Statistical Modelling Exam - KU Leuven

Claudio Olivelli - r0787524

June 2021

1 Question 1

1.1 1A

I built three semi parametric model using the variables $x_5(MeanIncome)$and $x_9(HealthSocial)$ and obviously the target variable $y(HousePrice)$ which already represents the median of the house prices. A semiparametric model takes into account both parametric and non-parametric components. In my examples I'm going to use additive models that comprise a linear parametric component, a uni variate function and a bi variate function. I'm going to choose the best model through AIC criterion. These results show that the most complicated model is not always the best. Indeed, the minimum AIC value corresponding to the best semi-parametric model is the first one showed, partially linear and partially function based.

- $Y_i = \beta_0 + \beta_1 X_{5i} + g(x_{9i}) + \epsilon_i \qquad AIC = 1619.334$
- $Y_i = \beta_0 + \beta_1 X_{9i} + g(x_{5i}) + \epsilon_i \qquad AIC = 1626.894$
- $Y_i = \beta_0 + g(x_{9i}, x_{5i}) + \epsilon_i \qquad AIC = 1622.878$

1.2 1B

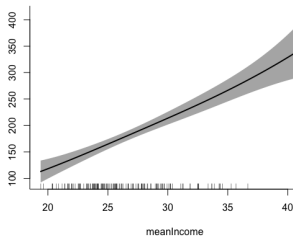

Figure 1: Plot for meanIncome non linear

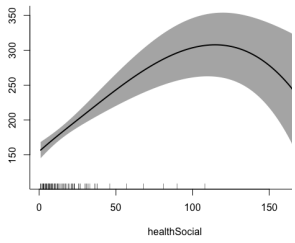

Figure 2: Plot for healthSocial non linear

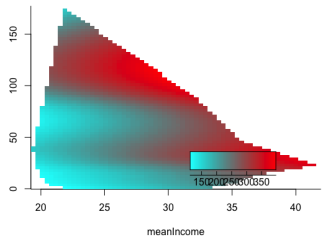

Figure 3: Heatplot for the bivariate function

1.3 1C

I selected the following model because of lower AIC:
$Y_i = \beta_0 + \beta_1 X_{5i} + g(x_{9i}) + \epsilon_i$
Since I used the function *spm()* on *R* in *library(SemiPar)*, the default function used is a radial basis function with p = 3. The notation of the function is as follows: $f(x_i) = \sum_{k=1}^{K} b_k |x_i - \kappa_k|^3$ where K is the number of knots of the polynomials. The smoothing parameter λ is automatically estimated via REML from the *spm()* function default setting.

1.4 1.4

```
library(SemiPar)
priceHouse = mydata$PriceHouse
meanIncome = mydata$MeanIncome
healthSocial = mydata$HealthSocial
fitsp3 = spm(priceHouse~ f(meanIncome) + healthSocial , spar.method = "ML")
plot(fitsp3)
fitsp4 = spm(priceHouse~ meanIncome+f(healthSocial) ,spar.method = "ML")
plot(fitsp4)
fit9 = spm(priceHouse ~ f(meanIncome , healthSocial) )
plot(fit9)
AICfitsp3 = -2*fitsp3$fit$logLik+2*fitsp3$aux$df.fit
AICfitsp3 #BEST AIC
AICfitsp4 = -2*fitsp4$fit$logLik+2*fitsp4$aux$df.fit
AICfitsp4
AICfit9 = -2*fit9$fit$logLik+2*fit9$aux$df.fit
AICfit9
```

2 Question 2

2.1 2.A

I'm going to perform a order selection test to select the best model among a parametric additive model and a set of an non parametric alternative hypothesis. The null hypothesis:
H_0 : there exist values (θ_0, θ_1) s.t. $\mu(x) = \theta_0 + \theta_1 x$
The alternative hypothesis:
H_a : $u(\cdot) \notin \{u_\theta(\cdot) : \theta = (\theta_0, \theta_1) \in R^2$
where we construct a sequence of possible models via orthogonal polynomial expansion starting from the null model $\mu_\theta(\cdot)$ as follows :
$\mu(x) = \mu_\theta(x) + \sum_{j=1}^{\inf} a_j \psi_j(x)$
where $\psi_j(\cdot)$ are the polynomial basis functions until degree $m = 2$. I am going to use a unique test statistic and not one for every alternative model. The test rejects the null hypothesis if the selected model, through AIC criteria, is different from $\mu_\theta(x)$. A nice condition to summarize this can be express as follows:

given $\hat{m}_{aic}$ be the order chosen via de model selection method AIC, the test rejects H_0 if and only if $\hat{m}_{aic} \geq 1$[**1**].
The test statistics, derived from AIC, is a function of the LRT(Likelihood ratio statistic) $T_{m,OS} = max\frac{LRT(M_m,M_0)}{m}$ [**1**]. This test statistic, under the null hypothesis, behaves asymptotically as a Chi squared distribution, so It's possible to evaluate it. The null model is going to be composed by the two covariates x_6 and x_9 representing respectively the number of tax forms and the number of health care and social service facilities plus their quadratic effects. So it follows:
Null model : $Y_0 = \theta_1 x_6 + \theta_2 x_9 + \theta_3 x_6^2 + \theta_4 x_9^2$
as alternative models , we can use all the combinations of the polynomial expansions for the covariates in the null model. Since I'm considering the polynomial expansion until the 2nd degree, and since my null model already contains my 2nd degree covariates terms, the alternative models are going to differ because of the interaction between the covariates, until the second degree. As example I chose:
$Y_{A1} = \theta_1 x_6 + \theta_2 x_9 + \theta_3 x_6^2 + \theta_4 x_9^2 + \theta_5 x_6 * x_9$

The R function *poly* doesn't recognise the degree of each term inserted, so as output it gives me also interaction between the quadratic and the linear covariates, which turns to be a 3rd degree term. To avoid this, it is possible to use the function *polym* which works well for polynomial regression for multiple independent variables.
However, my goal is to workout the test statistic and the value of m is arbitrary, thus, I choose another alternative model with a 3rd degree interaction:
$Y_{A2} = \theta_1 x_6 + \theta 2 x_9 + \theta 3 x_6^2 + \theta 4 x_9^2 + \theta 5 x_6 * x_9 + \theta 6 x_6 * x_9^2$
I obtained the three loglikelihoods:
$Y_0 = -834.1337$
$Y_{A1} = -834.0319$
$Y_{A2} = -834.1077$
which means our alternative hypothesis are not particularly interesting. Indeed, I obtained a very low $T.OS = 0.05195398$ and a $p - value = 0.9930903$ which states we cannot reject hyphotesis H_0.

2.2 2.B

```
y = mydata$PriceHouse
x6 = mydata$TaxForms
x9 = mydata$HealthSocial
m = 2#null model + 3 alternative models = 4 total models
X = poly(cbind(x6,x9, I(x6^2), I(x9^2)), m)
alternative1 <- X[,c(1,3,6,10,4)]
LogLik = rep(NA,m)
LogLik[1] = logLik(lm(y ~ x6 + x9 + I(x6^2)+ I(x9^2)))
for(j in 2 : m){
  LogLik[j] = logLik(lm(y ~ alternative1))}
LogLik
```

```
T.OS = max(2*(LogLik[2:m+1]-LogLik[1])/(1:(m)))
#WE WANT ONE test statistics NOT 3 different, we take the maximum
pvalue.Tos = function(Tos)
{mlimit = 100
1-exp(-sum((1-pchisq((1:mlimit)*Tos,1:mlimit))/(1:mlimit)))}
pvalue.Tos(T.OS)
```

3 Question 3

3.1 3.A

Linear mixed models comprise of all those models that contain both a fixed effect and a random effect in the linear regression. A random effect is a parameter that in the regression varies as a random variable. I built a trellis plot, from R library *lattice*, to show how the variable x_2(Province) affect the regression of the median house price Y on the covariate x_6(TaxForms). Before plotting, I imposed a condition on $x_6 < 40000$ to let be the plot more readable. Thus, 6 observations are not showed in the plot. However, this is not going to affect the significance of the illustration. From Figure 4 it is clear that the regression

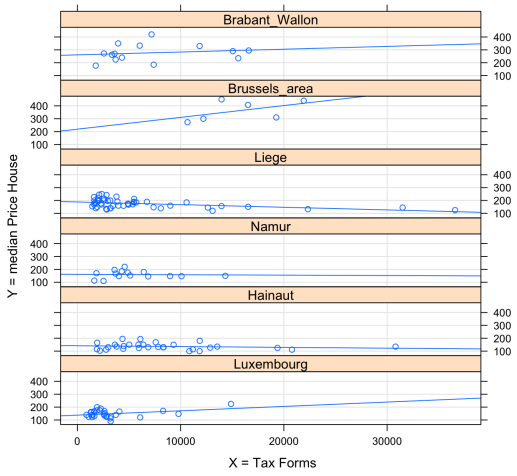

Figure 4: Trellis plot grouping by Province x_2

line changes slope Province by Province, so the use of a mixed model is justified since the random variability of the covariate x_2(Province).

3.2 3.B

I constructed a linear mixed model with covariates x_2 and x_6, with the R function lme from the R library $nlme$ to show that the plot of the previous section finds an analytical feedback in the regression R output. Let $Y_{i,j}$ be the median house price for the Tax Form j and the Province i. x_j represents the Tax Forms value, $U_i = (U_{i0}, U_{i1})^T$ represents the random intercept and slope, and ϵ_{ij} is the error terms. It is assumed independence between error terms, independence between the error terms and the random effects as well as independence among the 6 different random effects representing the Provinces. All the random effects and the error terms follow a normal distribution s.t. $u_i \sim N(0, G_0)$ where:
$G_0 = \begin{bmatrix} \sigma^2_{\mu 0} & \sigma_{\mu 01} \\ \sigma_{\mu 01} & \sigma^2_{\mu 1} \end{bmatrix}$
represents the variance-covariance matrix of the random effect vector u and $\epsilon_{ij} \sim N(0, \sigma^2_\epsilon)$.
The regression function follows:
$Y_{i,j} = \beta_0 + \beta_1 x_j + U_{i0} + U_{i1} x_j + \epsilon_{ij}$ The output in Figure 5 shows first AIC,

```
> fit_lme = lme(PriceHouse ~  TaxForms,random = ~1 | Province , data = mydata , method = "ML")
> summary(fit_lme)
Linear mixed-effects model fit by maximum likelihood
 Data: mydata
      AIC      BIC    logLik
  1548.726 1560.769 -770.3631

Random effects:
 Formula: ~1 | Province
        (Intercept) Residual
StdDev:    81.61483 37.50222

Fixed effects: PriceHouse ~ TaxForms
               Value Std.Error  DF   t-value p-value
(Intercept) 212.54297  33.84352 143  6.280167  0.0000
TaxForms     -0.00038   0.00025 143 -1.529541  0.1283
 Correlation:
         (Intr)
TaxForms -0.078

Standardized Within-Group Residuals:
       Min         Q1        Med         Q3        Max
-2.6954419 -0.5840588 -0.1553904  0.5156689  3.8258507

Number of Observations: 150
Number of Groups: 6
```

Figure 5: lme fit output

```
> intervals(fit_lme,which = "var-cov")
Approximate 95% confidence intervals

 Random Effects:
  Level: Province
                    lower     est.    upper
sd((Intercept)) 45.82429 81.61483 145.3591

 Within-group standard error:
   lower     est.    upper
33.41042 37.50222 42.09514
> library(RLRsim)
> exactRLRT(fit_lme)
Using restricted likelihood evaluated at ML estimators.
Refit with method="REML" for exact results.

        simulated finite sample distribution of RLRT.

        (p-value based on 10000 simulated values)

data:
RLRT = 152.95, p-value < 2.2e-16
```

Figure 6: Random effect std intervals and simulated LRT for the variance of the random effect

BIC and loglikelihood, measures interesting for model selection. Important for the analysis purpose is the standard deviation of the random effect, which is definitely higher than zero and so the linear mixed model with random effect x_2(Province) is justified. Thus, follows regression and statistical measures on the fixed effect as well as within group residuals, useful for model diagnostic. In Figure 6 it is showed the random effect standard error residual and a simulated LRT test on random effect variance. Since the p-value is close to zero, I have another confirm that my use of the mixed model is justified.

3.3 3.C

```
mydata_clean<-mydata[(mydata$TaxForms< 40000),]
library(lattice)
xyplot(PriceHouse ~ TaxForms | Province , mydata_clean, type = c("g","p","r"),
       index = function(x,y) coef(lm(y ~ x))[1],
       xlab = "X=Tax␣Forms",
       ylab = "Y=median␣Price␣House", aspect = "xy")
fit_lme = lme(PriceHouse ~  TaxForms,random = ~1 | Province , data = mydata
, method = "ML")
summary(fit_lme)
library(RLRsim)
exactRLRT(fit_lme)
library(nlme)
intervals(fit_lme,which = "var-cov")
```

4 Question 4

4.1 4.A

I use the *cv.glmnet* R function to perform 4 different regression models on the full dataset(*Municipalities* dropped), where y ($PriceHouse$) is the target variable, and there are no interactions. I used the following estimation methods: MLE, Ridge regression($\alpha = 0$), Lasso estimation ($\alpha = 1$))and an elastic net estimator($\alpha = 0.5$).

	Intercept	*meanIncome*	*healthSocial*	*industries*	*hotelRestaurant*	*regionWaals*	*prov.Brussels*
MLE	1.780e+02	6.709e+00	9.426e-01	1.404e-01	2.688e-02	-1.332e+02	NA
RIDGE	7.501e+01	7.555e+00	5.425e-01	8.135e-02	-1.802e-02	-7.182e+01	7.211e+01
$\alpha = 0.5$	1.661e+02	7.064e+00	7.708e-01	9.330e-02	.	-1.346e+02	.
LASSO	99.626	6.994	0.813	0.107	.	-65.978	68.390

	prov.Hainaut	*prov.Liege*	*prov.Luxembourg*	*prov.Namur*	*shops*	*bankruptcies*	*taxForms*
MLE	-8.004e+01	-5.543e+01	7.600e+01	-6.519e+01	-3.810e-01	-1.465e-02	-1.366e-03
RIDGE	-5.540e+01	-3.294e+01	-5.186e+01	-3.986e+01	2.439e-01	-2.766e-03	-4.967e-04
$\alpha = 0.5$	-7.383e+01	-4.974e+01	-7.032e+01	-5.832e+01	-2.755e-01	-6.582e-03	-9.645e-04
LASSO	-74.542	-50.481	-70.871	-59.245	-0.305	-0.007	-0.001

4.2 4.B

Now I take an observation with a low industrial firms amount(15th out of 150 in an increasing order for that variable) and an observation with a high number of

hotel(130th out of 150 in an increasing order for that variable). I want to predict the median price of the house y. I use the library glmnetUtils and the function *cv.glmnet* to create the fit and *predict.cv.glmnet.formula* for the prediction. In both cases the MLE estimation method gives the lowest prediction while the Ridge method give the highest.

	Low_industries	*High_hotel*
MLE	139.5456	182.6345
RIDGE	141.4122	184.2874
$\alpha = 0.5$	140.9964	182.8463
LASSO	140.442	183.3513

4.3 4.C

```
xmatrix <- model.matrix(PriceHouse ~ MeanIncome+HealthSocial+Industries
+HotelRestaurant +Region +Province + Shops + Bankruptcies +TaxForms
,data=mydata)[,-1]
alpha = c(0,1,.5)
for (a in alpha){
glmnet_fit  <- glmnet(xmatrix,priceHouse,alpha = a)
cv_fit <- cv.glmnet(xmatrix,priceHouse,alpha = a, folds=5)
coefficients <-coef(cv_fit,s = cv_fit$lambda.min)}
s <- cv_fit$lambda.min #minimum lambda value(the best, cross validation is included)
logitRegression = lm(PriceHouse ~ MeanIncome+HealthSocial+ Industries
+HotelRestaurant + Region +Province +Shops +Bankruptcies +TaxForms ,data=mydata,
family = binomial)
print(coef(logitRegression))
library(glmnetUtils)
mydata_grouped_1 <- mydata[order(mydata$Industries),]
ind <- select(mydata[15,],-PriceHouse, -Municipality)
for (a in alpha){
h <- glmnetUtils::cv.glmnet(PriceHouse ~ MeanIncome+HealthSocial+Industries +
HotelRestaurant +Region +Province +Shops + Bankruptcies +TaxForms,data = mydata,
alpha = a, folds=5)
pred1 <- glmnetUtils:::predict.cv.glmnet.formula( h, new = ind, s = "lambda.min")}
mydata_grouped_2 <- mydata[order(mydata$HotelRestaurant),]
hotel <- select(mydata[130,],-PriceHouse, -Municipality)
for (a in alpha){
  glm = cv.glmnet(PriceHouse ~ MeanIncome+HealthSocial+Industries +HotelRestaurant
+Region + Province + Shops +Bankruptcies +TaxForms,data = mydata,alpha = a, folds=5)
pred1 <- predict(glm,  new = hotel,  s = "lambda.min")}
#MLE prediction
pred_ind <- predict(logitRegression, newdata = ind, type = "response")
pred_hotel <-predict(logitRegression, newdata = hotel, type = "response")
```

5 References

[**1**] Statistical Modelling , G. Claeskens Acco course notes