Front page

Department of Mathematical Sciences

Examination paper for **TMA4268 Statistical learning**

Academic contact during examination: Mette Langaas
Phone: 988 47 649

Examination date: 23 May 2019
Examination time (from-to): 09:00 - 13:00
Permitted examination support material: C.

- One yellow A5 sheet with your own handwritten notes (stamped by the Department of Mathematical Sciences),
- specified calculator.

Other information:

- All answers must be justified, and relevant calculations provided.
- Observe that the maximal score is given for each problem, so you should not spend too much time and too many words on problems that have a low maximal score.
- The exam questions are only available in English since this is a course given in English.
- This exam is given in Inspera.
- For each problem you may write your answer into Inspera, or use the provided paper sheets. Remember to correctly specify the code for each problem (available in lower right corner of the problem) on your paper sheet, so that the scanned sheets will be matched with the correct problem. It is advisable that you write down the code when starting to write on the sheet.

Students will find the examination results in Studentweb. Please contact the department if you have questions about your results. The Examinations Office will not be able to answer this.

1 Warming up with an overview

[Maximal score: 3 points]
Choose the correct alternative for each drop-down menu.

i) In this course we have considered two types of learning, A: [ ] (neural networks, classification, supervised, regression) and B: [ ] (support vector machines, unsupervised, logistic, crossvalidation). In A both covariates and a known response is present, but in B only [ ] (responses, covariates) are present. In A we have spent time studying two problem types: C [ ] (dimensionality reduction, clustering, regression, regularization) and [ ] (matrix algebra, covariance, classification, crossvalidation). Linear methods have been an important starting point for both problem types.

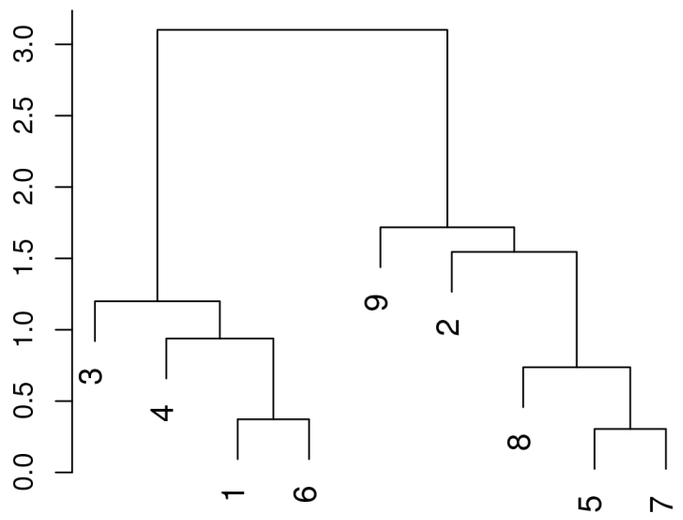
ii) In C we started with studying the relationship between one univariate response and one covariate in [ ] (K-nearest neighbour regression, simple linear regression, local regression, multiple linear regression), and then we added more covariates, and got [ ] (regression splines, principal component regression, multiple linear regression, simple linear regression). The least squares estimator was $\hat{\boldsymbol{\beta}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$, which was [ ] (unbiased, biased, skewed) and had a covariance matrix that was [ ] (dependent on the responses, dependent on the covariates, a constant) in the multiple linear regression model. We looked at including non-linear effects in the covariates by adding [ ] (linear transformations, principal components, polynomials). Non-linear effects were

also the topic of [ ] (ridge regression, K-nearest neighbour regression, support vector classifier, lasso regression). Interactions (the fact that the effect of one covariate on the response is dependent on the value of another covariate) were easily included in [ ] (regression trees, ridge regression, lasso regression, step functions). Model selection was performed using [ ] (bagging, boosting, the AIC penalty, linear discriminant analysis) and [ ] (ridge regression, principal component regression, splines, lasso regression). More about regression in the last part of this exam.

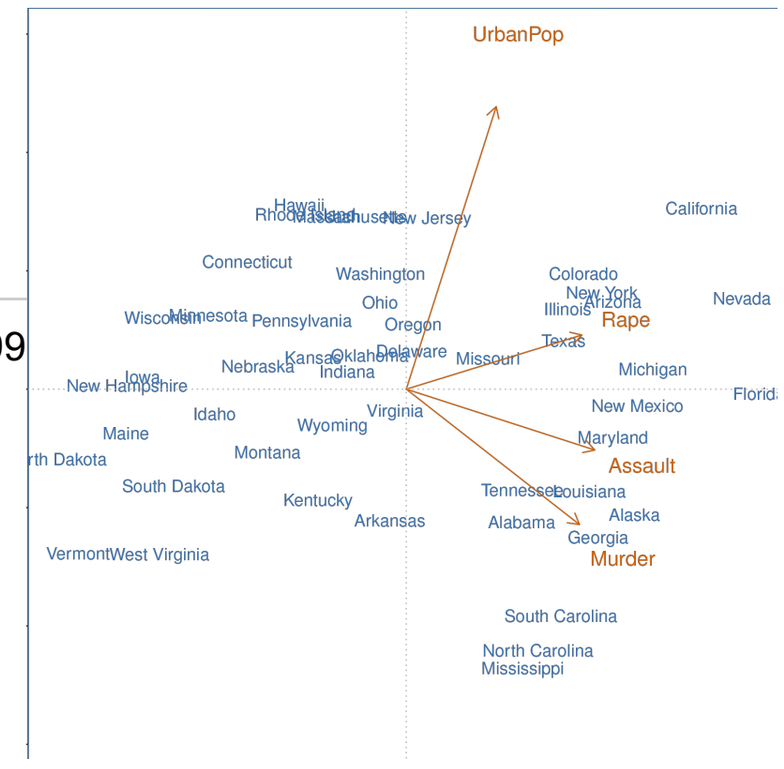
iii) The starting point for classification was the sampling and the [ ] (diagnostic, deviance, Gini, Bayes) paradigm. From the sampling paradigm we studied the [ ] (MMC, SVC and SVM, logistic regression, KNN, LDA and QDA), and from the other paradigm many more methods, which we will work with soon.

iv) In [ ] (crossvalidation, the bias-variance trade-off, supervised learning, unsupervised learning) the goal is to discover interesting aspects about the measurements $x_1, x_2, \ldots, x_p$ when y is not present. We have considered two types of methods: [ ] (principal component analysis, ridge regression, regularization, neural networks) and [ ] (lasso regression, clustering, R-programming, Python programming). Both methods have strong focus on finding [ ] (outliers, groups, residuals, errors) in the data, and on [ ] (the bias-variance trade-off, visualisation, crossvalidation, debugging).

The top figure on the right is an example of [ ] (hierarchical clustering, dimension reduction, response reduction, k-means clustering),



The bottom figure to the right gives a biplot from [ ] (ridge regression, k-means clustering, partial least squares, principal component analysis).



Maximum marks: 2.99

2 Match optimization criterion and method

[Maximal score: 2 points]
For each optimization criterion choose the correct method from the drop-down menu:

a) Maximize M by choosing $\beta_0, \beta_1, \ldots, \beta_p$ subject to $\sum_{j=1}^{p} \beta_j^2 = 1$ and $y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip}) \geq M$ for $i = 1, \ldots, n$

[ ] (least squares regression, lasso regression, logistic regression, support vector classifier, regression tree, maximal margin classifier, ridge regression)

b) $\mathbf{argmin}_{\boldsymbol{\beta}} \sum_{i=1}^{n} (y_i - \mathbf{x}_i^T \boldsymbol{\beta})^2$ [ ] (least squares regression, maximal margin classifier, support vector classifier, ridge regression, lasso regression, regression tree, logistic regression)

c) $\mathbf{argmin}_{\boldsymbol{\beta}} \sum_{i=1}^{n} (y_i - \mathbf{x}_i^T \boldsymbol{\beta})^2 + \lambda \sum_{j=1}^{p} \beta_j^2$

[ ] (least squares regression, ridge regression, lasso regression, maximal margin classifier, support vector classifier, regression tree, logistic regression)

d) $\mathbf{argmin}_{R_1(j,s), R_2(j,s)} \left[\sum_{i: x_i \in R_1(j,s)} (y_i - \hat{y}_{R_1})^2 + \sum_{i: x_i \in R_2(j,s)} (y_i - \hat{y}_{R_2})^2 \right]$

[ ] (least squares regression, ridge regression, lasso regression, regression tree, maximal margin classifier, support vector classifier, logistic regression)

e) $\mathbf{argmin}_{\boldsymbol{\beta}} \sum_{i=1}^{n} (y_i - \mathbf{x}_i^T \boldsymbol{\beta})^2 + \lambda \sum_{j=1}^{p} |\beta_j|$

[ ] (least squares regression, lasso regression, ridge regression, maximal margin classifier, regression tree, support vector classifier, logistic regression)

f) Maximize M by choosing $\beta_0, \beta_1, \ldots, \beta_p, \varepsilon_1, \ldots, \varepsilon_n$ subject to $\sum_{j=1}^{p} \beta_j^2 = 1$ and $y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip}) \geq M(1 - \varepsilon_i)$ where $\varepsilon_i \geq 0$ for all $i = 1, \ldots, n$ and $\sum_{i=1}^{n} \varepsilon_i \leq C$

[ ] (least squares regression, regression tree, ridge regression, lasso regression, maximal margin classifier, support vector classifier, logistic regression)

Maximum marks: 2.04

3 Neural network – mathematical formula

[Maximal score: 4 points]

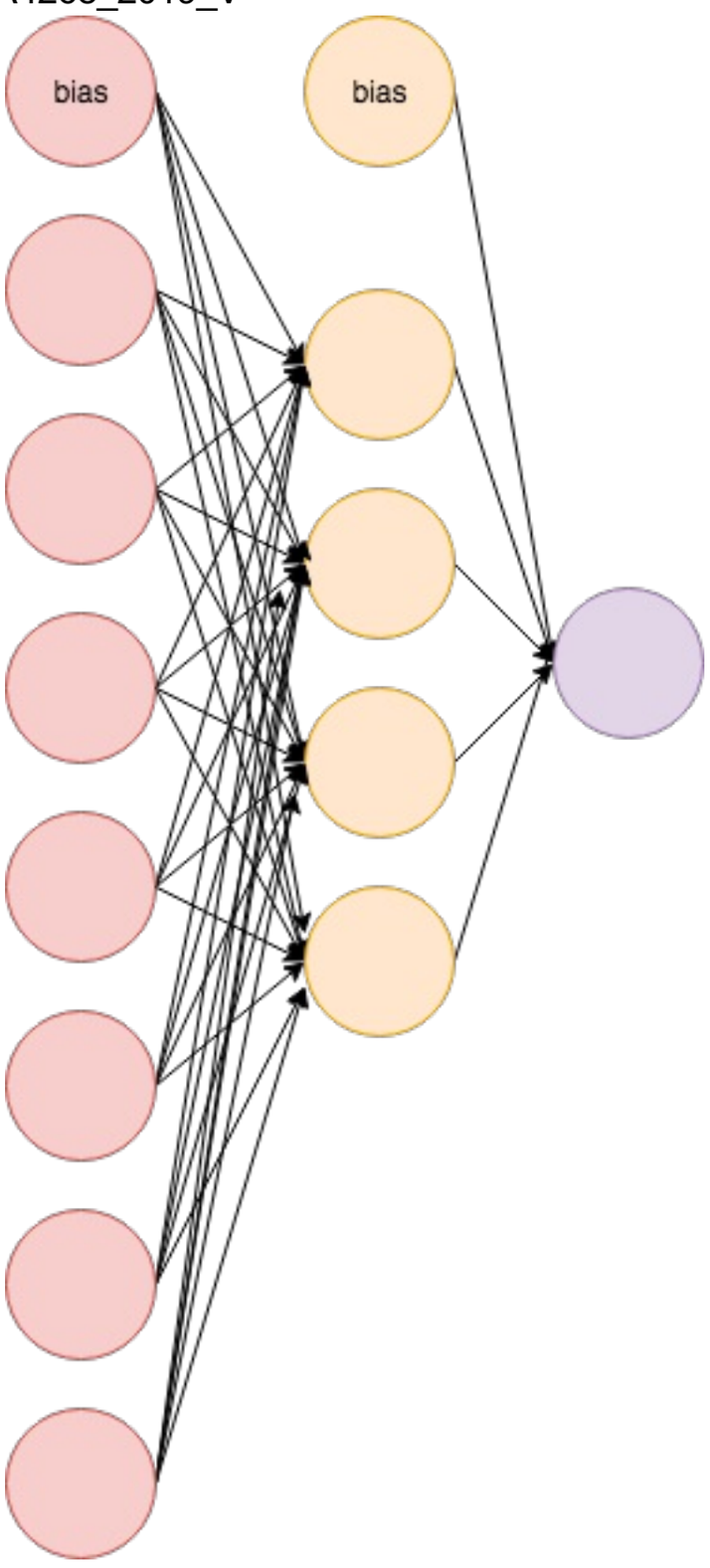
a) Write down the mathematical formula for a feedforward neural network with one input layer, one hidden layer and one output layer, with the following architecture

- input layer: seven input nodes
- one bias node for the hidden layer,
- hidden layer: four nodes,
- one bias node for the output layer,
- output layer: one node.

Let the nodes in the hidden layer have sigmoid activiation function and the node in the output layer have linear activiation function. A drawing of the feedforward neural network is given below.

b) Which type of problem can this network solve?

c) What would be an appropriate loss function for this network (give mathematical formula)?



Fill in your answers here, and/or use the paper sheets provided.

Format

Words: 0

Maximum marks: 4

4 Neural network – understanding and estimation

[Maximal score: 6 points]

a) Which feedforward neural network architecture and activation functions does this formula give? Include a drawing of the network architecture.

$$\hat{y}_1(\mathbf{x}) = [1 + \exp(-\beta_{01} - \sum_{m=1}^{2} \beta_{m1} \max(\gamma_{0m} + \sum_{l=1}^{3} \gamma_{lm} \max(\alpha_{0l} + \sum_{j=1}^{4} \alpha_{jl} x_j, 0), 0))]^{-1}$$

b) How many parameters are estimated in this network?

c) Which type of problem can this network be used to solve, and what would you use as loss function (give mathematical formula)?

d) Discuss briefly how parameters nowadays are estimated in a feedforward neural network.

Fill in your answer here and/or use the sheets provided.

Format

Words: 0

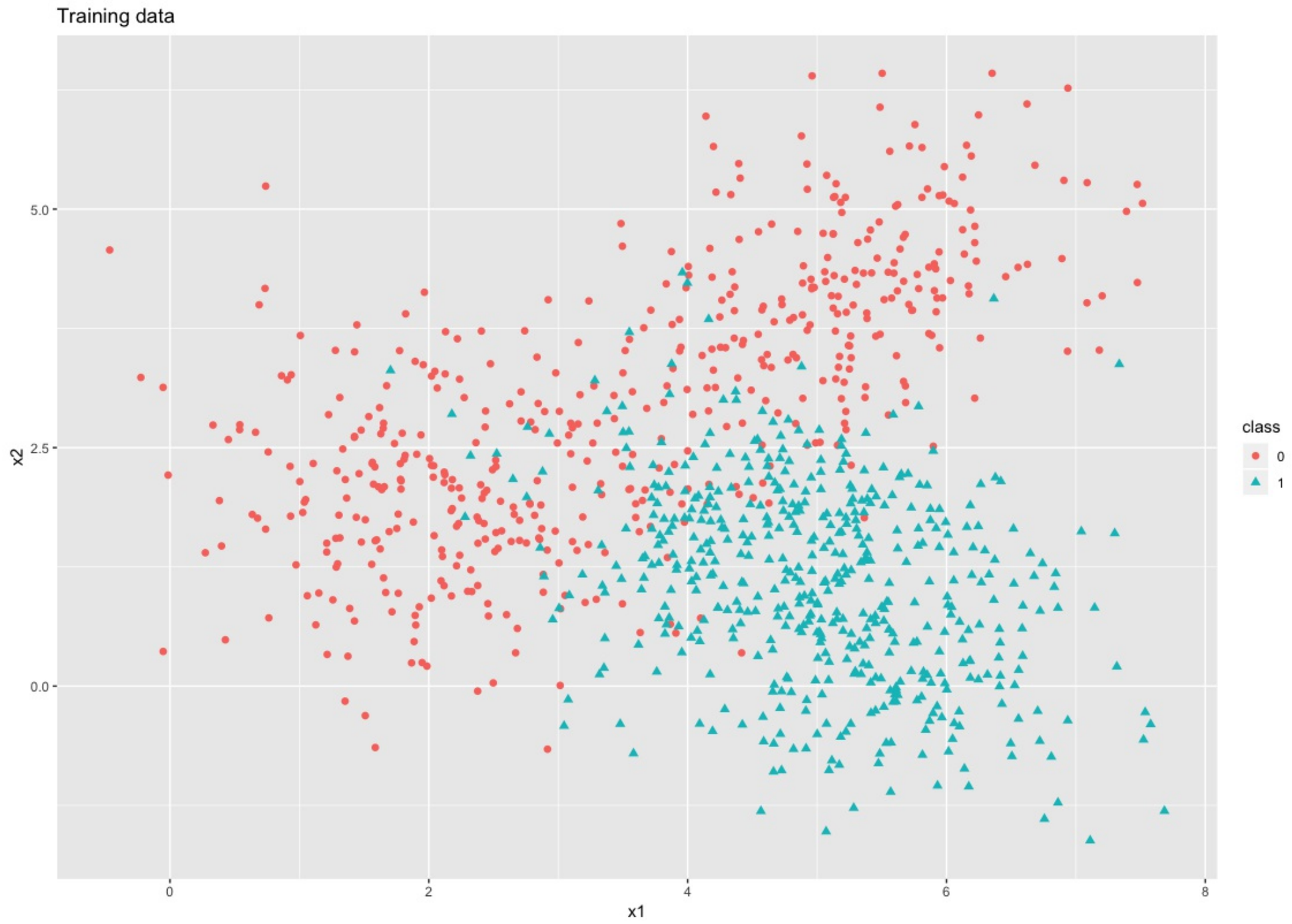
Maximum marks: 6

5 Classification with K-nearest neighbour

Comment: We will now have 5 problems on classification, that use the same data set-up.

We will study a simulated data set with two classes (labelled 0 and 1) and two numerical covariates x_1 and x_2. Let $\mathbf{x} = (x_1, x_2)^T$ be a column vector with the two covariates.

A training set with 500 observations of each class is available, and a scatter plot is given below.



[Maximal score: 3 points]

a) Write down the mathematical formula for the probability of class 1 for the K-nearest neighbour classifier at a covariate value $\mathbf{x_0}$, and explain your notation.

b) What does the rule *majority vote* mean when applied to K -neareste neighbour classification? In our situation with two classes, what is the cut-off on the probability of class 1 equivalent of a majority vote?

c) List one positive and one negative aspect of using K -nearest neighbour classification.

Fill in your answers here, and/or use the paper sheets provided.

Format

Words: 0

Maximum marks: 3

6 Class boundaries for K-nearest neighbour

[Maximal score: 2 points]

In the figure to the left we have used the K-nearest neighbour classification method, with K equal to $1, 3, 25$ and 199, to get a classification boundary between class 0 and 1 based on the training data. (To make the boundary stand out the training data are not shown in the plots.)

Match panels A–D to values of $K(1, 3, 25, 199)$ in the drop-down menus.

Panel A: (1, 3, 25, 199)

Panel B: (1, 3, 25, 199)

Panel C: (1, 3, 25, 199)

Panel D: (1, 3, 25, 199)

Maximum marks: 2

7 Choosing K in K-nearest neighbour classification

[Maximal score: 5 points]

Crossvalidation can be used to choose a good value for K, and is performed in the R-code and print-out shown in the pdf-file on the left.

a) Explain shortly what is done, and what the results are. Your explanation should include a drawing and the words: fold and error measure.

b) Which value of K (number of neighbours) would you choose? Elaborate.

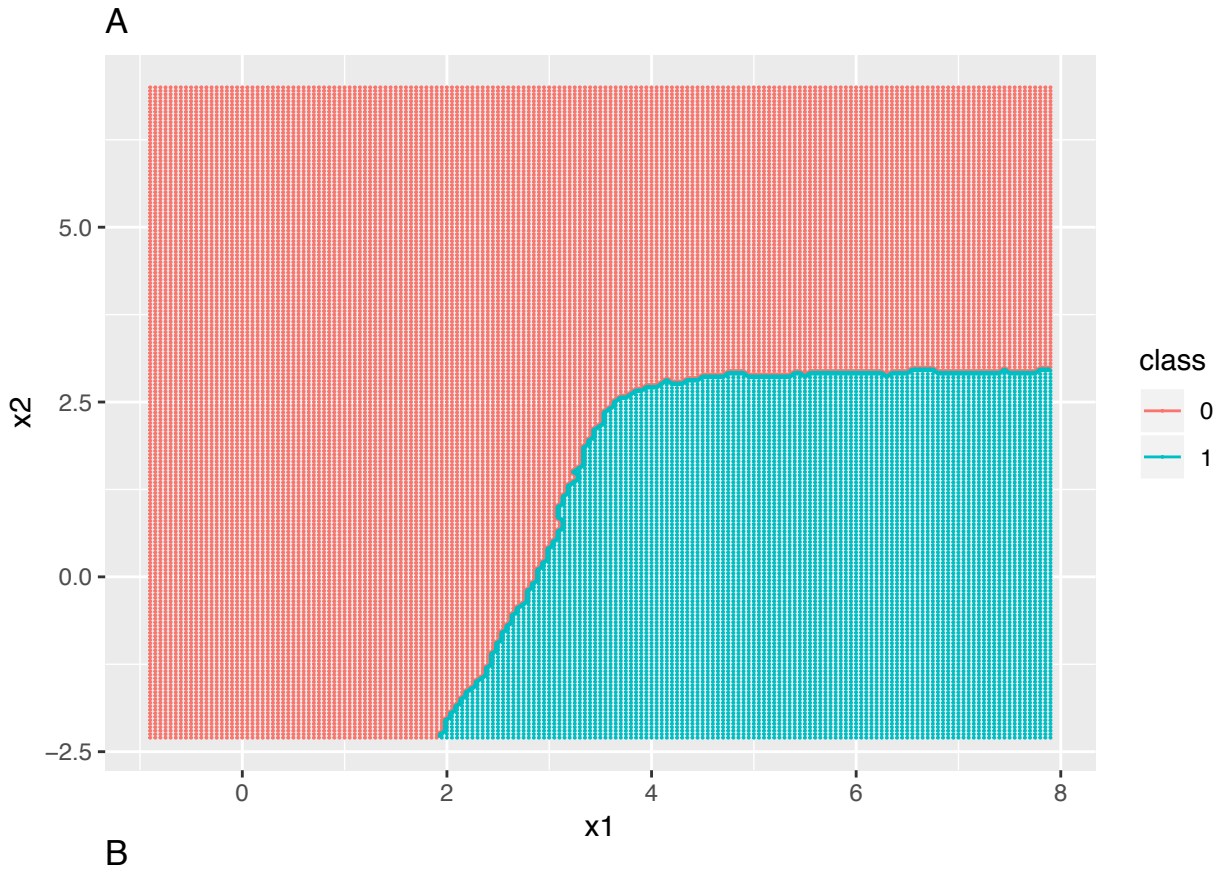
c) What is the one-standard-error-rule? Can you from the print-out use this rule to choose K? If not, what can be added or changed in the R-code to be able to use this rule? You are not asked to rewrite the R-code.

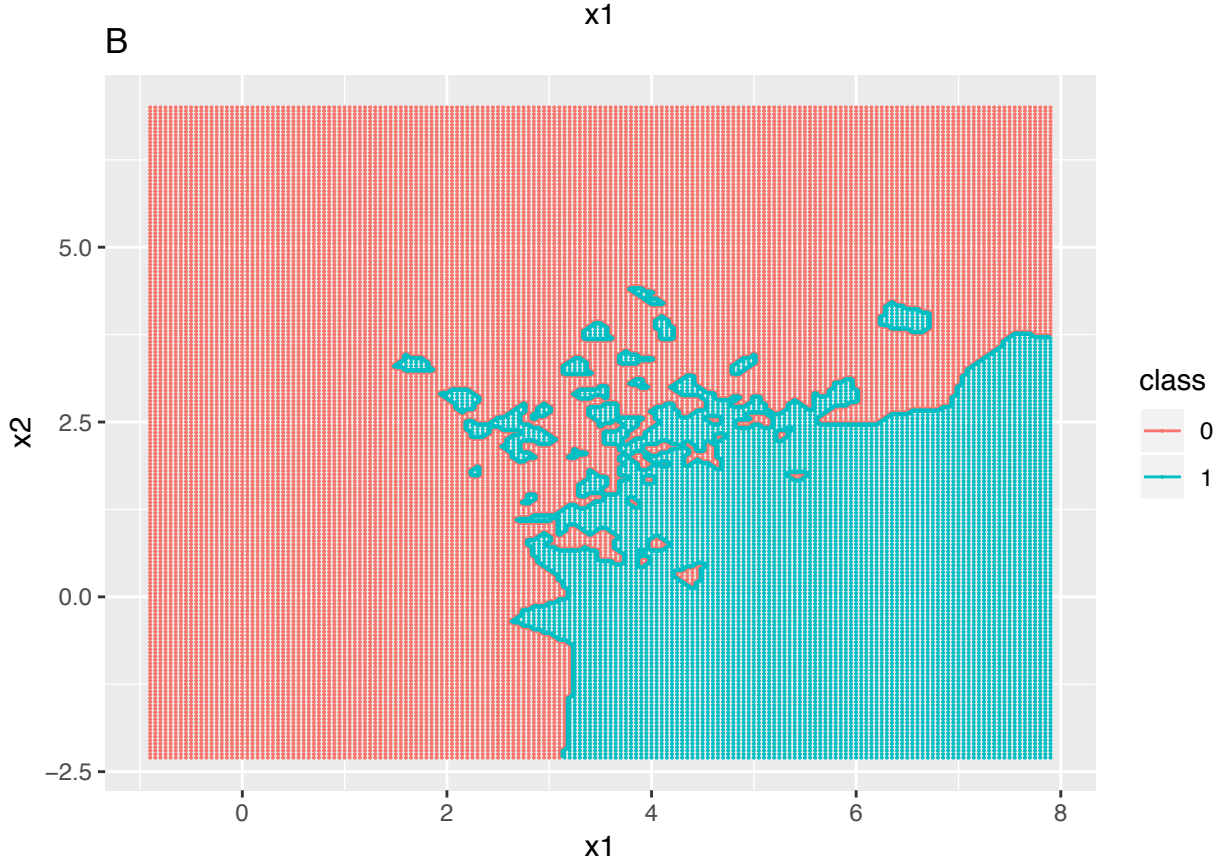
Fill in your answer here and/or use the paper sheets provided.

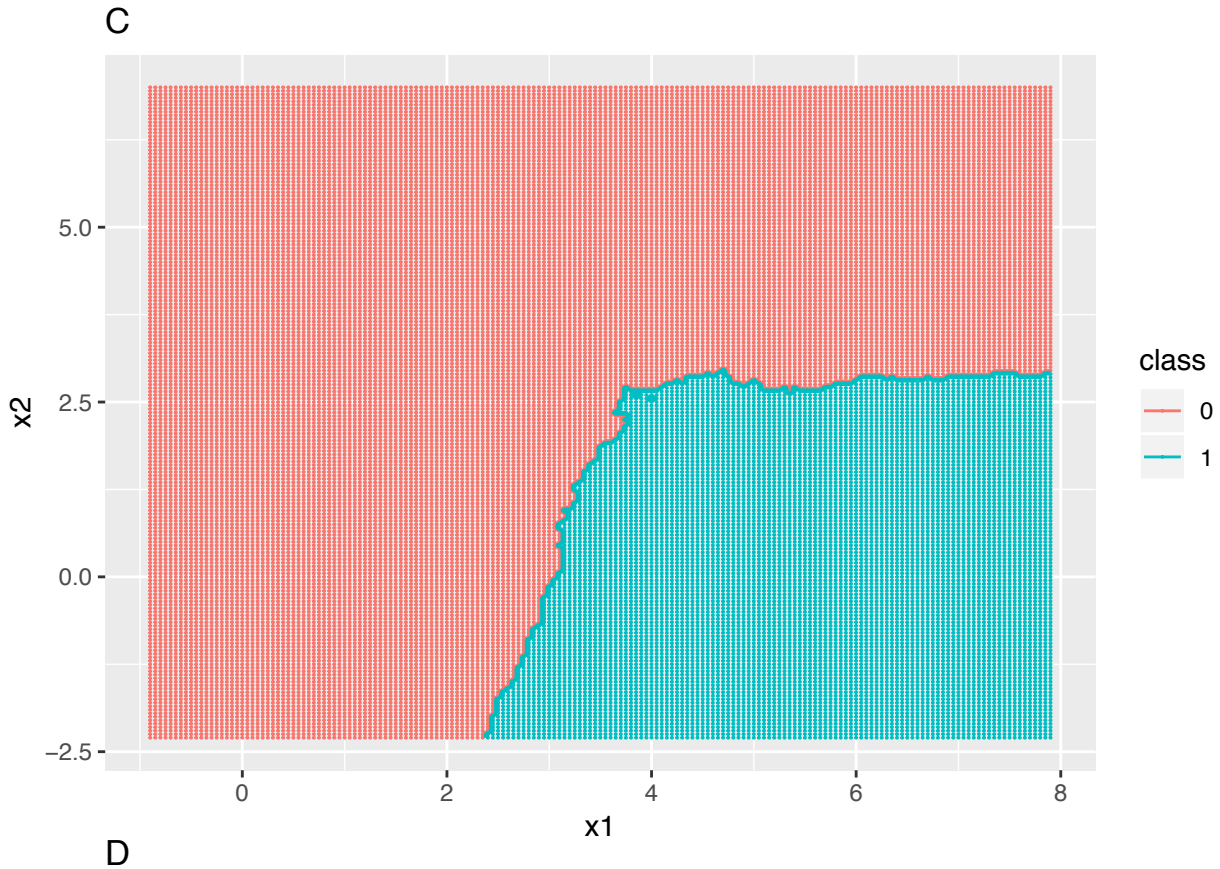
Format

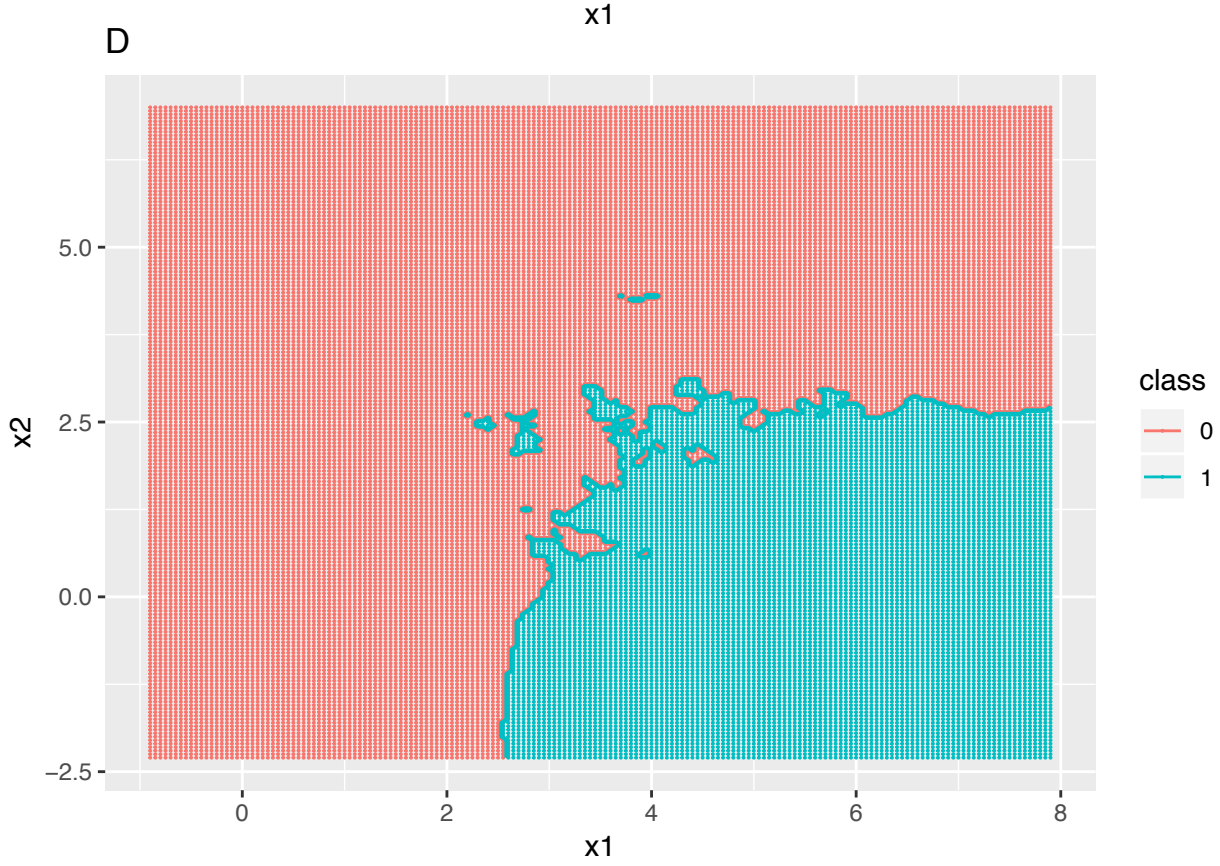
Words: 0

Maximum marks: 5


A
5.0
2.5
0.0
−2.5
x2
0
2
4
6
8
x1
class
0
1


B
5.0
2.5
0.0
−2.5
x2
0
2
4
6
8
x1
class
0
1


C
5.0
2.5
0.0
−2.5
x2
0
2
4
6
8
x1
class
0
1


D
5.0
2.5
0.0
−2.5
x2
0
2
4
6
8
x1
class
0
1

Choosing K in K-nearest neighbour classification

```
# the training data contains 500 observations from each class
# train.x: a 1000 times 2 matrix with training covariates x1 and x2
# train.y: a 1000 vector with the class (0 or 1) as a factor for the training data

Kgrid=seq(1,199,by=2)
Kgrid
```

```
##   [1]   1   3   5   7   9  11  13  15  17  19  21  23  25  27  29  31  33
##  [18]  35  37  39  41  43  45  47  49  51  53  55  57  59  61  63  65  67
##  [35]  69  71  73  75  77  79  81  83  85  87  89  91  93  95  97  99 101
##  [52] 103 105 107 109 111 113 115 117 119 121 123 125 127 129 131 133 135
##  [69] 137 139 141 143 145 147 149 151 153 155 157 159 161 163 165 167 169
##  [86] 171 173 175 177 179 181 183 185 187 189 191 193 195 197 199
```

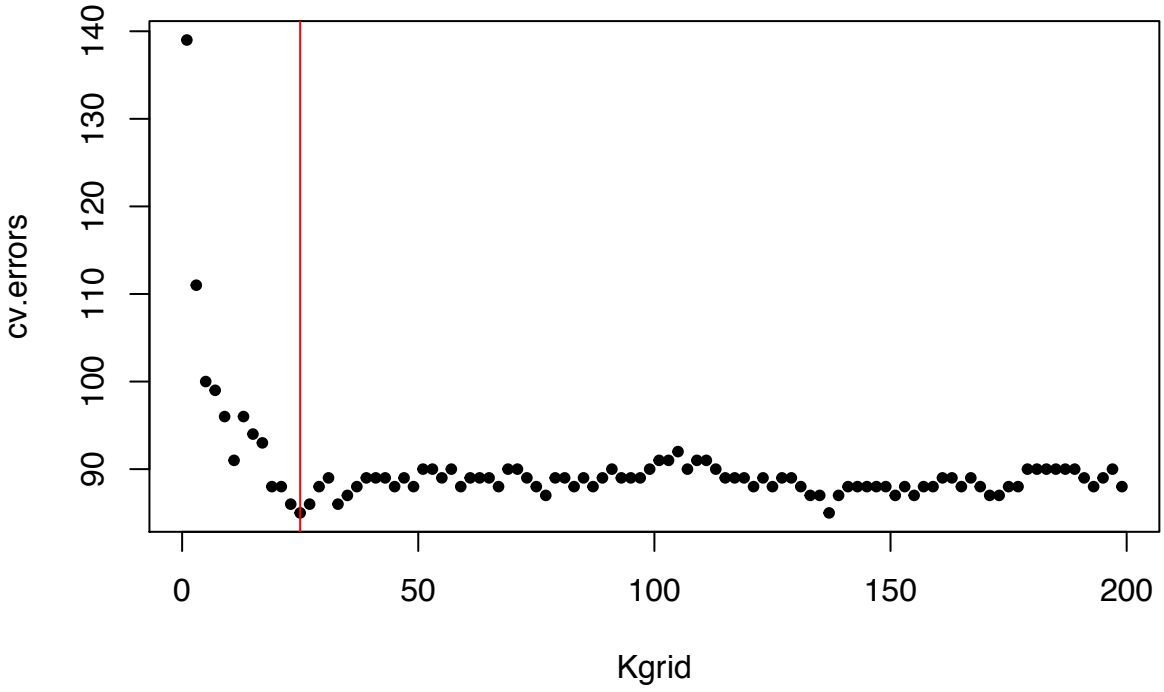
```
nmodels=length(Kgrid)
nfolds=5

set.seed(4268)
folds <- createFolds(train.y,k=nfolds)
# the random sampling is done within the levels of y when y is a factor
# in an attempt to balance the class distributions within the splits.
# folds[[1]] gives indecies for training observations in fold 1,
# folds[[2]] the same for fold 2 etc.

cv.errors=rep(0,nmodels)

for(i in 1:nfolds)
{
  for(j in 1:nmodels)
  {
     pred=class:::knn(train=train.x[-folds[[i]],],
                      test=train.x[folds[[i]],],
                      cl=train.y[-folds[[i]]], k=Kgrid[j])
     # gives the predicted class 0/1 for the validation fold
     cv.errors[j]=cv.errors[j]+sum(pred!=train.y[folds[[i]]])
  }
}
cv.errors
```

```
##   [1] 139 111 100  99  96  91  96  94  93  88  88  86  85  86  88  89  86
##  [18]  87  88  89  89  89  88  89  88  90  90  89  90  88  89  89  89  88
##  [35]  90  90  89  88  87  89  89  88  89  88  89  90  89  89  89  90  91
##  [52]  91  92  90  91  91  90  89  89  89  88  89  88  89  89  88  87  87
##  [69]  85  87  88  88  88  88  88  87  88  87  88  88  89  89  88  89  88
##  [86]  87  87  88  88  90  90  90  90  90  90  89  88  89  90  88
```

```
Kgrid[which.min(cv.errors)]
```

```
## [1] 25
```

```
plot(x=Kgrid,y=cv.errors,pch=20)
abline(v=25,col=2)
```


cv.errors
90
100
110
120
130
140
0
50
100
150
200
Kgrid

8 Model evaluation on test set with 3-nearest neighbour classification

[Maximal score: 10 points]

A test set is available, with 500 observations from each of the two classes. We will study classifiation at a value $\mathbf{x}_0$ in the test set.

a) For 3-nearest neighbour classification (that is, K-nearest neighbour classification with $K = 3$) what are the possible values for $P(Y = 1 \mid \mathbf{x}_0)$?

b) A classification rule may be to classify to class 1 if $P(Y = 1 \mid \mathbf{x}_0) \geq \frac{1}{2}$. For our test set this classification rule gave the following confusion matrix.

	Predicted 0	Predicted 1	Total
True class 0	464	36	500
True class 1	50	450	500

Calculate the sensitivity and the specificity and explain what the two numbers mean.

c) What is a reciever operating characteristic (ROC) curve, and what can such a curve tell us?

d) Below you find a table with the number of observations in the test set where different predicted probabilities of class 1 is as given in the column headers, and the true class is given in the rows. To be clear, from the table you may read that there are 391+14 observations in the test data where the predicted probability of class 1 is 0, and 391 of these have true class 0 and 14 has true class 1. Further, there are 22+54 observation where the predicted probability of class 1 is 2/3, and 22 of these have true class 0 and 54 have true class 1.

Probability of class 1	0	1/3	2/3	1	Total
True class 0	391	73	22	14	500
True class 1	14	36	54	396	500

Draw a ROC curve based on the information given in the table above. Include the calculations you have made in your answer. In your drawing of the ROC curve the numerical values you calculated must be clearly visible.

Fill in your answer here and/or use the paper sheet provided.

Format

Words: 0

Maximum marks: 10

9 Bayes decision rule

[Maximal score: 10 points]

a) What is a Bayes classifier, Bayes decision boundary and Bayes error rate?

In our classification problem the training and test data sets were simulated as follows:

- Prior class probabilities:$\pi_0 = P(Y = 0) = 0.5$ and $\pi_1 = P(Y = 1) = 0.5.$
- Class conditional probabilites:
  - $f_0(\mathbf{x}) = 0.5 \cdot \frac{1}{2\pi|\Sigma_{01}|}\exp[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_{01})^T\Sigma_{01}^{-1}(\mathbf{x}-\boldsymbol{\mu}_{01})] + 0.5 \cdot \frac{1}{2\pi|\Sigma_{02}|}\exp$ $[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_{02})^T\Sigma_{02}^{-1}(\mathbf{x}-\boldsymbol{\mu}_{02})]$
  - $f_1(\mathbf{x}) = \frac{1}{2\pi|\Sigma_1|}\exp[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_1)^T\Sigma_1^{-1}(\mathbf{x}-\boldsymbol{\mu}_1)]$
- with mean vectors $\mu_{01} = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$, $\mu_{02} = \begin{bmatrix} 5 \\ 4 \end{bmatrix}$ and $\mu_1 = \begin{bmatrix} 5 \\ 1 \end{bmatrix}$, and
- covariance matrices $\Sigma_{01} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $\Sigma_{02} = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}$, and $\Sigma_1 = \begin{bmatrix} 1 & -0.3 \\ -0.3 & 1 \end{bmatrix}$

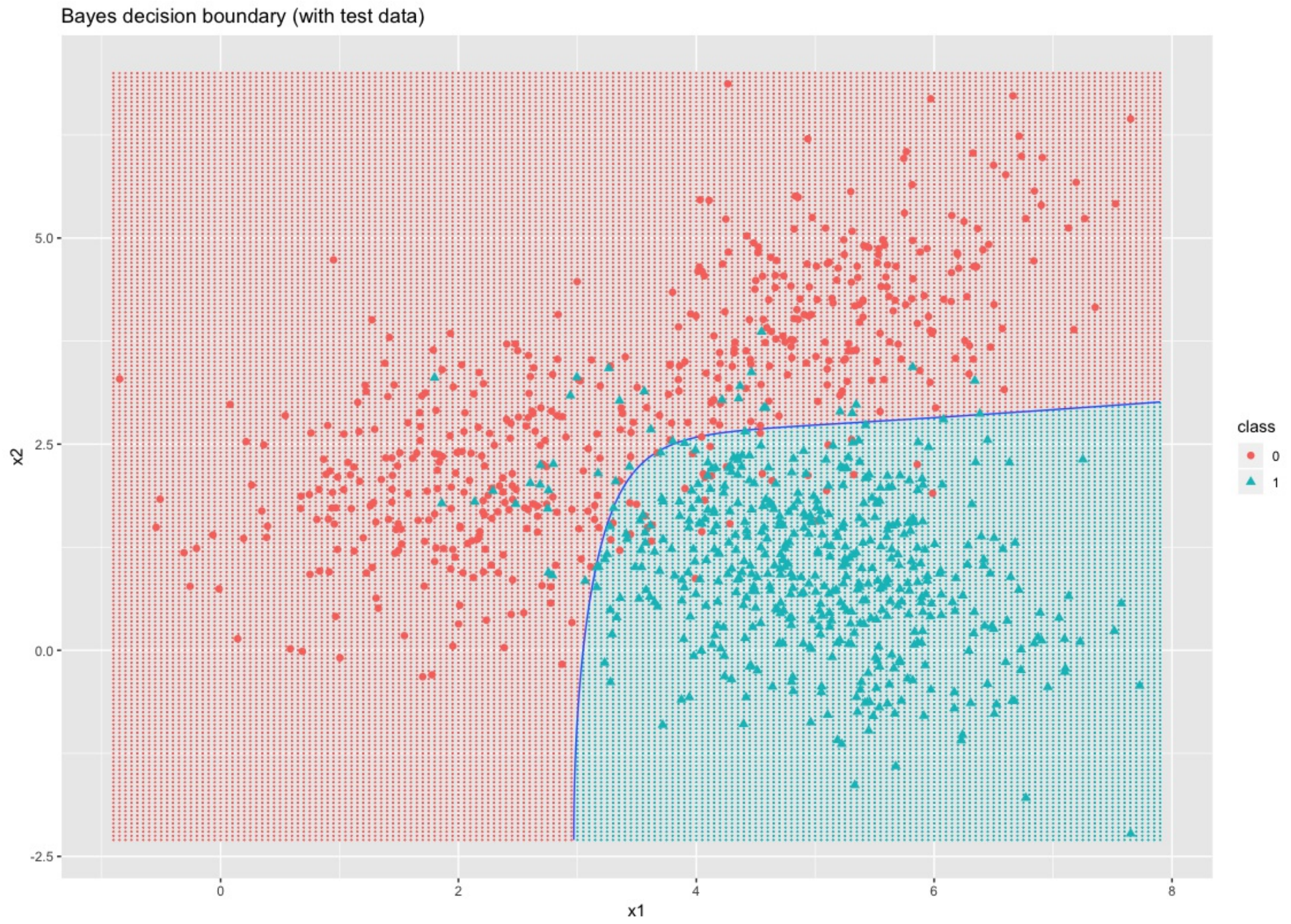
b) How would you proceed to find the Bayes decision boundary? Write down the equation to be solved, explain what are the unknowns. You are *not* asked to solve the equation.

c) The Bayes decision boundary is shown as a (blue) curve in the plot below. The Bayes error rate was found (by simulation) to be 8%. Explain what this means.

d) What would happen to the Bayes decision boundary and Bayes error rate if we change the prior probabilities of the two classes to $\pi_0 = 0.9$ and $\pi_1 = 0.1$? Elaborate.

e) The methods we have considered in this course are K—nearest neighbour classification, linear discriminant analysis, quadratic discriminant analysis, logistic regression, classification tree, bagged trees, random forest, maximal margin classifier, support vector classifier, support vector machine, and feedforward neural network.

Now that you know what the Bayes decision boundary looks like, which of the classification methods (select only one method) do you think would provide the best solution to our classification problem? Justify your answer.



Fill in your answer here and/or use the paper sheets provided.

Format

Words: 0

Maximum marks: 10

10 Regression with bike rentals

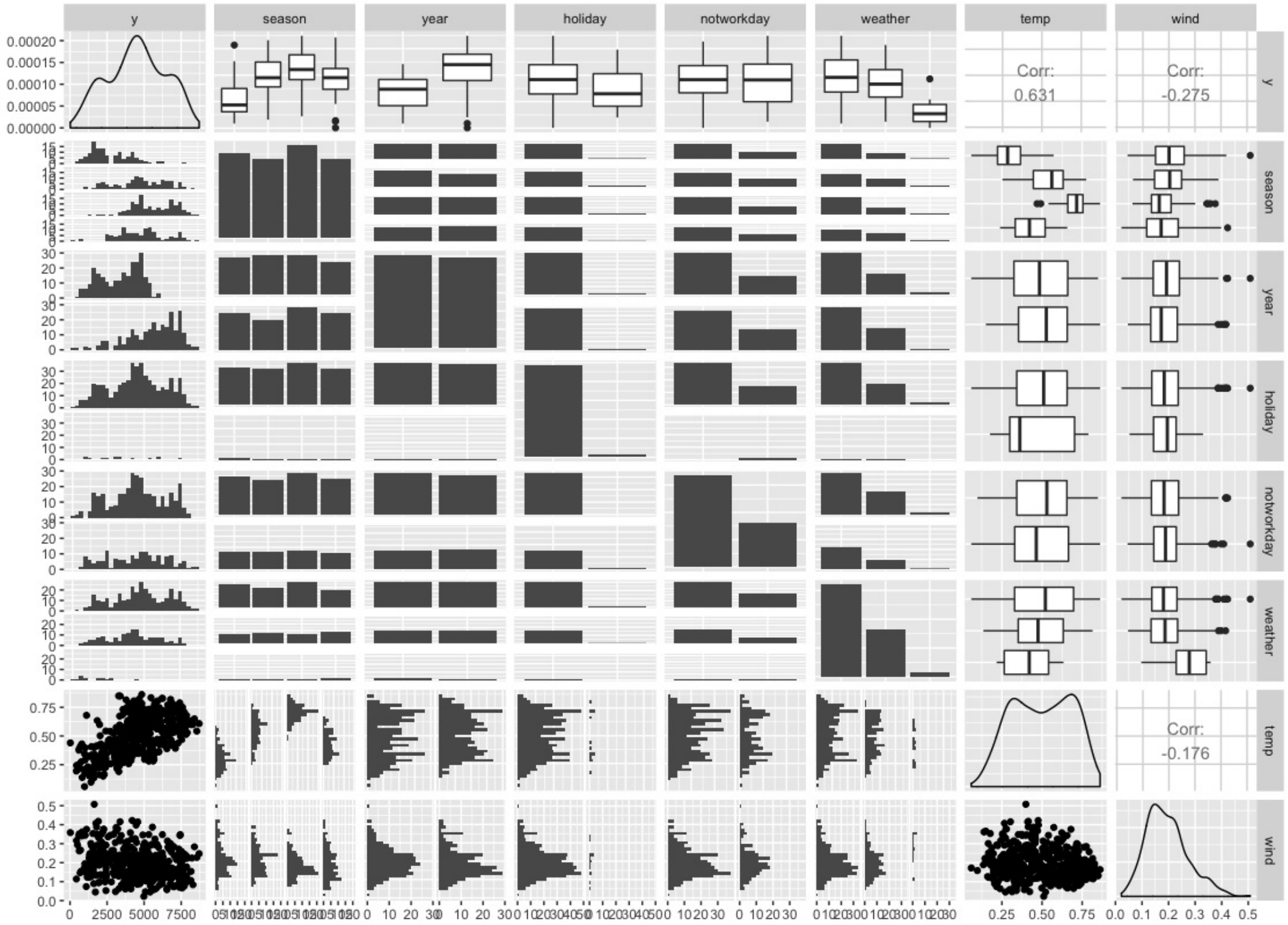
Comment: We will now have 4 problems on regression with the same data set-up.

We will look at a data set of daily counts of bike rentals in a bike sharing system in Washington DC in the first two years of operation, matched with data on climate, season and type of day.

(Source: http://archive.ics.uci.edu/ml/datasets/bike+sharing+dataset)

- y: daily count of number of bike rentals (our response)
- year: (0: 2011, 1: 2012)
- season: factor with four levels (1: spring, 2: summer, 3: autumn, 4: winter)
- holiday: (0: not holiday, 1: holiday)
- notworkday: (0: neither weekend nor holiday, 1: weekend or holiday)
- weather: factor with three levels (1: clear to partly cloudy, 2: mist and/or clouds, 3: snow, rain, thunderstorm)
- temp: normalized temperature in Celsius
- wind: normalized wind speed

The data was divided randomly into a training set of size 500 and a test set of size 231.
The training data is presented in the pairs-plot below.



Smoothing spline and polynomials
[Maximal score: 6 points]

We use the daily count of number of bike rentals (y) as response and temperature (temp) as the only covariate.

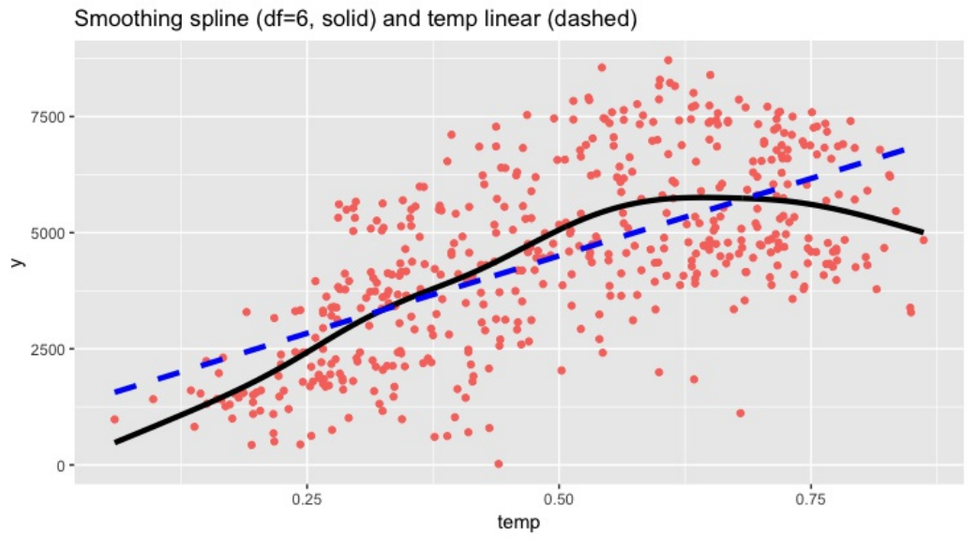
a) What is the idea behind a smoothing spline, and which criterion is the smoothing spline minimizing?
What is the connection between a smoothing spline and a cubic spline?

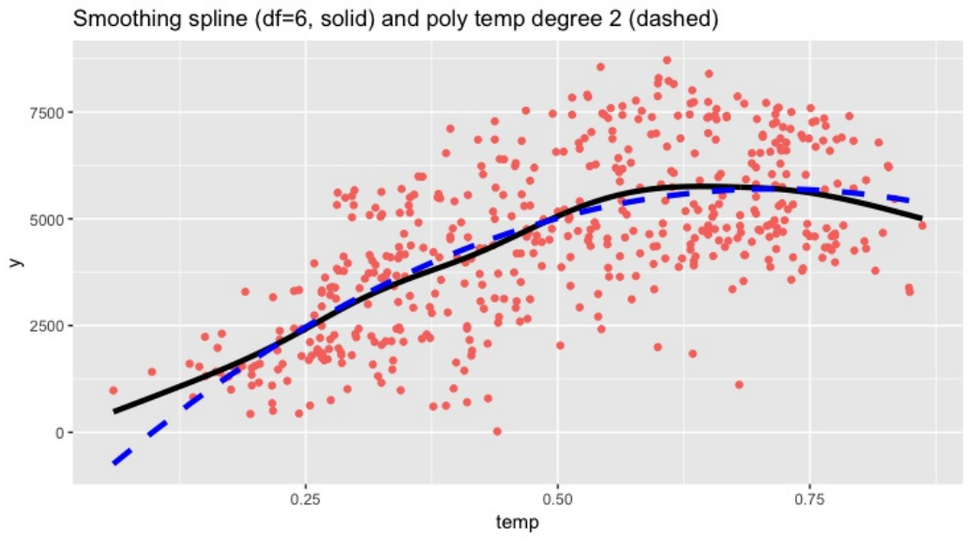
A total of five curves were fitted to the training data:

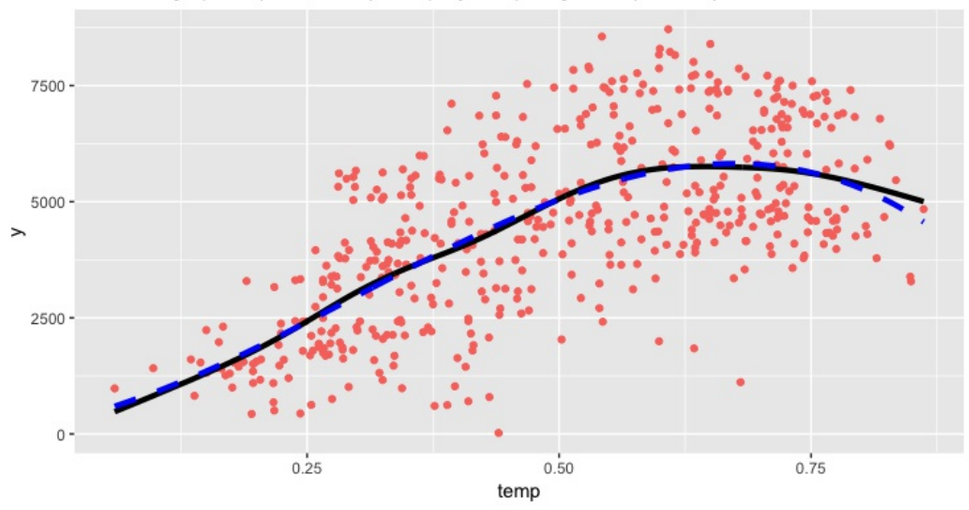
- a smoothing spline with 6 effective degrees of freedom (solid, black)
- polynomial of degrees 1, 2, 3 and 4 (dashed, blue)

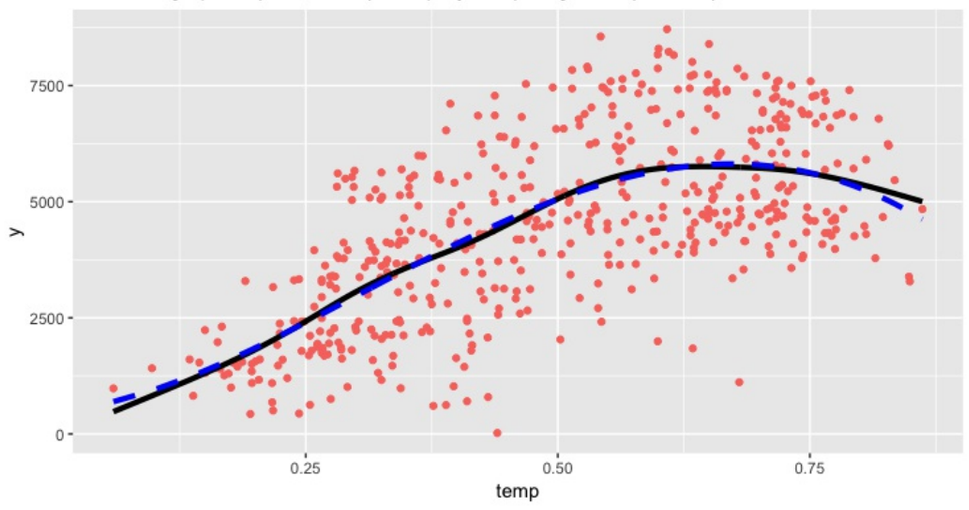
See the results in the four panels below.

b) What does it mean that the smoothing spline has 6 effective degrees of freedom?
How would you evaluate the different model fits in the four panels?
We will next fit a multiple linear regression with all available covariates. What is your suggestion for how to handle the temp covariate?









Fill in your answer here and/or use the paper sheets provided.

Format

Words: 0

Maximum marks: 6

11 Multiple linear regression

[Maximal score: 4 points]

A multiple linear regression was fitted to the data, with linear effects of all covariates, and in addition also a quadratic and a cubic effect of temperature. Dummy variable coding (treatment contrast) was used for the factors, with the lowest level (season=1 and weather=1) as the reference category. A pdf with R-code and print-out is found in the left panel.

a) Write down the mathematical formula for the regression on matrix form. What is the dimension of the design matrix? Write out the row of the design matrix for an observation with

- year=1
- season=1
- holiday=0
- notworkday=0
- weather=3
- temp=0.3
- wind=0.2

b) What are the assumptions made for the error term in a normal multiple linear regression model? Why might we think that the response, which is a count, can be modelled to be normally distributed? Comment briefly on the two residual plots.

Fill in your answer here and/or use the paper sheets provided.

Format

Words: 0

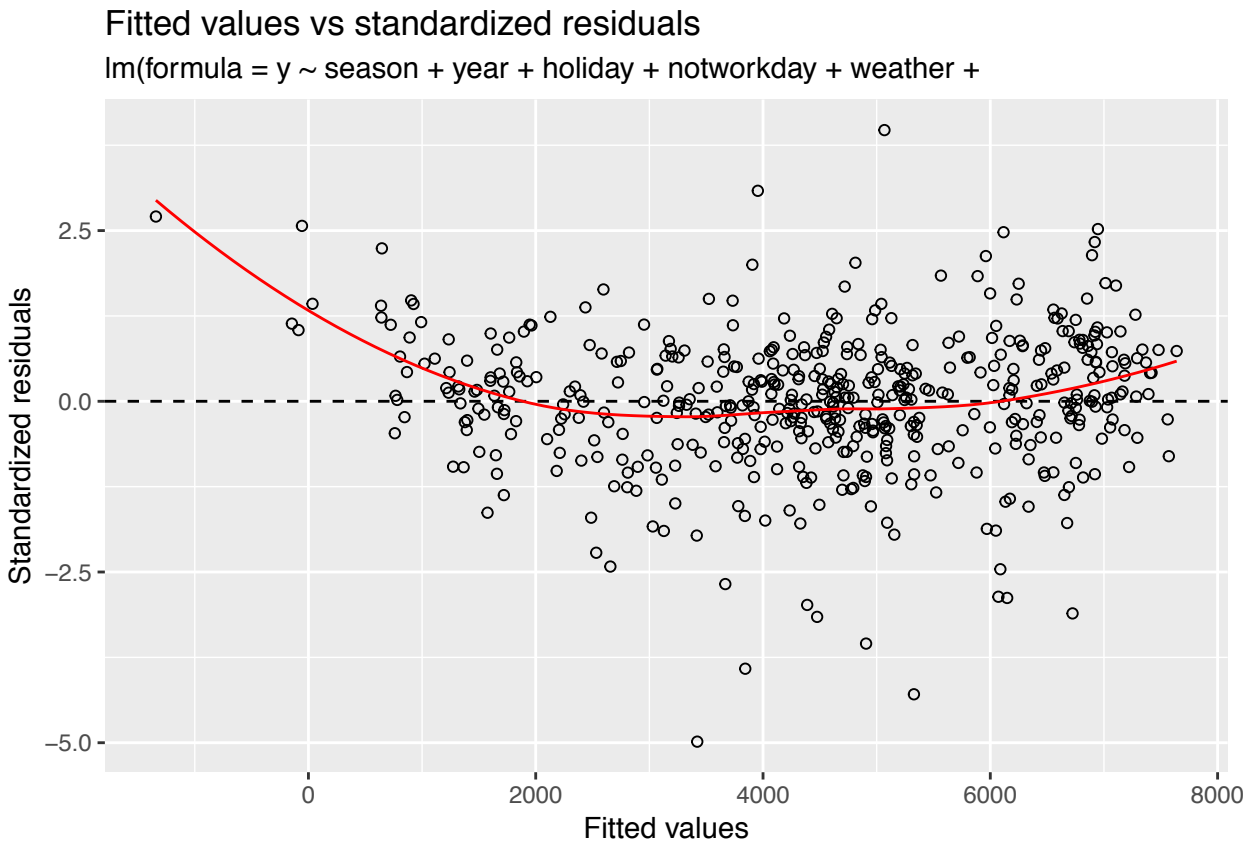
Maximum marks: 4

Multiple linear regression

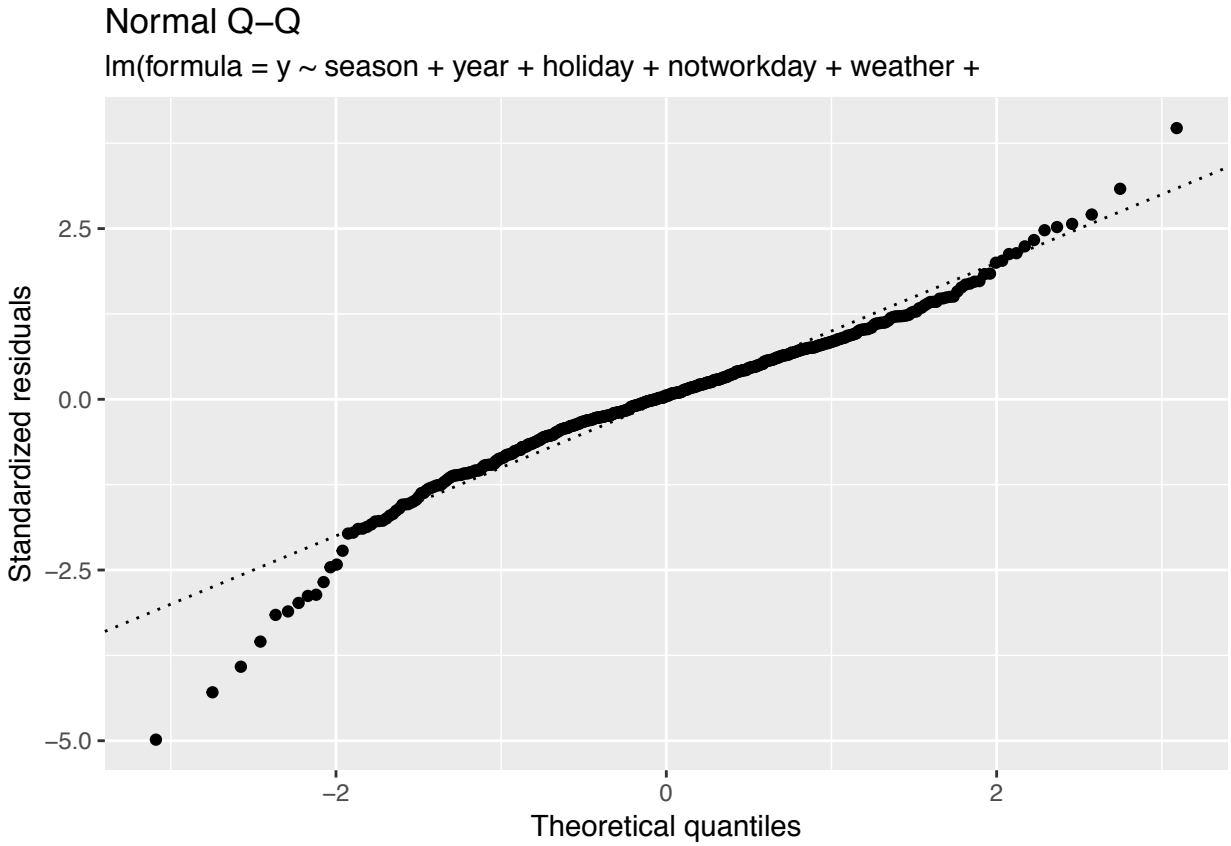
```
fitlm=lm(y~season+year+holiday+notworkday+weather+temp+I(temp^2)+I(temp^3)+wind,data=train)
# I(temp^2) just means temp^2, but in a model formula this extra I() is needed
summary(fitlm)
```

```
##
## Call:
## lm(formula = y ~ season + year + holiday + notworkday + weather +
##     temp + I(temp^2) + I(temp^3) + wind, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3399.3  -344.9    36.9   429.2  2767.6
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2902.20     481.36   6.029 3.26e-09 ***
## season2        828.76     121.20   6.838 2.41e-11 ***
## season3       1093.71     155.38   7.039 6.62e-12 ***
## season4       1286.18     104.99  12.251  < 2e-16 ***
## year          2052.39      64.82  31.663  < 2e-16 ***
## holiday       -479.99     199.13  -2.410   0.0163 *
## notworkday    -100.66      70.22  -1.434   0.1524
## weather2      -705.99      69.99 -10.087  < 2e-16 ***
## weather3     -2495.95     186.56 -13.379  < 2e-16 ***
## temp        -14643.01    3430.61  -4.268 2.37e-05 ***
## I(temp^2)    57717.61    7464.37   7.732 6.10e-14 ***
## I(temp^3)   -47873.21    5071.47  -9.440  < 2e-16 ***
## wind         -2726.69     425.89  -6.402 3.61e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 709.4 on 487 degrees of freedom
## Multiple R-squared:  0.8707, Adjusted R-squared:  0.8675
## F-statistic: 273.2 on 12 and 487 DF,  p-value: < 2.2e-16
```

```
ggplot(fitlm, aes(.fitted, .stdresid)) + geom_point(pch = 21) + geom_hline(yintercept = 0,
    linetype = "dashed") + geom_smooth(se = FALSE, col = "red", size = 0.5,
    method = "loess") + labs(x = "Fitted values", y = "Standardized residuals",
    title = "Fitted values vs standardized residuals", subtitle = deparse(fitlm$call))
```




```
ggplot(fitlm, aes(sample = .stdresid)) + stat_qq(pch = 19) + geom_abline(intercept = 0,
    slope = 1, linetype = "dotted") + labs(x = "Theoretical quantiles",
    y = "Standardized residuals", title = "Normal Q-Q", subtitle = deparse(fitlm$call))
```




```
library(nortest)
ad.test(rstudent(fitlm))
```

```
##
##  Anderson-Darling normality test
##
## data:  rstudent(fitlm)
## A = 3.9514, p-value = 7.525e-10
```

```
trainerrorlm=mean((predict(fitlm)-train$y)^2)
predlm = predict(fitlm, newdata=test)
testerrorlm=mean((predlm-test$y)^2)
c(trainerrorlm,testerrorlm)
```

```
## [1] 490198.7 568906.1
```

12 Ridge regression

[Maximal score: 10 points]

A ridge regression was fitted to the bike rental data, and R-code and print-out is found in the pdf-file in the left panel.

a) Write down the optimization problem for the ridge regression, and illustrate with a drawing.
How was the penalty parameter chosen in the analysis in the pdf-file? What was the optimal value?

b) Write down the formula for the ridge regression estimator, and derive the mean vector and variance-covariance matrix for the estimator. Specify the assumptions you make and comment on your findings.

c) In ridge regression regularization is performed by adding a penalty term to a loss function. Why would we want to add a penalty term to the loss function? What is the connection between regularization and the bias–variance trade–off?

d) In the bottom of the pdf-file there is a comparison between the estimated parameters of the multiple linear regression and the ridge regression. Comment on the comparison. Would you prefer to use multiple linear regression or ridge regression in the analysis of the bike rental data?

Fill in your answer here and/or use the paper sheets provided.

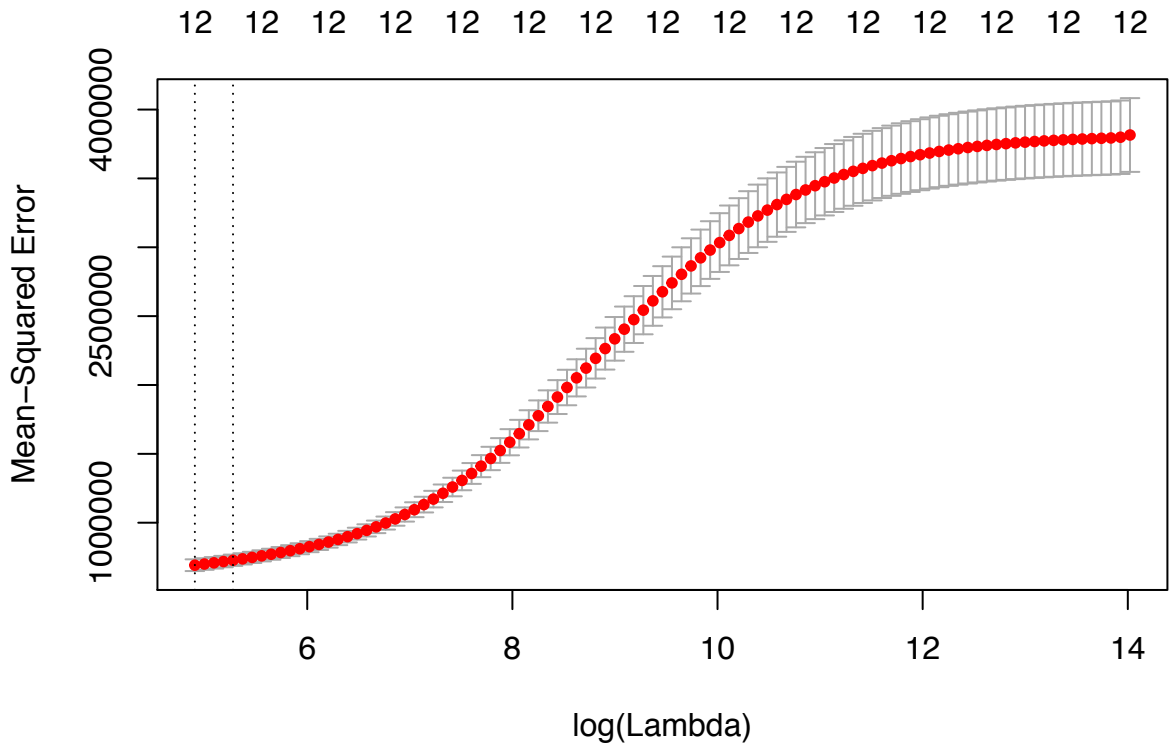
Format | B I U x_2 x^2 | I_x | | | | | Ω | | | ∑ | ABC |

Words: 0

Maximum marks: 10

Ridge regression

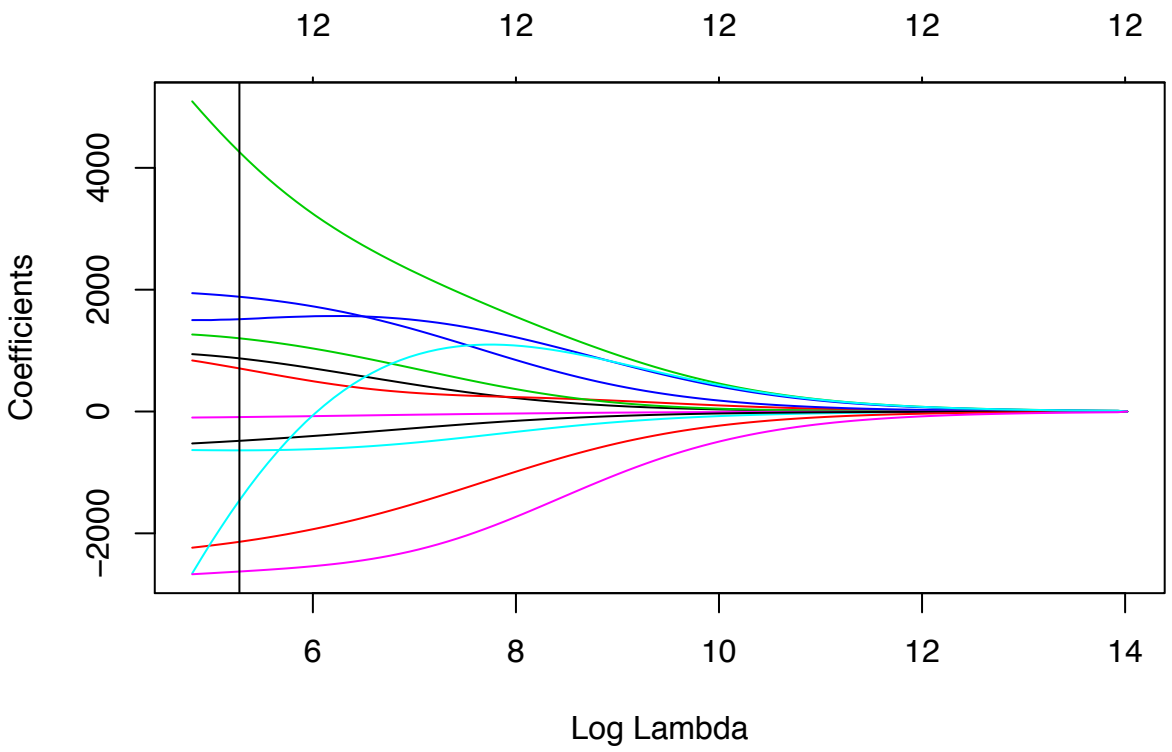
```
library(glmnet)
# set up model matrix, not include the intercept term because glmnet doesn't want that
model = formula(~season+year+holiday+notworkday+weather+temp+I(temp^2)+I(temp^3)+wind)
mm = model.matrix(model,data=train)[,-1]
set.seed(4268) #for reproducibility
cvfitridge = cv.glmnet(mm,train$y,alpha=0,standardize=TRUE,nfolds=10)
plot(cvfitridge)
```




```
print(cvfitridge$lambda.1se)
```

```
## [1] 195.6469
```

```
# also plot how coeffients change with (log)lambda
plot(cvfitridge$glmnet.fit,xvar="lambda")
abline(v=log(cvfitridge$lambda.1se))
```




```
# look at final model
coef(cvfitridge)
```

```
## 13 x 1 sparse Matrix of class "dgCMatrix"
##                        1
## (Intercept)  1363.37674
## season2       872.55853
## season3       707.40757
## season4      1201.65306
## year         1883.97691
## holiday      -637.68536
## notworkday    -91.38161
## weather2     -481.23220
## weather3    -2139.03652
## temp         4259.71239
## I(temp^2)    1515.42979
## I(temp^3)   -1458.51139
## wind        -2626.81004
```

```
# mse on training and test set
trainerrorridge=mean((predict(cvfitridge, s="lambda.1se", newx=mm)-train$y)^2)
predridge =predict(cvfitridge, s="lambda.1se", newx=model.matrix(model,data=test)[,-1])
testerrorridge=mean((predridge-test$y)^2)
c(trainerrorridge,testerrorridge)
```

```
## [1] 687947.1 730840.7
```

```
# estimated regression coefficients
# for least squares (lm) and ridge regression (glmnet)
res=cbind(fitlm$coefficients,coef(cvfitridge))
colnames(res)=c("lm","ridge")
res
```

```
## 13 x 2 sparse Matrix of class "dgCMatrix"
##                       lm       ridge
## (Intercept)   2902.1995  1363.37674
## season2        828.7597   872.55853
## season3       1093.7146   707.40757
## season4       1286.1808  1201.65306
## year          2052.3907  1883.97691
## holiday       -479.9857  -637.68536
## notworkday    -100.6592   -91.38161
## weather2      -705.9897  -481.23220
## weather3     -2495.9517 -2139.03652
## temp        -14643.0085  4259.71239
## I(temp^2)    57717.6057  1515.42979
## I(temp^3)   -47873.2054 -1458.51139
## wind         -2726.6905 -2626.81004
```

```
# and remembering the errors for lm and ridge
c(trainerrorlm,testerrorlm)
```

```
## [1] 490198.7 568906.1
```

```
c(trainerrorridge,testerrorridge)
```

```
## [1] 687947.1 730840.7
```

13 Regression tree

[Maximal score: 5 points]

We have fitted a regression tree to the training data, and R-code and print-out is found in the pdf-file in the left panel.

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

- year=1
- season=1
- holiday=0
- notworkday=0
- weather=3
- temp=0.3
- wind=0.2

Remember: at splits the criterion at the node is to the left in the tree.

b) The top split in the regression tree is on temp <0.442083 (left) and temp ≥ 0.442084 (right). What is the criterion that leads to this split? Include a mathematical formula in your argument.

c) What are similarities and differences between the greedy tree fitting algorithm and forward stepwise selection in regression?

Fill in your answer here

Format | B I U x_2 x^2 I_x | Ω | ∑ | ABC

Words: 0

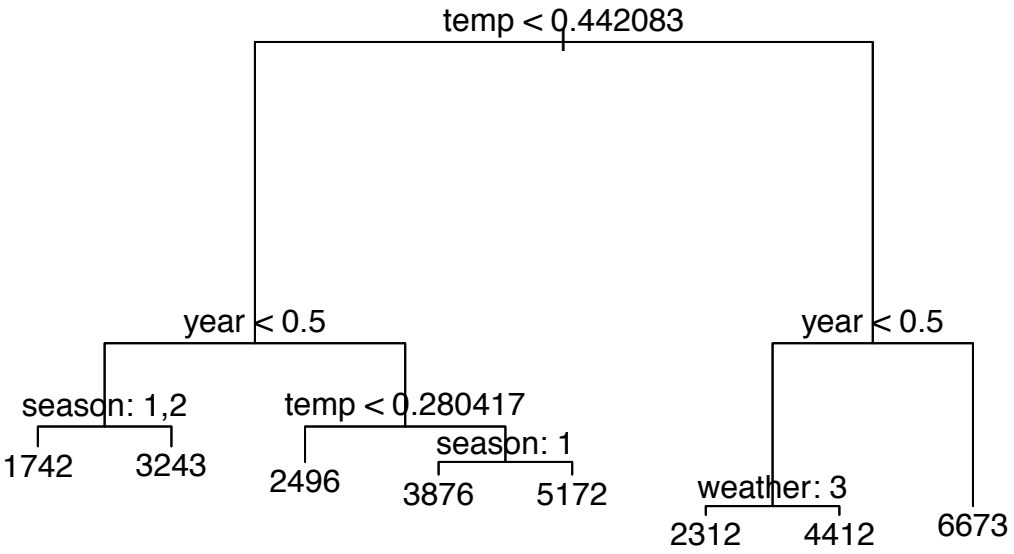
Maximum marks: 5

Regression tree

```
library(tree)
fitT=tree(y~., data=train)
summary(fitT)
```

```
##
## Regression tree:
## tree(formula = y ~ ., data = train)
## Variables actually used in tree construction:
## [1] "temp"    "year"    "season"  "weather"
## Number of terminal nodes:  8
## Residual mean deviance:  774800 = 381200000 / 492
## Distribution of residuals:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -5150.0  -430.7   126.0     0.0   504.4  2110.0
```

```
plot(fitT)
text(fitT,pretty=TRUE)
```




```
fitT
```

```
## node), split, n, deviance, yval
##       * denotes terminal node
##
##  1) root 500 1.895e+09 4493
##    2) temp < 0.442083 210 5.322e+08 3073
##      4) year < 0.5 110 1.009e+08 2138
##        8) season: 1,2 81 3.150e+07 1742 *
##        9) season: 4 29 2.128e+07 3243 *
##      5) year > 0.5 100 2.293e+08 4101
##       10) temp < 0.280417 25 1.599e+07 2496 *
##       11) temp > 0.280417 75 1.274e+08 4636
##         22) season: 1 31 1.819e+07 3876 *
##         23) season: 2,4 44 7.866e+07 5172 *
##    3) temp > 0.442083 290 6.319e+08 5522
```

```
##      6) year < 0.5 143 9.863e+07 4339
##       12) weather: 3 5 5.695e+05 2312 *
##       13) weather: 1,2 138 7.677e+07 4412 *
##      7) year > 0.5 147 1.382e+08 6673 *
```