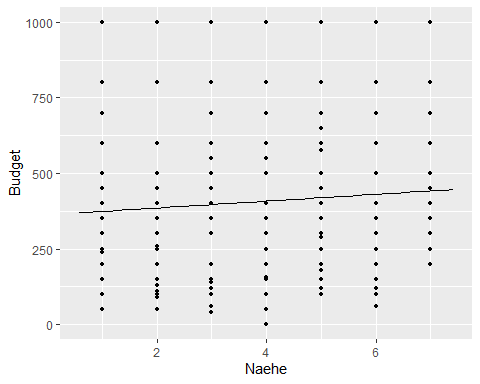
lm.budget.GPM <- lm(Budget ~ GPM, data = training)  
plotModel(lm.budget.GPM, title = "Weihnachtsbudget in Relation zur Variable GPM")



summary(lm.budget.GPM)

##   
## Call:  
## lm(formula = Budget ~ GPM, data = training)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -395.54 -177.85 -47.85 109.22 680.41   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 293.936 32.232 9.119 < 2e-16 \*\*\*  
## GPM 25.652 6.791 3.777 0.000184 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 243.1 on 374 degrees of freedom  
## Multiple R-squared: 0.03675, Adjusted R-squared: 0.03417   
## F-statistic: 14.27 on 1 and 374 DF, p-value: 0.0001844

lm.budget.Naehe <- lm(Budget ~ Naehe, data = training)  
plotModel(lm.budget.Naehe, title = "Weihnachtsbudget in Relation zur Variable Naehe")



summary(lm.budget.Naehe)

##   
## Call:  
## lm(formula = Budget ~ Naehe, data = training)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -407.23 -185.61 -42.16 114.39 625.70   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 362.988 27.484 13.207 <2e-16 \*\*\*  
## Naehe 11.310 6.392 1.769 0.0776 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 246.7 on 374 degrees of freedom  
## Multiple R-squared: 0.008302, Adjusted R-squared: 0.00565   
## F-statistic: 3.131 on 1 and 374 DF, p-value: 0.07764

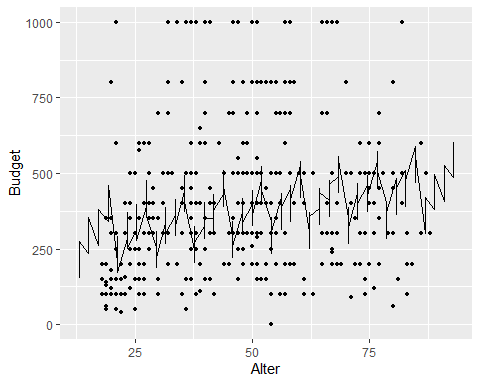
# Methodik

Wir haben uns die Daten angeguckt und sind zu dem Schluss gekommen, dass die Verwendung eines generalisierten linearen Modells die beste Verfahrensweise ist, alle Einflussfaktoren in ein Modell zu verpacken.

# Modellierung

Das Modell wird mit den in R vorgegebenen Funktionen erstellt

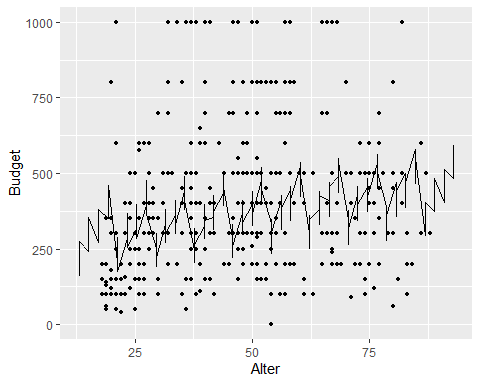
#General Linear Model mit allen Variablen, die laut voriger Untersuchung einen signifikanten Einfluss auf das Budget haben.  
glm.drei.Sterne <- glm(Budget ~ Alter + Partner + Kinder + gift.count + GPM, data = training)  
plotModel(glm.drei.Sterne, title = "Weihnachtsbudget in Relation zu allen Variablen mit starker statistischer Relevanz in der Relation")



summary(glm.drei.Sterne)

##   
## Call:  
## glm(formula = Budget ~ Alter + Partner + Kinder + gift.count +   
## GPM, data = training)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -472.2 -151.3 -39.1 103.9 710.7   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 39.4917 56.1797 0.703 0.482527   
## Alter 1.9415 0.7737 2.509 0.012526 \*   
## Partner 75.3113 29.0747 2.590 0.009969 \*\*   
## Kinder 97.8603 29.0230 3.372 0.000826 \*\*\*  
## gift.count 33.3773 11.3746 2.934 0.003551 \*\*   
## GPM 12.9506 6.5699 1.971 0.049447 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for gaussian family taken to be 50452.32)  
##   
## Null deviance: 22953292 on 375 degrees of freedom  
## Residual deviance: 18667359 on 370 degrees of freedom  
## AIC: 5146.6  
##   
## Number of Fisher Scoring iterations: 2

#General Linear Model mit allen Variablen, die laut voriger Untersuchung einen einigermaßen signifikanten Einfluss auf das Budget haben.  
glm.zwei.Sterne <- glm(Budget ~ Alter + Partner + Kinder + gift.count + GPM + Soziale.Institutionen, data = training)  
plotModel(glm.zwei.Sterne, title = "Weihnachtsbudget in Relation zu allen Variablen mit mindestens mittlerer statistischer Relevanz in der Relation")



summary(glm.zwei.Sterne)

##   
## Call:  
## glm(formula = Budget ~ Alter + Partner + Kinder + gift.count +   
## GPM + Soziale.Institutionen, data = training)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -469.12 -149.94 -36.01 102.69 709.61   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 53.7493 58.9985 0.911 0.362875   
## Alter 1.7911 0.7969 2.248 0.025192 \*   
## Partner 75.8303 29.0965 2.606 0.009527 \*\*   
## Kinder 101.3482 29.3669 3.451 0.000623 \*\*\*  
## gift.count 29.6496 12.3081 2.409 0.016488 \*   
## GPM 12.9485 6.5732 1.970 0.049597 \*   
## Soziale.Institutionen 44.1137 55.4798 0.795 0.427048   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for gaussian family taken to be 50502.52)  
##   
## Null deviance: 22953292 on 375 degrees of freedom  
## Residual deviance: 18635430 on 369 degrees of freedom  
## AIC: 5148  
##   
## Number of Fisher Scoring iterations: 2

Anhand des erstellten Modells schätzen wir die Daten für das Anwendungs-Data-Set

#Zieldatenset laden und analog zum Trainingsdatenset transformieren  
raw.anwendung <- read.csv2("sources/Anwendungsdaten.csv")  
raw.anwendung <- raw.anwendung %>% mutate(gift.count = (X9.1\*1) + (X9.2\*1) +(X9.3\*1) + (X9.4\*1) + (X9.5\*1) + (X9.6\*1) + (X9.7\*1) + (X9.8\*1))  
colnames(raw.anwendung) <- c("Beratung", "Angebote","Bequemlichkeit","Einkaufsatmosphaere","Marken","GPM","Naehe","Partner","Eltern","Verwandte","Kommilitonen","Kinder","Freunde","Arbeitskollegen","Soziale.Institutionen","Alter","Geschlecht","gift.count") #Spaltennamen setzen  
ergglm <- predict.glm(glm.drei.Sterne, newdata = raw.anwendung, interval="prediction")  
anwendung <- raw.anwendung %>% mutate(Budget = ergglm)  
write.csv2(anwendung, "Prognose\_Vincent\_Schmalor\_Hendrick\_Kaiser\_Lukas\_Kueppers.csv")

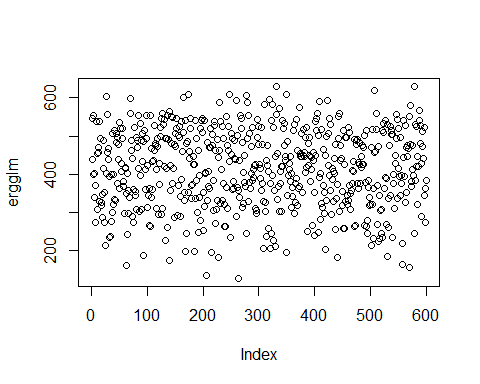
# Ergebnis

## Überblick

favstats(ergglm)

## min Q1 median Q3 max mean sd n  
## 126.5903 339.0032 418.5929 494.7188 630.1488 412.7282 102.1293 600  
## missing  
## 0

plot(ergglm)

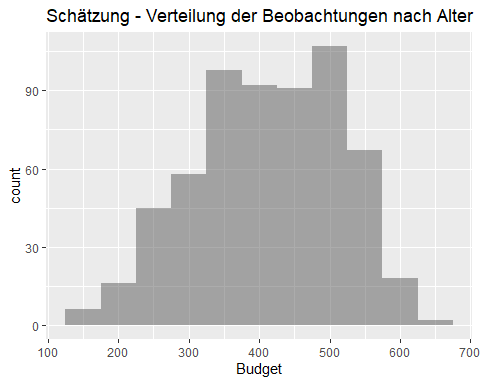


head(anwendung, n= 10)

## Beratung Angebote Bequemlichkeit Einkaufsatmosphaere Marken GPM Naehe  
## 1 7 5 5 6 3 6 6  
## 2 7 4 4 5 6 4 3  
## 3 7 5 1 7 1 6 7  
## 4 7 4 7 3 3 6 6  
## 5 5 4 7 4 3 6 2  
## 6 2 6 6 5 1 5 4  
## 7 5 7 7 7 4 5 5  
## 8 1 1 4 1 1 7 4  
## 9 5 6 2 1 3 7 4  
## 10 4 4 3 5 5 3 2  
## Partner Eltern Verwandte Kommilitonen Kinder Freunde Arbeitskollegen  
## 1 1 1 0 0 1 1 1  
## 2 1 1 0 0 1 0 0  
## 3 0 0 0 0 1 0 0  
## 4 1 0 0 0 1 1 0  
## 5 1 1 1 0 0 1 1  
## 6 1 1 1 0 0 0 0  
## 7 1 1 1 0 0 1 0  
## 8 0 1 1 0 0 0 0  
## 9 1 0 0 0 1 1 0  
## 10 1 1 0 0 0 1 0  
## Soziale.Institutionen Alter Geschlecht gift.count Budget  
## 1 0 46 2 5 546.5607  
## 2 0 38 2 3 438.3732  
## 3 0 77 2 1 397.9256  
## 4 0 84 1 3 553.5817  
## 5 0 22 2 5 402.1052  
## 6 0 30 1 3 337.9317  
## 7 0 30 2 4 371.3090  
## 8 0 39 1 2 272.6175  
## 9 0 70 2 3 539.3518  
## 10 0 27 2 3 306.2061

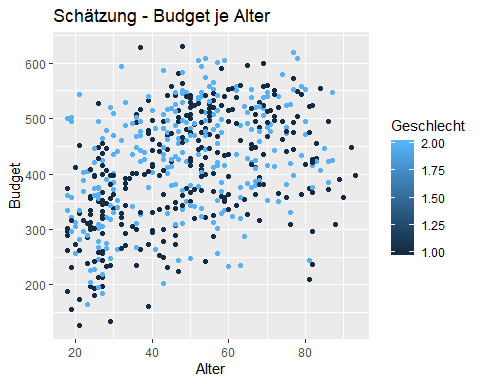
## Histogramme

gf\_histogram(~ Budget, data = anwendung, binwidth = 50, center = 50, title = "Schätzung - Verteilung der Beobachtungen nach Alter")

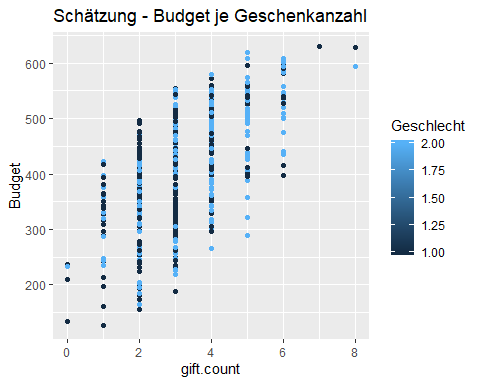


## Punktwolken

gf\_point(Budget ~ Alter, data= anwendung, colour = ~ Geschlecht, title = "Schätzung - Budget je Alter")



gf\_point(Budget ~ gift.count, data = anwendung, colour = ~ Geschlecht, title = "Schätzung - Budget je Geschenkanzahl")



# Fazit

Best analysis ever! 10 outta 10. Would analyse again