TMA4268 V2024 Exam

TMA4268 Statistical Learning V2024

Stefanie Muff and Sara Martino, Department of Mathematical Sciences, NTNU

May 11, 2024

Check in the end which libraries we need:

Problem 1 (Fill-in-the-blank text, 5P)

0.5P per correct answer

Read the whole text and fill in the blanks such that the whole text makes sense (you might only understand which answer is correct after you continued reading):

In this course we learned about methods that can be used for statistical learning. We have mainly discussed supervised statistical learning methods, broadly divided into regression and classification problems. We were thereby discriminating between models that we use for prediction versus inference.

If the main goal is inference, we usually prefer *parametric* (non-parametric, unsupervised, more advanced, more flexible, regression, classification) methods, because the goal is then to *understand* (predict, minimize the sum of bias and variance, minimize prediction error). If the goal is pure prediction, we can use any method that performs well, whereas *non-parametric* (regression, classification, clustering, supervised) methods are very flexible and thus more complex and *less interpretable* (better interpretable, more appealing, better suited for inference) than, for example, linear models.

One important concept in the course was the bias-variance trade-off, and in that context we discussed variable selection and methods based on shrinkage. Lasso and ridge regression are two shrinkage methods we learned about. Lasso is an alternative to AIC-based model selection in linear models and should be preferred since it avoids *model selection bias* (underestimated parameter estimates, large prediction error, irreducible error, over-fitting). In the context of trees, shrinkage-type regularization is performed via *tree pruning* (bagging, random forests, boosting, classification trees, regression trees), whereas neural networks do this via *weight penalization* (data augmentation, label smoothing, dropout, early stopping). More generally, any type of regularization has as a main goal to *reduce test error* (reduce training error, reduce bias, increase the flexibility of the model, shrink parameters) by avoiding that the model over-fits the data.

In unsupervised learning the goal is to discover interesting aspects about the data when the *response variable* (covariate, loss function, parametric model, reducible error, bias-variance trade-off) is not present. We have considered two types of methods: principal component analysis and clustering. Both methods have strong focus on finding *groups* (good predictions, bias, variance) in the data, and on visualization.

Problem 2 (Multiple choice and numeric answer, 7P)

a) (3P)

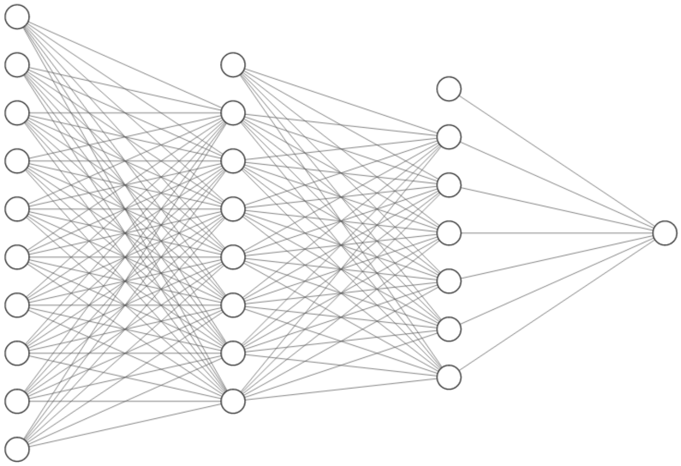
Which statements are correct? Here n are the number of data points and p the number of available variables.

- Forward selection requires that $p < n$.
- [correct] Backward selection requires that $p < n$.

- Lasso requires that $p < n$.
- [correct] Ridge regression is possible even if $p > n$.
- Neural networks must have $p > n$.
- [Correct] Boosted regression trees allow for $p > n$.

b) (2P)

Look at the following architecture of a feed-forward neural network:



Which statements are correct?

- [correct] This network has 9 input variables, two hidden layers and a continuous or binary outcome.
- This network has 10 input variables and two hidden layers with 8 and 7 nodes, respectively.
- [correct] This network can be used for binary classification.
- In this network we estimate 143 parameters.

Solution:

The second statement is wrong, because there are bias nodes in each layer. We estimate $125 = 10 \cdot 7 + 8 \cdot 6 + 7$ parameters.

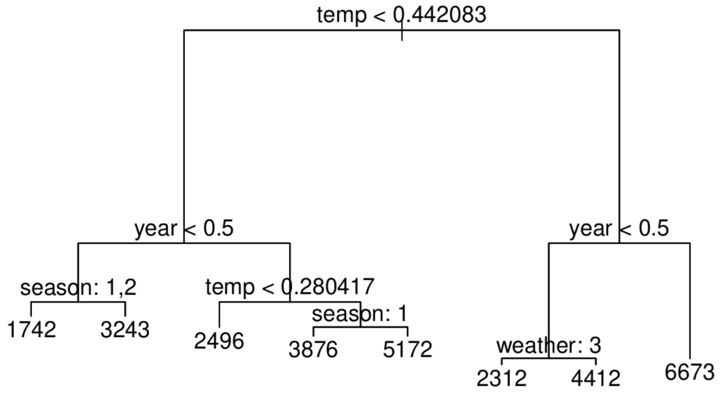
c) (1P) Numeric answer

A covariate is included in a regression model as a natural cubic spline with three cut points. How many degrees of freedom does this spline term consume?

Solution: 4

d) (1P) Numeric answer

We have fitted a tree to some training data to predict the number of bikes rented out in a given city on a specific day, and get the following result:



Using the fitted regression tree, what would you predict as the number of bikes rented on a day with the following covariates:

- `year=0`
- `season=3`
- `holiday=0`
- `notworkday=0`
- `temp=0.3`
- `weather=3`
- `wind=0.3`

Solution: 3243

Problem 3 (theory, 8P)

a) (4P)

We learned about k-fold cross-validation (CV) as a way of doing model selection.

(i) (2P) Explain how k-fold CV is implemented and how the MSE is computed.
(ii) (2P) State what the advantages and disadvantages of k-fold cross-validation (CV) are with respect to the Leave-one-out cross-validation (LOOCV) approach.

Solution:

(i) A training set is randomly divided into k groups of equal size. The first group is used as the validation set, while the model is fit on the remaining $k-1$ groups which constitute the training set. The MSE is calculated using the validation set. This procedure is repeated k times and on each occasion the validation set and training sets will be different than the previous one. We then take the average of all the MSE's as the final MSE.

(ii) k-fold CV is less computationally demanding: If $k=5$ then only 5 models need to be fitted, while with LOOCV where n models need fitting. k-fold CV has higher bias than LOOCV, as fewer observations are used, but tends to have lower variance.

b) (4P)

Consider a regression setting and assume an additive error model:

$$Y_i = f_\theta(X_i) + \epsilon_i, \qquad i = 1, \ldots, N$$

where θ denotes the model parameters defining the regression function f_θ, and ϵ is the error term.

i) (2P) Describe the least squares method and the maximum likelihood method and how they are used to estimate θ. It is enough to state the optimization problems, you do not need to solve them. State also the assumption that are made for each estimation method

ii) (2P) Show that, if you assume a Gaussian distribution for the error term, then the two methods are equivalent with respect to the estimate of θ.

Solution:

i) In the least squares method we assume that the error terms $\epsilon_1, \dots, \epsilon_N$ are independent, incorrelated, and all have the same variance. We estimate parameters by minimizing the the residual sum-of-squares (RSS):

$$\hat{\theta}_{LSE} = \text{argmax}_\theta \left\{ \sum_{i=1}^{N} (y_i - f_\theta(X_i))^2 \right\}$$

In the maximum likelihood method we additionally assume taht the error terms are normally distributed. We then estimate parameters by maximizing the log-likelihood

$$\hat{\theta}_{MSE} = \text{argmax}_\theta \left\{ \sum_{i=1}^{N} \log(p(y_i | f_\theta(X_i), \sigma^2)) \right\}$$

ii) Assuming a Gaussian distribution for the error term implies that $Y_i \sim \mathcal{N}(f_\theta(X_i), \sigma^2)$ which gives the log-likelihood function

$$\begin{aligned} l(\theta) &= \sum \left(-\frac{1}{2}\log(2\pi) - \frac{1}{2}\log(\sigma^2) - \frac{1}{2\sigma^2}(y_i - f_\theta(X_i))^2 \right) \\ &= -\frac{N}{2}\log(2\pi) - \frac{N}{2}\log(\sigma^2) - \frac{1}{2\sigma^2}\sum (y_i - f_\theta(X_i))^2 \end{aligned}$$

Since the first two terms in the expression above do not depend on θ, the only term to maximize is the sum $\sum (y_i - f_\theta(X_i))^2$, which gives the same optimization problem as for the least square method.

Problem 4 – Data analysis 1 (16P)

Here we are looking at a regression problem, where we want to understand the factors that affect the sales of child carseats in 400 different stores. We use the `Carseats` data set from the ISLR package, which you can load using the code below

```r
library(ISLR)
library(leaps)
library(glmnet)

data(Carseats)
```

It is useful to investigate the data for example using the code below and by typing `?Carseats` into the R console:

```r
# Look at the data, for example using:
pairs(Carseats)
str(Carseats)
```

a) (8P)

(i) (1P) Fit a multiple regression model to predict `Sales` using `Price`, `ShelveLoc`, `US` and `Population` as predictors.

(ii) (1P) Provide an interpretation of the estimated coefficient for Price. How much does the sales changes if price is increased by 10 dollars?

(iii) (2P) Perform and F test to asses the significance of the `ShelveLoc` covariate and state your conclusion. If you move an item from a Medium to a Good shelve location, how does the expected number of sales changes?

(iv) (2P) Compute the predicted sales for two observations both with a price of 100 dollars and population of 10 000 inhabitants but one with bad shelving location and not located in the US while the other with medium shelving and located in the US.

(v) (2P) Look at the plots produced by

```
plot(mod)
```

where `mod` is your fitted model. Explain how each of the plots can be used to assess the fitted model. When relevant, state which assumptions are addressed and if those are met for the model under consideration.

Solution

(i)

```
mod = lm(Sales ~Price + ShelveLoc + US + Population , data = Carseats)
summary(mod)
```

```
## 
## Call:
## lm(formula = Sales ~ Price + ShelveLoc + US + Population, data = Carseats)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -5.1485 -1.2626 -0.0447  1.2712  4.6181 
## 
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)    
## (Intercept)     11.2530762  0.5269228  21.356  < 2e-16 ***
## Price           -0.0577499  0.0039355 -14.674  < 2e-16 ***
## ShelveLocGood    4.8346990  0.2771233  17.446  < 2e-16 ***
## ShelveLocMedium  1.9057673  0.2274991   8.377 9.65e-16 ***
## USYes            0.9980082  0.1952198   5.112 4.98e-07 ***
## Population       0.0008152  0.0006322   1.290    0.198    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 1.856 on 394 degrees of freedom
## Multiple R-squared:  0.5736, Adjusted R-squared:  0.5682 
## F-statistic:   106 on 5 and 394 DF,  p-value: < 2.2e-16
```

(ii) The model suggests a negative relationship between price and sales. If we increase the price by 10 dollars the sales are espected to diminish by -0.577 thousands units.

iii)

Anovae for `SelveLoc`:

```
r.lm <- lm(Sales ~Price +  US + Population + ShelveLoc , data = Carseats)
anova(r.lm)
```

```
## Analysis of Variance Table
## 
## Response: Sales
##             Df  Sum Sq Mean Sq  F value    Pr(>F)    
## Price        1  630.03  630.03 182.9484 < 2.2e-16 ***
## US           1  131.37  131.37  38.1472 1.635e-09 ***
## Population   1    3.40    3.40   0.9858    0.3214    
## ShelveLoc    2 1060.64  530.32 153.9942 < 2.2e-16 ***
## Residuals  394 1356.84    3.44                      
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We conclude that the shelve location appears to be very relevant.

By moving an item from a Medium to a Good location, we expect to increase the sales by 2.9289317 thousends units.

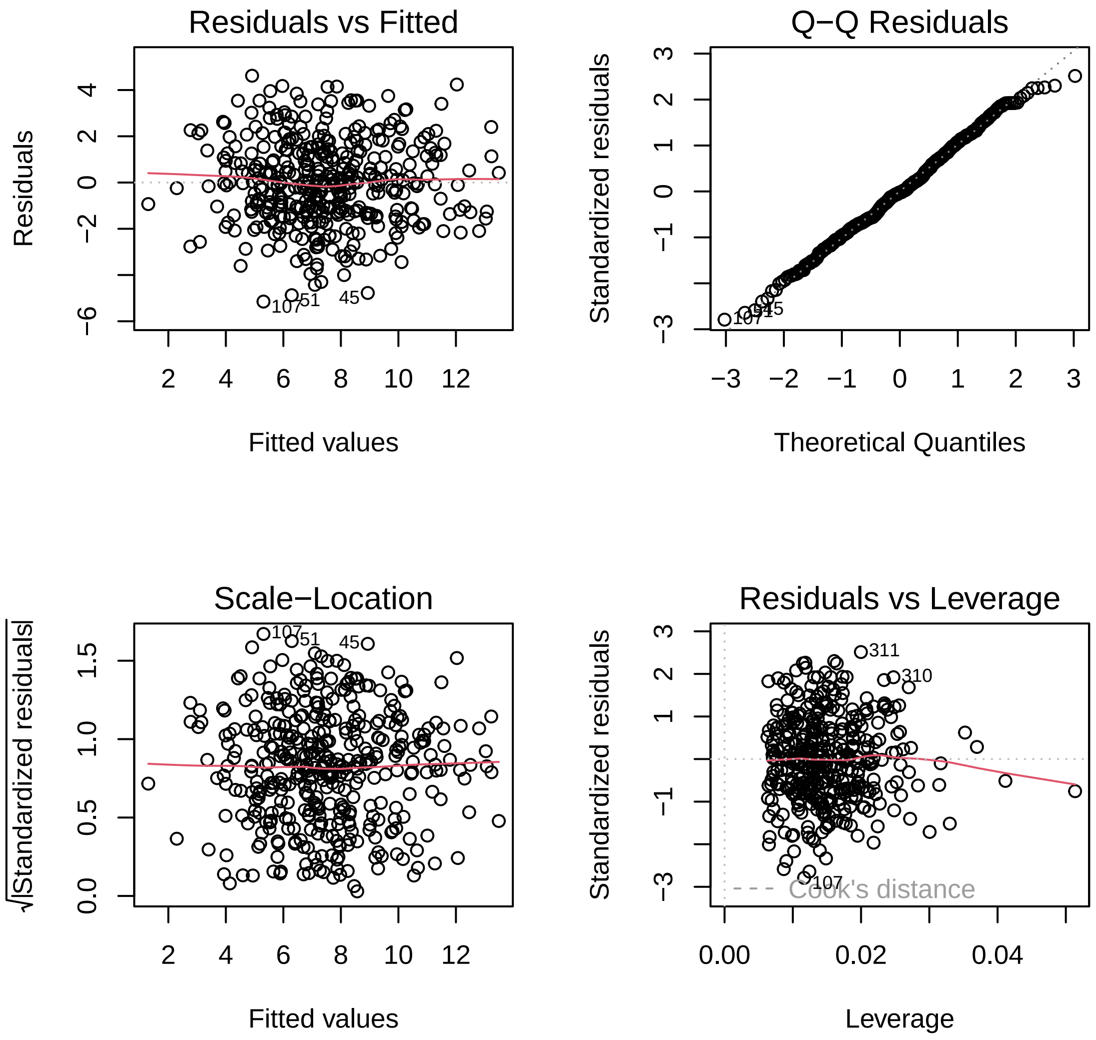
(iii)

```
new_data = data.frame(Price = rep(100,2),
                      ShelveLoc = factor(c("Bad", "Medium"), levels = c("Bad","Medium","Good")),
                      US = factor(c("No", "Yes"), levels = c("Yes","No")),
                      Population = rep(10000,2))

predict(mod, newdata = new_data)
```

```
##        1        2 
## 13.63044 16.53422
```

iv)

```
par(mfrow=c(2,2))
plot(mod)
```



The assumptions we want to check are that the mean of the error terms is zero, that the error variance is constant, that the errors are (approximately) independent and that the distribution of the error term is Gaussian.

The first plots tells us that residuals are centered around 0 and that their variance does not seem to change with the fitted values. The second plots show a QQ plot and it is consistent with the assumption of Gaussian distribution. The third plot also does not show any worrying pattern. The fourth plot one datapoint seems to have a high leverage but not a high residual.

b) Model Selection (8P)

We now consider all predictors in the `Carseats` dataset except for `ShelveLoc`. Our goal is to build a model with as few predictor as possible but that still gives good predictions.

In order to assess the robustness of our models, we split the data into a training and a test set as follows:

```
set.seed(1234)
# remove ShelveLoc covariate
car.all = Carseats[,-7]
```

```r
# create train and test datasets
samples <- sample(1:400,250, replace=F)
car.train <- car.all[samples,]
car.test <- car.all[-samples,]
```

i) (1P) Fit a model on the train data with all `Sales` as response. Use all predictors except `ShelveLoc` and add also a quadratic effect of `Age`.

ii) (1P) Look at the summary of your model. How do you interpret the F-statistics in the last row?

iii) (2P) Carry out Ridge regression on the training set, choose the largest lambda within 1 standard error from the lambda with the minimal error in a 5-fold cross-validation. As above, include the quadratic term for `Age` . Report the MSE of test data.

Requirement: Use set.seed(1100) before running the cross-validation.

iv) (2P) Carry out Lasso regression on the training set, choose the largest λ within 1 standard error from the λ with the minimal error in a 5-fold cross-validation. As above, include the quadratic term for `Age` . Report the MSE of test data.

Requirement: Use set.seed(1100) before running the cross-validation.

v) (2P) Compare the estimated coefficients you obtained with Ordinary least square, Ridge and Lasso regression. What patterns do you notice? Are there some advantages/disadvantages of Lasso over ridge regression?

R-hints:

```r
x.train <- model.matrix(Sales ~ ..., data = car.train)
set.seed(1100)

#ridge
cv_ridge <- cv.glmnet(x.train,  car.train$Sales, alpha = ...)
plot(cv.ridge)
cv.ridge$...
fit_ridge = glmnet(..., ... , alpha=..., lambda = ...)
```

Solution

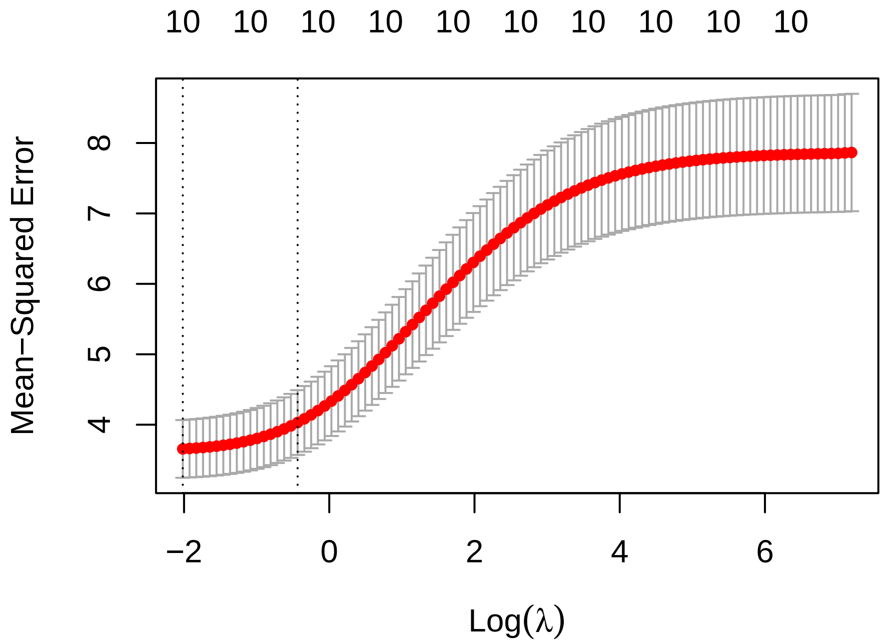
i)

```r
mod  = lm(Sales ~ CompPrice + Income + Advertising + Population + Price + Age + I(Age^2) + Education +U


p = predict(mod, car.test)
mse = mean((p-car.test$Sales)^2)
```

ii) The F-Statistic is a "global" test that checks if at least one of your coefficients are nonzero. In this case the p-value is very small so we reject the null hypothesis and conclude that at least one of the coefficients is not zero.

iii)

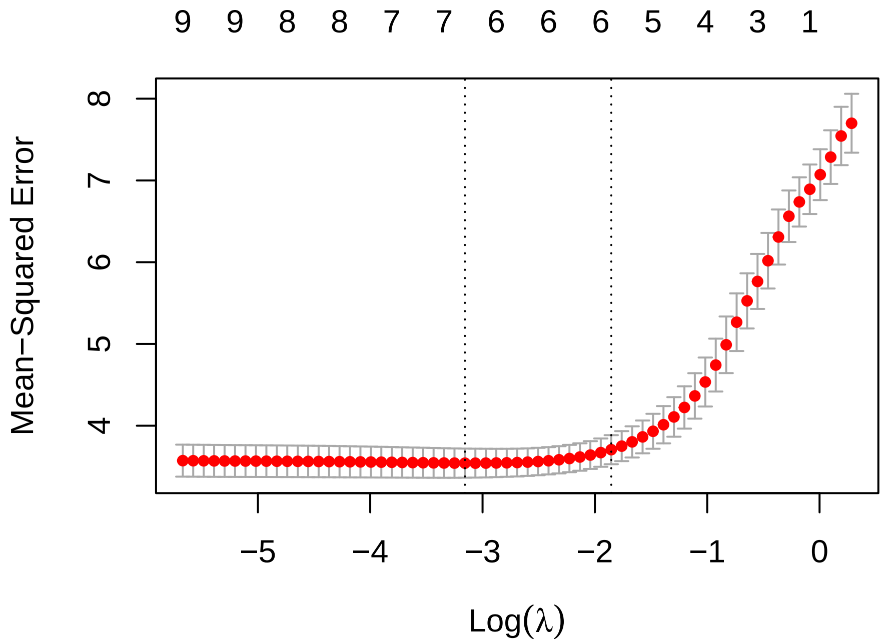
```r
x.train <- model.matrix(Sales ~ . + I(Age^2), data = car.train)
set.seed(1100)

#ridge
cv_ridge <- cv.glmnet(x.train,  car.train$Sales, alpha = 0, nfolds = 5)
plot(cv_ridge)
```



```
fit_ridge = glmnet(x.train, car.train$Sales, alpha=0, lambda = cv_ridge$lambda.1se)
p <- predict(fit_ridge,   model.matrix(Sales ~ . + I(Age^2), data = car.test), s = fit_ridge$lambda.1se

mse_ridge <- mean((p - car.test$Sales)^2)
coef_ridge = coef(fit_ridge)
```

(iv)

```
#lasso
cv_lasso <- cv.glmnet(x.train, car.train$Sales, alpha = 1, nfolds = 5)
plot(cv_lasso)
```



```
fit_lasso = glmnet(x.train, car.train$Sales, alpha=1, lambda = cv_lasso$lambda.1se)
p <- predict(cv_lasso, model.matrix(Sales ~ . + I(Age^2), data = car.test), s = cv_lasso$lambda.1se)
(mse_lasso <- mean((p - car.test$Sales)^2))
```

```
## [1] 4.550921
```

```
coef_lasso = coef(fit_lasso)
```

(v)

```
c(mse, mse_ridge, mse_lasso)
```

```
## [1] 4.361316 4.775475 4.550921
```

```
cbind(coef(mod), coef_ridge, coef_lasso)
```

```
## 12 x 3 sparse Matrix of class "dgCMatrix"
##                                         s0            s0
## (Intercept)  7.6749702958  9.4731665389  9.567817e+00
## (Intercept)  0.0854487046  .             .
## CompPrice    0.0092523031  0.0462011331  5.869672e-02
## Income       0.1282580257  0.0081055117  3.745711e-03
## Advertising  0.0006150580  0.0791587547  9.748975e-02
## Population  -0.0968060042  0.0010575210  7.976899e-05
## Price        0.0196436861 -0.0652430396 -7.920475e-02
## Age         -0.0006945855 -0.0201088590  .
## Education   -0.0141359220 -0.0044518521  .
## UrbanYes     0.0193260559  0.0039205759  .
## USYes       -0.1679413948  0.2424104757  .
## I(Age^2)     7.6749702958 -0.0002341977 -4.106228e-04
```

The estimated parameters with Lasso and Rigde regression are shrunken towards 0. While in Lasso regression some parameters are exactly 0 the same does not happen with ridge regression. This is expected due to the diffent form of the penalization used.

Problem 5 – Data analysis 2 (16P)

We are using a dataset on purchase of orange juice, where the costumors either purchased the brand "Citrus Hill" (CH) or "Minute Maid" (MM). The dataset is available in the `ISLR` package and can be loaded and modified as follows:

```
library(ISLR)
data(OJ)
# Select a subset of columns
d.OJ <- OJ[,c("Purchase","WeekofPurchase","PriceDiff","PriceCH","PriceMM","SpecialMM","LoyalCH","PctDis
d.OJ$Purchase <- ifelse(d.OJ$Purchase=="CH",0,1)
```

Note that we have coded the purchase of Citrus Hill as 0, and Minute Maid as 1.

You can look up what the different covariates in the dataset actually mean by typing `?OJ` in the R console.

Before starting, it is smart to investigate the data a little bit, for example by making `pairs` plots or looking at the structure using `str(d.OJ)` etc.

We are interested in understanding and predicting the purchase of Minute Maid vs Citrus Hill.

a) (3P)

(i) (1P) Fit a logistic regression model on the full data set with `Purchase` as response variable, using all the covariates.

(ii) (2P) Since we have a model where the prices for both brands, as well as the price differences are included, we fit another logistic regression model where we use all covariates *except* `PriceCH` and `PriceMM`. Given the fitted model, quantify the effect of the price difference (`PriceDiff`) on purchase of MM when the price difference increases by 0.1 units.

Solution

(i)

```
r.glm <- glm(Purchase ~ ., data=d.OJ, family="binomial")
summary(r.glm)
```

```
##
## Call:
## glm(formula = Purchase ~ ., family = "binomial", data = d.OJ)
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)      4.202288   1.713031   2.453   0.0142 *
## WeekofPurchase  -0.003816   0.009344  -0.408   0.6830
## PriceDiff      -18.243629   7.978000  -2.287   0.0222 *
## PriceCH        -14.499587   7.912922  -1.832   0.0669 .
## PriceMM         14.682898   7.794797   1.884   0.0596 .
## SpecialMM        0.330915   0.261519   1.265   0.2057
## LoyalCH         -6.316655   0.395262 -15.981   <2e-16 ***
## PctDiscMM      -33.372974  16.732043  -1.995   0.0461 *
## PctDiscCH       27.627027  15.014633   1.840   0.0658 .
## Store7Yes       -0.500780   0.216383  -2.314   0.0206 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1430.85  on 1069  degrees of freedom
## Residual deviance:  821.05  on 1060  degrees of freedom
## AIC: 841.05
##
## Number of Fisher Scoring iterations: 5
```

(ii)

```
r.glm2 <- glm(Purchase ~ . -PriceCH - PriceMM, data=d.OJ, family="binomial")
summary(r.glm2)
```

```
##
## Call:
## glm(formula = Purchase ~ . - PriceCH - PriceMM, family = "binomial",
##     data = d.OJ)
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)     3.5626200  1.4805725   2.406  0.01612 *
## WeekofPurchase -0.0001635  0.0060666  -0.027  0.97849
## PriceDiff      -3.3545092  0.8500701  -3.946 7.94e-05 ***
## SpecialMM       0.2964072  0.2565358   1.155  0.24792
## LoyalCH        -6.2853233  0.3933704 -15.978  < 2e-16 ***
## PctDiscMM      -2.2020920  2.1759063  -1.012  0.31152
## PctDiscCH      -0.3373778  2.1948169  -0.154  0.87783
## Store7Yes      -0.5823150  0.2108294  -2.762  0.00574 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1430.85  on 1069  degrees of freedom
## Residual deviance:  824.64  on 1062  degrees of freedom
## AIC: 840.64
##
## Number of Fisher Scoring iterations: 5
```

The *odds ratio* for purchase decreases by a factor of $\exp(\beta \cdot 0.1) = 0.715$

Common mistakes:

- Used linear regression, e.g. by not specifying family="binomial" (-1)
- Incorrect explanation or conclusion (-1)

b) (5P)

Now split the dataset into a training and a test sample for prediction (assuming our aim is to predict purchase of MM). Split the dataset as in the code below, using the same seed. Then

(i) (1P) Perform a quadratic discriminant analysis on the training data.
(ii) (1P) Use the fitted model to predict purchase of MM in the test set using a probability cutoff of $p = 0.5$.
(iii) (1.5P) Generate the confusion table and calculate the error rate, sensitivity and specificity for the prediction on the test set.

(iv) (1.5P) Generate the confusion table and calculate the error rate, sensitivity and specificity also for the logistic regression model from a) i), when trained on the training data and then predicted on the test data. Use again $p = 0.5$ as cutoff.

R-hints:

```
set.seed(4268)
samples <- sample(1:1070,1070*0.7, replace=F)
d.OJ.train <- d.OJ[samples,]
d.OJ.test <- d.OJ[-samples,]
```

```
library(MASS)
qdaMod <- qda()
postQDA = predict(qdaMod,newdata=..)$class
table(...)
```

Solution:

```
library(MASS)
qdaMod = qda(Purchase~ . , data = d.OJ.train)
# $class is doing the 0.5 cutoff, while $posterior gives the actual probabilities:
predQDA = predict(qdaMod, newdata = d.OJ.test)$posterior
predQDA = predict(qdaMod, newdata = d.OJ.test)$class
tQDA = table(true=d.OJ.test$Purchase, predict=predQDA)
tQDA
```

```
##     predict
## true   0   1
##    0 161  37
##    1  27  96
```

```
sensQDA = tQDA[2, 2]/(sum(tQDA[2,]))
spesQDA = tQDA[1, 1]/(sum(tQDA[1,]))
c(sensitivity = sensQDA, specificity =  spesQDA)
```

```
## sensitivity specificity
##   0.7804878   0.8131313
```

```
(error.rate.QDA <- (tQDA[1, 2] +tQDA[2, 1]) /(sum(tQDA)))
```

```
## [1] 0.1993769
```

iv)

```
r.glm.train <- glm(Purchase ~ ., data=d.OJ.train, family="binomial")
pred.glm <- round(predict(r.glm.train,newdata=d.OJ.test,type="response"),0)
t.glm = table(true=d.OJ.test$Purchase, predict=pred.glm)
t.glm
```

```
##     predict
## true   0   1
##    0 167  31
##    1  29  94
```

```
sens.glm = t.glm[2, 2]/(sum(t.glm[2,]))
spes.glm = t.glm[1, 1]/(sum(t.glm[1,]))
c(sensitivity = sens.glm, specificity =  spes.glm)
```

```
## sensitivity specificity
##   0.7642276   0.8434343
```

```
(error.rate.glm <- (t.glm[1, 2] +t.glm[2, 1]) /(sum(t.glm)))
```

```
## [1] 0.1869159
```

Common mistakes:

- trained model on all data (-1)
- didn't include/wrong sens. and/or spec. (-0.5) (only if they show confusion matrix, otherwise -1)

c) (4P)

(i) (3P) We continue analyzing the same that, focusing on the task to obtain good predictions of what the custormers purchase. To this end, use a generalized additive model (still for the binary outcome `Purchase`) that only contains smoothed versions of `PriceDiff` and `LoyalCH` (no other covariates). Fit it on the training data and explain the details of your choice, for example how many degrees of freedom the smoothed terms consume and what functional form they have. Use maximum 5 sentences.

(ii) (1P) Calculate the misclassification error rate on the test data. Is it lower than for logistic regression and QDA?

Solution:

(i)

```
r.gam.train <- glm(Purchase ~ ns(PriceDiff,5) + ns(LoyalCH,5), data=d.OJ.train, family="binomial")
pred.gam <- round(predict(r.gam.train,newdata=d.OJ.test,type="response"),0)
t.gam = table(true=d.OJ.test$Purchase, predict=pred.gam)
t.gam
```

```
##     predict
## true   0   1
##    0 168  30
##    1  29  94
```

(ii) Yes the error rate is lower than before, but not much (and depending on the choice, it might be the same or slightly higher even than for logistic regression):

```
(error.rate.gam <- (t.gam[1, 2] +t.gam[2, 1]) /(sum(t.gam)))
```

```
## [1] 0.1838006
```

d) (4P)

(i) (2P) Finally, you should use a gradient tree boosting method. Take a large enough number of trees for a given learning rate and then choose the tree number that gives the lowest error rate for a 10-fold CV. Explain your choices. Use the R-hints below and fit the model on the training data.

(ii) (2P) Calculate the misclassification error on the test data. Compare to the findings from b) and c) and interpret in 1-2 sentences.

R-hints:

```
library(gbm)
# Check the help file:
?gbm()
gbm(..., n.trees=..., shrinkage=...,interaction.depth=...,cv.folds=10)
```

Solution:

(i) Students should explain `n.trees`, `shrinkage` and `interaction.depth`. Since we are choosing the best number of trees at the end, `n.trees` should be large enough, because we then anyway do early stopping.

```r
library(gbm)
gbm1 <- gbm(
  formula = Purchase ~ .,
  data = d.OJ.train,
  distribution = "bernoulli",
  n.trees = 1000,
  shrinkage = 0.01,
  interaction.depth = 2,
  n.minobsinnode = 10,
  cv.folds = 10
)

# To check out which number of trees gives the best CV error
(best.iter <- which.min(gbm1$cv.error))
```

```
## [1] 568
```

(ii) Calculation of the error rate:

```r
pred.gbm <- round(predict(gbm1,n.trees=best.iter,newdata=d.OJ.test,type="response"),0)
t.gbm = table(true=d.OJ.test$Purchase, predict=pred.gbm)
t.gbm
```

```
##     predict
## true   0   1
##    0 167  31
##    1  29  94
```

```r
(error.rate.gbm <- (t.gbm[1, 2] +t.gbm[2, 1]) /(sum(t.gbm)))
```

```
## [1] 0.1869159
```

Here, most students will find an error rate that is not lower than for logistic regression or the GAM. So they should hypothesize that boosted trees are probably not needed, and simpler models can (or should) be used.

Multiple and single choice and numerical answer questions

Problem 6 (7.5P)

a)(1P) (numerical answer)

We have a dataset that we want to use to predict the starting salary after graduation, based on three predictors: the GPA (Grade Point Average, a number that indicates how high you scored in your courses on average), the IQ and the gender.

Assume we use R to fit the following linear model

```r
formula = salary ~ GPA + IQ + GENDER + GPA:IQ + GPA:GENDER
mod = lm(formula, data = dataset)
```

Suppose we use least squares to fit the model, and get the following estimates

```
mod$coefficients
   (Intercept)      GPA      IQ     GENDERFemale    GPA:IQ     GPA:GENDERFemale
     50             20      0.07          35         0.01              -10
```

i) What is the predicted salary for a female with IQ of 110 and GPA of 4.0? Give your answer with a precision of one decimal after the period.

Solution:

[137.1]

b) Single Choice (1P)

In the same problem as **a)** , which sentence is correct:

- For a fixed value of IQ and GPA, males earn more on average than females.
- For a fixed value of IQ and GPA, females earn more on average than males.
- For a fixed value of IQ and GPA, males earn more on average than females provided that the GPA is high enough.[correct]
- For a fixed value of IQ and GPA, females earn more on average than males provided that the GPA is high enough.

c) (1P)

For which of the following techniques it is important to standardize the predictors:

- Multiple Linear Regression
- Ridge Regression
- Principal Component Analysis
- K-nearest neighbour classification

Solution:

[FALSE, TRUE, TRUE, TRUE]

d) (2P)

For each optimization criterion choose the correct method from the drop down menu:

- $\text{argmin}_\beta \sum_{i=1}^n (y_i - \mathbf{x}_i^T\beta)^2$: *least square regression*
- $\text{argmin}_\beta \sum_{i=1}^n (y_i - \mathbf{x}_i^T\beta)^2 + \lambda \sum_{j=1}^p |\beta_j|$: *Lasso regression*
- $\text{minimize}_{C_1,\ldots,C_K} \left\{ \sum_{k=1}^K \frac{1}{|C_k|} \sum_{i,i'\in C_k} \sum_{j=1}^p (x_{ij} - x_{i'j})^2 \right\}$: *K-means clustering*
- $\text{argmin}_{R_1(j,s),R_2(j,s)} \left[\sum_{i:x_i\in R_1(j,s)} (y_i - \hat{y}_{R_1})^2 + \sum_{i:x_i\in R_2(j,s)} (y_i - \hat{y}_{R_2})^2 \right]$: *Regression tree*

e) (2.5P)

We are looking at the `state.x77` dataset given in R. This dataset consists of data related to the 50 states of the United States of America. The dataset contains 8 variables regarding different aspect of the state. You can check the data in R typing:

```
data(state)
?state.x77
```

to see what the different variables mean.

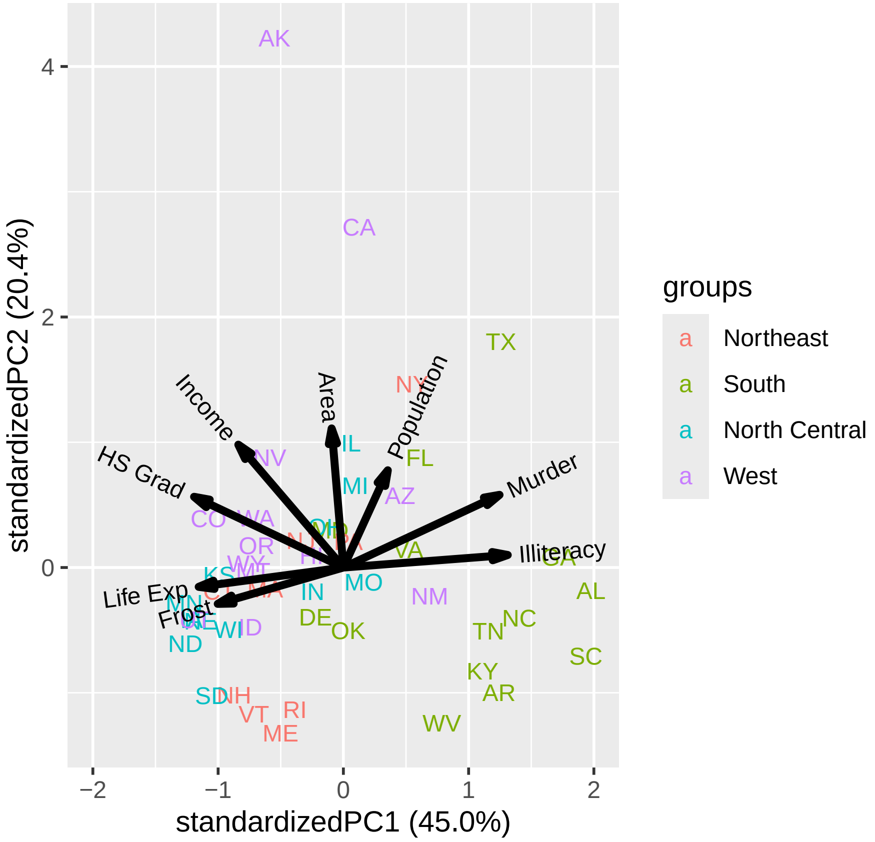
Here we carried out a principal component analysis and give the biplot and the scree plot below. In the biplot we also color the states according to their region (Northeast, South, North, Central, West). Which of the following statements are correct?

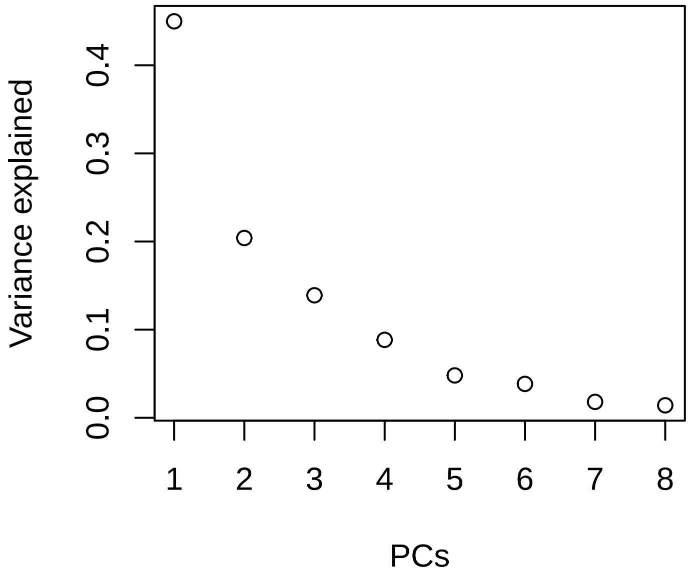
(i) [correct] Population, income and area contribute most to the second PC.
(ii) The first component explains 45% of the variability of the response variable.

(iii) [correct] California (CA) has a very high loading for the second principal component.

(iv) [correct] The first three PCs explain about 80% of the variability in the data.

(v) [correct] Illiteracy has a low loading on the second PC.




Variance explained
0.0
0.1
0.2
0.3
0.4
1
2
3
4
5
6
7
8
PCs