Statistical Data Analysis – solved problems

Goals: The text provides a pool of solved problems for labs in the course on Statistical Data Analysis. The exercises help to deepen knowledge gained in parallel Rmd files. At the same time, they serve as illustrative examples of future exam questions.

1 Linear and non-linear regression

Problem 1. *(10 p) You built a linear model that predicts the median value of owner-occupied homes in $1,000's in a certain town (medv). The model works with the only independent variable (lstat) that captures the percentage of population with lower (economical) status in the given town. The model was built from a training set based on 506 towns and is this:*

```
lm(formula = medv ~ lstat, data = Boston)

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 34.55384    0.56263   61.41   <2e-16 ***
lstat       -0.95005    0.03873  -24.53   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6.216 on 504 degrees of freedom
Multiple R-squared:  0.5441,        Adjusted R-squared:  0.5432
F-statistic: 601.6 on 1 and 504 DF,  p-value: < 2.2e-16
```

(a) *(2 p)* Verbally describe the relationship between lstat and medv. Decide whether lstat affects medv and quantify how. Is it a statistically significant relationship? Why?

The model says that with each additional percentage of people with lower status the median value of homes decreases on average by $950. This relationship is statistically significant, based on the F-statistic as well as the lstat's t-value we can reject the null hypothesis that there is no relationship between lstat and medv.

(b) *(1 p)* How do you understand the meaning of Intercept? Is the value of this coefficient a reliable figure to be interpreted literally? Explain.

The value of Intercept says that the average median value of homes in a town with 0 percentage of people with lower status is around $34,553. This value looks reasonable, however, its true reliability depends on how far the model extrapolates (do we have any towns that at least approach no lower status representation in our training set?) and how far the model meets the linear regression assumptions (is the relationship between these variables truly linear?).

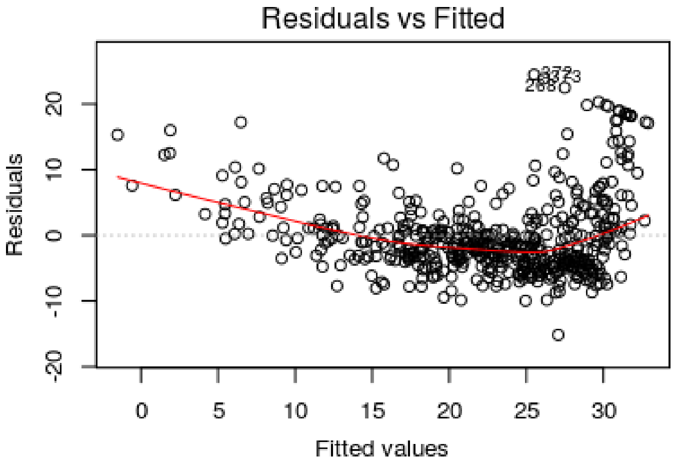
(c) *(1 p)* How much do we improve our median value forecast compared to the simple average forecast that ignores the knowledge of lstat? In other words, how much the knowledge of lstat helps?

The value of R-squared shows that we will reduce the variance of the median home value estimates by about half.

(d) *(2 p)* Calculate/estimate 95% confidence interval for β_{lstat}. What is this interval good for?

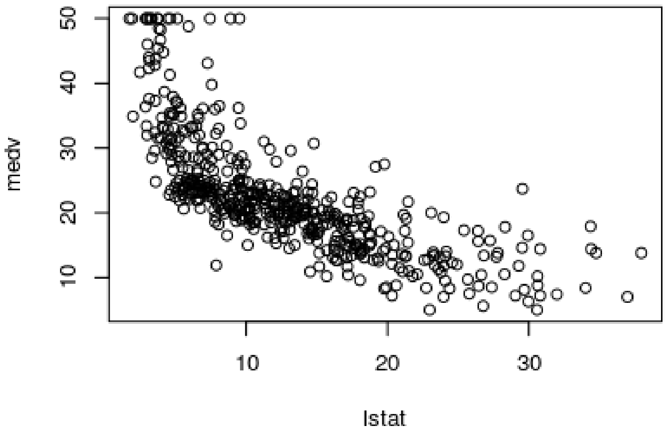
A rough estimate could be [-0.95005-2*0.03873,-0.95005+2*0.03873]=[-1.02751,-0.87259]. A more precise estimate puts $|t_{\alpha/2,m-2}| = |t_{0.025,504}|$ instead of 2 into the formula above, however, the value 1.964682 is close to the rough estimate. This confidence interval has about 95% chance to contain the true value of β_{lstat}. This interval helps us to assume on the strength of relationship between lstat and medv, the interval does not contain 0, the relationship could be considered significant.

(e) *(2 p)* Look at the model residual plot in the figure below (it plots differences between the actual and predicted values of the dependent variable). What conclusions can be drawn from the figure?

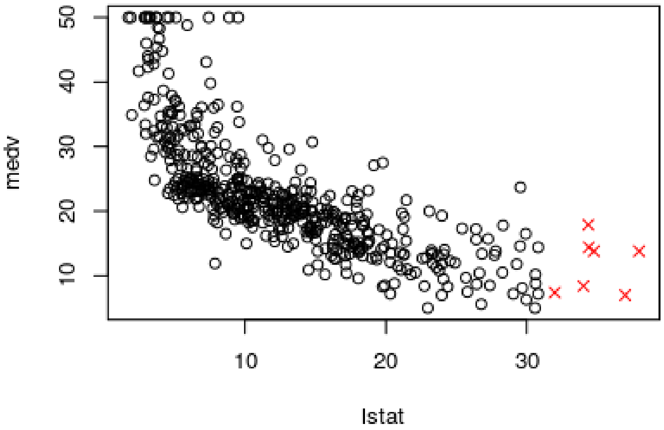


The plot shows that the assumption of linearity has not been met. The residuals should follow the normal distribution for all the values of lstat, they are heavily skewed in the plot. We should conclude that the relationship is non-linear and introduce new terms into the regression formula (lstat^2 is a good idea to start with).

(f) *(2 p)* Explain the concept of influential observations. Denote a couple of influential points in the scatter plot below and explain how you would find them.



An influential observation is an observation whose deletion from the dataset would noticeably change the model parameter estimates. It could either be an outlier (a data point that differs significantly from other observations) or a high-leverage point (an observation made at extreme values of independent variables). The most influential observations can be found in the figure below.



Problem 2. *(10 p) You are a mechanical locksmith and you are trying to find out how the shaft machining error is related to the machine tool parameter setting. You have compiled a multivariate linear model. The model expresses the relationship between the production error (the difference between the ideal shaft diameter and the actual shaft diameter, ProdError) and the setting of ten different continuous machine parameters (P1-P10). Below is the output you received:*

```
summary(lm(ProdError ∼ P1+P2+P3+P4+P5+P6+P7+P8+P9+P10),data=d)
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.05270    0.09576  -0.550   0.5835
X1           0.01298    0.08924   0.145   0.8847
```

```
X2           0.01596    0.10939   0.146   0.8843
X3          -0.02865    0.09079  -0.316   0.7531
X4           0.04611    0.09548   0.483   0.6303
X5           0.14151    0.09343   1.515   0.1334
X6          -0.02375    0.10277  -0.231   0.8178
X7           0.25522    0.10516   2.427   0.0172 *
X8           0.06672    0.08972   0.744   0.4590
X9           0.09949    0.10171   0.978   0.3306
X10         -0.04003    0.09317  -0.430   0.6685
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.9039 on 89 degrees of freedom
Multiple R-squared:  0.1145,        Adjusted R-squared:  0.01502
F-statistic: 1.151 on 10 and 89 DF,  p-value: 0.3346
```

(a) *(2 p)* Decide whether at least one of the machine parameters (independent variables) is useful for estimating a manufacturing error (ProdError). In other words, formally decide whether you can decline $H_0 : \beta_1 = \beta_2 = \cdots = \beta_{10} = 0$. Justify correctly.

The reasoning should be based on the F-statistic and its corresponding p-value. The null hypothesis cannot be rejected, the model does not seem to be useful. The reasoning that stems from the statistics reached for the individual variables could be misleading due to multiple comparisons. For 10 variables, truly valid H_0 and $\alpha = 0.05$, there is only $0.95^{10} = 0.6$ probability that there will be no type I error in the individual coefficient tests, 40% of trials will find at least one falsely significant coefficient.

(b) *(2 p)* Let us compare the full model constructed above with the intercept model and with the model that employs only the variable $P7$ identified as the most relevant. Let us compare them with F-test through an ANOVA run. Interpret the ANOVA table below.

```
lm.const<-lm(ProdError ~ 1,data=d) # the intercept model
lm.sel<-lm(ProdError ~ P7,data=d) # the P7 model
anova(lm.const,lm.sel,lm.all)
Analysis of Variance Table

  Res.Df    RSS Df Sum of Sq      F   Pr(>F)
1     99 82.114
2     98 76.076  1    6.0384 7.3911 0.007879 **
3     89 72.711  9    3.3647 0.4576 0.899016
```

ANOVA easily compares nested models where the independent variables of a simpler model make a subset of the independent variables of a more complex model. We order the models from the most simple to the most complex and ANOVA compares all the pairs of neighboring models. The ANOVA table suggests that lm.sel outperforms lm.const while lm.all does not further improve lm.sel. This conclusion is in contradiction with the conclusion in the previous answer. The contradiction arises from a methodological fault that we did. We used the same dataset to select $P7$ as the best variable and to test whether it performs well. This approach suffers from bias and could be misleading.

(c) *(2 p)* The dataset under consideration contains 100 samples. How do the type I error and type II error in the individual coefficient tests change with increasing number of samples if we maintain a constant level of significance α?

Type I error is a controlled parameter and its probability remains unchanged with the α value unchanged. However, the power of the test will increase, so the type II error will decrease. At the same time, the robustness of RSS, R^2 and consequently the F-test power will increase as well.

(d) *(4 p)* Describe in detail the way in which you would validate your models over the samples that you currently have. You can create additional auxiliary models. Describe the validation method, define the error function, and specify with which baseline you will compare the calculated error.

Let us assume that we want to compare lm.const, lm.sel and lm.all. Let us assume that our sample set is small and thus the hold-out method that splits the sample set on training and testing set is inappropriate (we need to use as many training samples as possible, the same holds for testing set). Then, a good option seems to be to run 10-fold cross-validation. We will always train our models on 9 folds and test them on the remaining one. We will gradually shift the testing fold. The dependent variable is continuous, we can use the root mean square error (RMSE) or mean absolute percentage error (MAPE). The error will always be calculated over the testing fold and averaged over the folds. If we repeat 10-fold cross-validation multiple times, we can statistically test whether performances of the individual models truly differ.

Watch out. Feature selection is a part of training process. It cannot be done only once before cross-validation, it must be repeated again and again for each split. Consequently, we will have 10 different lm.sel models to test, the set of relevant variables included into the model may change over folds as well as their regression coefficients. These 10 models will serve to estimate the performance of the final lm.sel model. Only the final model (to be reported and deployed) could be based on all the available samples, and will thus certainly employ the variable $P7$.

Problem 3. *(10 p) There is a cubic spline with one knot ξ given by the formula: $f(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 (x - \xi)^3_+$.*

(a) *(1 p)* Define the basis function $(x - \xi)^3_+$.

The definition is: $(x - \xi)^3$ for $x > \xi$, otherwise 0.

(b) *(1 p)* How many degrees of freedom does the given cubic spline have? Why?

A cubic spline with K knots has $K + 4$ parameters or degrees of freedom. Our spline has one knot and thus it has 5 independent parameters/degrees of freedom. The number of parameters can be seen from the formula above too, there are $\beta_0, \ldots, \beta_4$ there.

(c) *(1 p)* What are the properties of the cubic spline at the knot?

The spline is continuous at the knot and it has a continuous first and second derivative there. The properties follow from the general properties for d-degree splines.

(d) *(2 p)* Write down a cubic spline with one knot as a piecewise polynomial. Note: you will only change the form of notation, name the parameters differently from the spline parameters above.

$f_1(x) = a_1 + b_1x + c_1x^2 + d_1x^3$ for $x < \xi$

$f_2(x) = a_2 + b_2x + c_2x^2 + d_2x^3$ for $x \geq \xi$

(e) *(3 p)* Express the piecewise polynomial parameters using the cubic spline parameters $\beta_0, \beta_1, \ldots, \beta_4$.

The procedure is straightforward: the spline must match f_1 before the knot and f_2 after the knot. For the first polynomial it is trivial, because the basis function is zero before the first knot: $a_1 = \beta_0$, $b_1 = \beta_1$, $\ldots$, $d_1 = \beta_3$. For the second polynomial it holds: $a_2 + b_2x + c_2x^2 + d_2x^3 = \beta_0 + \beta_1x + \beta_2x^2 + \beta_3x^3 + \beta_4(x-\xi)^3$. By developing the last term we get: $\beta_0 + \beta_1x + \beta_2x^2 + \beta_3x^3 + \beta_4(x^3 - 3x^2\xi + 3x\xi^2 - \xi^3) = (\beta_0 - \beta_4\xi^3) + (\beta_1 + 3\beta_4\xi^2)x + (\beta_2 - 3\beta_4\xi)x^2 + (\beta_3 + \beta_4)x^3$, of which follows: $a_2 = \beta_0 - \beta_4\xi^3$, $b_2 = \beta_1 + 3\beta_4\xi^2$, $c_2 = \beta_2 - 3\beta_4\xi$, $d_2 = \beta_3 + \beta_4$.

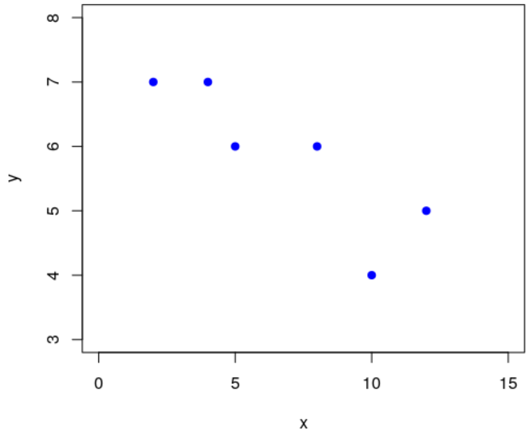
(f) *(2 p)* Proof that the piecewise cubic polynomial found in the previous two steps maintains the knot properties of a cubic spline.

Continuity $f_1(\xi) = f_2(\xi)$ can be proven by substituting for coefficients a, b, c, d: $f_1(\xi) = \beta_0 + \beta_1\xi + \beta_2\xi^2 + \beta_3\xi^3$, $f_2(\xi) = \beta_0 - \beta_4\xi^3 + (\beta_1 + 3\beta_4\xi^2)\xi + (\beta_2 - 3\beta_4\xi)\xi^2 + (\beta_3 + \beta_4)\xi^3 = \beta_0 + \beta_1\xi + \beta_2\xi^2 + \beta_3\xi^3 = f_1(\xi)$.

Continuity of the first derivative $f_1'(\xi) = f_2'(\xi)$ can be confirmed by substituting for the coefficients and deriving: $f_1'(\xi) = \beta_1 + 2\beta_2\xi + 3\beta_3\xi^2 = f_2'(\xi)$.

Continuity of the second derivative $f_1''(\xi) = 2\beta_2 + 6\beta_3\xi = f_2''(\xi)$.

Problem 4. *(10 p) A robot must traverse a path defined by six points, as shown in the figure below. The robot is required to visit all the points while finding the shortest path that is smooth. Discuss the extent to which known non-linear methods achieve these goals.*



(a) *(2 p)* Name the methods that lack smoothness and can, therefore, be excluded. Present the best solution these methods can provide.

Obviously, we cannot use step functions. A step function that passes directly through the set of points and uses uses the minimum possible number of parameters is: $y = \beta_0 I(x < \xi_1) + \beta_1 I(\xi_1 \leq x < \xi_2) + \beta_2 I(\xi_2 \leq x < \xi_3) + \beta_3 I(x > \xi_3)$, where $\beta_0 = 7$, $\beta_1 = 6$, $\beta_2 = 4$, $\beta_3 = 5$ and $\xi_1 = 4.5$, $\xi_2 = 9$, $\xi_3 = 11$.

Similarly, the linear spline would not be smooth too: $y = \beta_0 + \beta_1 x + \beta_2(x - \xi_1)_+ + \beta_3(x - \xi_2)_+ + \beta_4(x - \xi_3)_+ + \beta_5(x - \xi_4)_+$, where $\beta_0 = 7$, $\beta_1 = 0$, $\beta_2 = -1$, $\beta_3 = 1$, $\beta_4 = -1$, $\beta_5 = 1.5$, and $\xi_1 = 4$, $\xi_2 = 5$, $\xi_3 = 8$, $\xi_4 = 10$.

Both the solutions are shown in the summary figure below.

(b) *(2 p)* Could polynomial regression be applied? How many parameters would it need?

Polynomial regression proposes a solution that is smooth. The degree 5 polynomial with 6 parameters interpolates all the points that robot must visit: $y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 x^4 + \beta_5 x^5$. The path does not seem to be long and oscillate, extrapolation beyond the visited places ($x < 2$ and $x > 12$) is not important here.

If we used a lower degree, the robot would miss (most) points. For larger degrees, the problem is ill-posed. The degree 5 solution as well as the degree 3 solution are shown below.

(c) *(2 p)* Show a couple of solutions with higher-order regression splines. Propose the best parametrization, compare them in terms of path length.

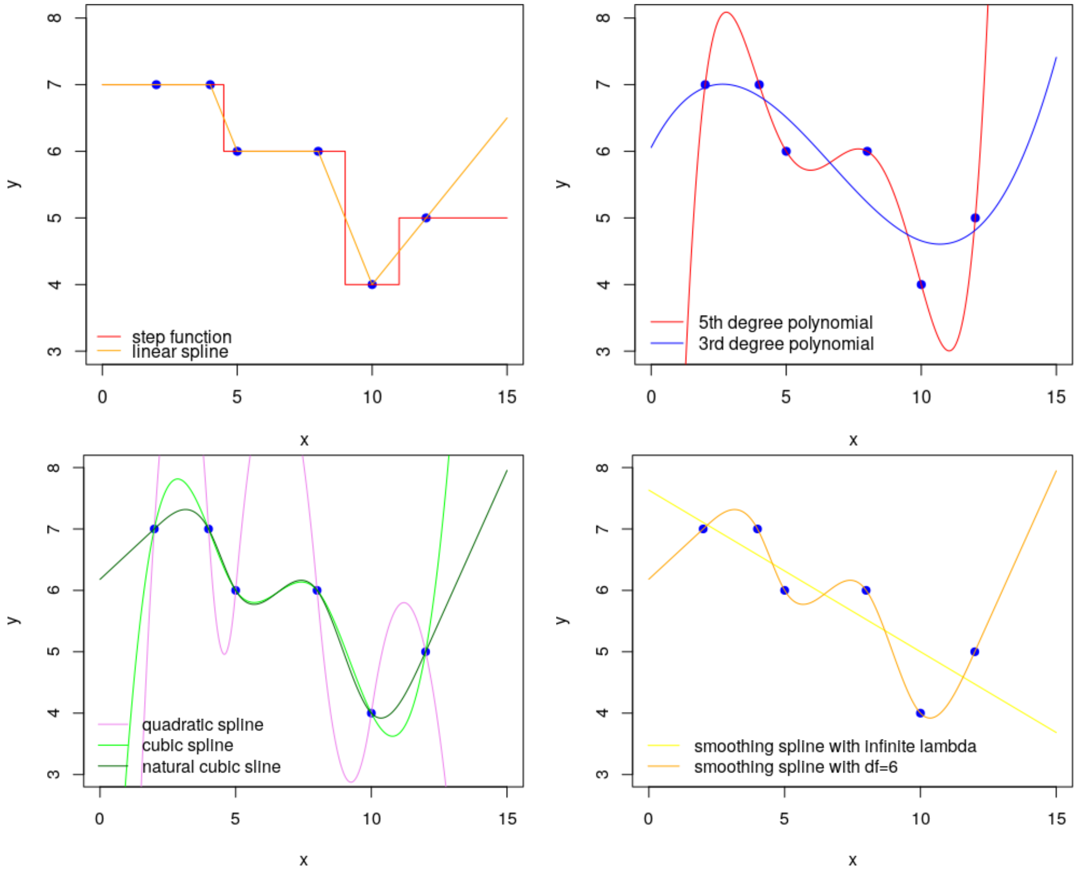
Quadratic or cubic spline could be a good solution here. The quadratic spline needs three knots, for fewer it underfits (does not visit all the places), for more it overfits (starts to oscillate and propose too long paths): $y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3(x - \xi_1)^2_+ + \beta_4(x - \xi_2)^2_+ + \beta_5(x - \xi_3)^2_+$. The cubic spline needs two knots only to reach 6 parameters and pass through the data directly: $y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4(x - \xi_1)^3_+ + \beta_5(x - \xi_2)^3_+$.

Natural splines can further help shorten the path and provide reasonable extrapolation as they extrapolate linearly. Despite the fact the assignment does not specify whether we should consider the outer regions or simply start in the leftmost place and finish in the rightmost one, linear extrapolation also helps to shorten the path through the places.

All the three solutions are shown below. Obviously, the natural spline with four knots represents the optimal solution of our task. The quadratic spline tends to oscillate more than its counterparts.

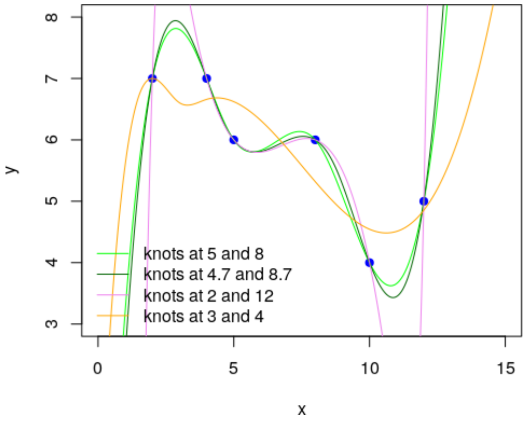
(d) *(2 p)* Discuss to what extent smoothing splines meet the robot goals.

Smoothing spline is a shrinked natural cubic spline with knots in all the observed xs. For $\lambda \to \infty$ it shrinks heavily and matches the straight line. The requirements are not met. For $\lambda = 0$ it shows zero residual sum of squares and passes directly through all the points. This solution equals the natural cubic spline shown above. The same outcome is reached if the number of degrees of freedom is set to 6 (df=6). Note that there are two alternative ways to parametrize smoothing spline, through λ or the number of degrees of freedom.



(e) *(2 p)* Eventually, where would you place the cubic spline knots and does their placement matter in our task?

The knot placement definitely matters. We will deal with the cubic spline with two knots as it represents its best parametrization. The best general solution is to distribute the knots equally. In our case, we should also place them directly to observed internal xs (not generally recommended for spline learning in regression because of overfitting). The figure below compares four solutions: a) the best solution: $\xi_1 = 5$, $\xi_2 = 8$, b) close the best is the automated solution that splits the observed xs into 3 bins: $\xi_1 = 4.7$, $\xi_2 = 8.7$, c) a bad selection with outer knot selection that causes oscillations: $\xi_1 = 2$, $\xi_2 = 12$, d) a bad selection with too close knots that underfits the data: $\xi_1 = 3$, $\xi_2 = 4$.



2 Linear regression and ANOVA

Problem 5. *(10 p) You are analyzing salary data from an unknown university stored in a data frame df. You want to find the key factors that actually influence the professors' wages. Besides the target variable salary you deal with the following set of independent variables: rank . . . a factor with levels AssocProf, AsstProf, Prof; discipline . . . a factor with levels A ("theoretical" departments) or B ("applied" departments); yrs.since.phd . . . a numerical variable that gives the number of years since PhD completion; yrs.service . . . a numerical variable that gives the number of years of service and sex . . . a factor with levels Female and Male.*

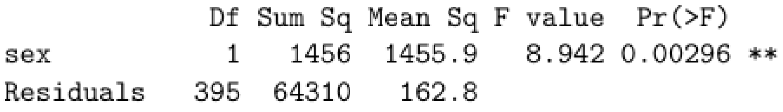
(a) *(2 p)* Explain in which way you would best decide whether *sex* influences *salary* with the aid of linear regression. Below there are two ultimate sample *lm* calls. Interpret both of them, decide whether any of them could be used to answer the role of *sex*. If they are not applicable, propose your own *lm* call.

```
Call 1:
lm(salary ~ sex, data = df)

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   101002       4809  21.001  < 2e-16 ***
sexMale        14088       5065   2.782  0.00567 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 30030 on 395 degrees of freedom
Multiple R-squared:  0.01921,       Adjusted R-squared:  0.01673
F-statistic: 7.738 on 1 and 395 DF,  p-value: 0.005667

Call 2:
lm(salary ~ rank + discipline + yrs.since.phd + yrs.service + sex, data = df)
```

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    65955.2     4588.6  14.374  < 2e-16 ***
rankAssocProf  12907.6     4145.3   3.114  0.00198 **
rankProf       45066.0     4237.5  10.635  < 2e-16 ***
disciplineB    14417.6     2342.9   6.154 1.88e-09 ***
yrs.since.phd    535.1      241.0   2.220  0.02698 *
yrs.service     -489.5      211.9  -2.310  0.02143 *
sexMale         4783.5     3858.7   1.240  0.21584
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 22540 on 390 degrees of freedom
Multiple R-squared:  0.4547,        Adjusted R-squared:  0.4463
F-statistic:  54.2 on 6 and 390 DF,  p-value: < 2.2e-16
```

The first call suggests that males have higher salaries and the difference is significant. However, we have to assume that the individual independent variables are related and deal with the multivariate model that contains all the predictors.

The actual role of *sex* can be assumed from the coefficient that is related to *sex* variable in the Call 2 and its p-value. There, the absolute value of the regression coefficient adjoined to *sex* is much smaller than in Call 1 and the difference between males and females seems to be insignificant. We would need more data to decide with more power, however, the significance of *sex* in Call 1 was obviously caused by the fact that males have higher ranks and longer careers. This could easily be checked by e.g. *aov(yrs.since.phd ∼sex,df)*:

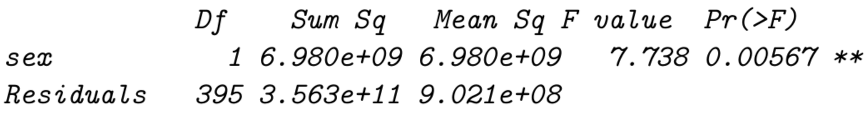


```
             Df Sum Sq Mean Sq F value  Pr(>F)
sex           1   1456  1455.9   8.942 0.00296 **
Residuals   395  64310   162.8
```

This ANOVA application confirms that there is significant relationship between gender and the length of career. Consequently, *yrs.since.phd* may play a role of confounder in Call 1. It has a relationship both with *salary* and *sex*, its influence could thus be explained by *sex* in Call 1.

(b) *(2 p)* Now you will make the following call: *summary(aov(salary∼sex,df))*. Explain what you will learn from the call.

One-way ANOVA (one independent variable and one dependent variable) could be considered equivalent to single variate *lm* calls. Consequently, we will get nearly the same message as we got in previous *lm* Call 1 (the same F-statistic and the same p-value). The main difference is in the form of the output (the coefficients are not reported here). In particular:



```
             Df    Sum Sq   Mean Sq F value  Pr(>F)
sex           1 6.980e+09 6.980e+09   7.738 0.00567 **
Residuals   395 3.563e+11 9.021e+08
```

(c) *(2 p)* Now you will make the following call: *summary(aov(salary ∼rank + discipline + yrs.since.phd + yrs.service + sex,df))*. Explain what you will learn from the call.

Unlike the previous case, this multivariate ANOVA call will not match the previous multivariate *lm* call. The reason is that both the methods treat sum of squares in regression differently. In *lm*, we search for main effects of the individual variables, their influence is considered in parallel (Type III sum of squares). In *aov*, we evaluate the individual predictors sequentially, in the order of their appearance in the formula (Type I sum of squares). For example, if we enter *rank* first, its sums of squares are computed ignoring *discipline* and other variables. Therefore, any variance in *salary* that is shared by *rank* and *discipline* will be attributed solely to *rank*. The sums of squares for *discipline* will then be computed excluding any variance that has already been attributed to *rank*. Consequently, in this type of call we may learn what is the contribution of a new variable to the existing model based on all the variables that precede it in the formula.

The above described *aov* call in fact performs an analysis of covariance (ANCOVA) which blends ANOVA and regression. It evaluates whether the means of a dependent variable (DV) are equal across levels of a categorical independent variable (IV) often called a treatment, while statistically controlling for the effects of other (continuous) variables that are not of primary interest, known as covariates (CV). In particular:

```
                 Df    Sum Sq   Mean Sq F value    Pr(>F)
rank              2 1.432e+11 7.162e+10 140.979   < 2e-16 ***
discipline        1 1.843e+10 1.843e+10  36.280  3.95e-09 ***
yrs.since.phd     1 1.656e+08 1.656e+08   0.326    0.5683
yrs.service       1 2.576e+09 2.576e+09   5.072    0.0249 *
sex               1 7.807e+08 7.807e+08   1.537    0.2158
Residuals       390 1.981e+11 5.080e+08
```

(d) *(2 p)* Explain in which way you would best decide whether *sex* influences *salary* with the aid of ANOVA. Show a particular R call or calls (*aov* and *anova* commands).

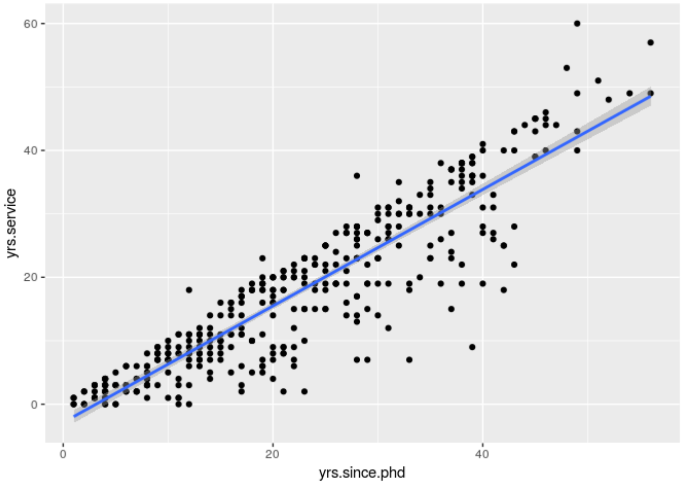
We have already shown that one-way ANOVA could be considered equivalent to single variate *lm* call and it may oversimplify in multivariate tasks because of ignoring confounders. The best calls could be: *summary(aov(salary ∼rank + discipline + yrs.since.phd + yrs.service + sex,df))*, in fact any *aov* call that puts *sex* last in the full multivariate model. Another option is: *anova(lm(salary ∼rank + discipline + yrs.since.phd + yrs.service,df),lm(salary ∼rank + discipline + yrs.since.phd + yrs.service + sex,df))*. The outcome of both the calls is exactly the same wrt the role of *sex* under discussion.

However, this approach studies sequential effect of *sex* wrt the existing model. The question that we answer with these calls is different from the original question whether *sex* influences *salary*. It is much better answered by the previous Call 2 *lm(salary ∼rank + discipline + yrs.since.phd + yrs.service + sex, data = df)*.

(e) *(2 p)* Discuss the relationship between *yrs.service* and *yrs.since.phd*. Is there any issue to be checked? Consider both the real meaning of these two variables and *lm* calls above. If so, propose a solution.

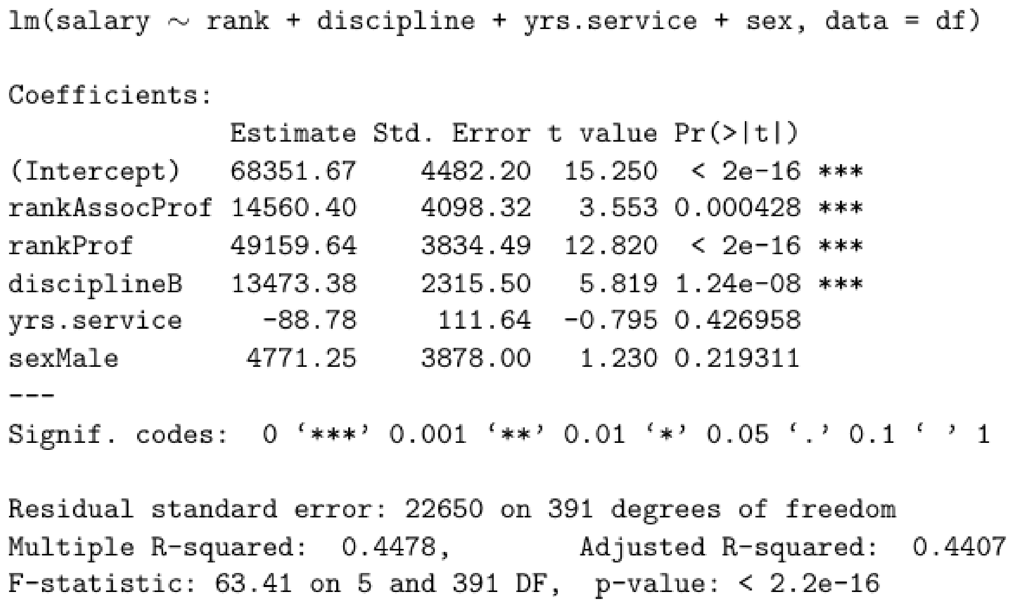
Obviously, these two variables are closely related and necessarily correlated. In general, the more years from PhD, the longer service. This correlation could be strong and may cause collinearity problems in the multivariate model. The issue may affect estimates regarding the individual predictors. In the full multivariate model (*lm* Call 2) it is unexpected and suspicious that salary decreases with the length of service. We would expect exactly the opposite and collinearity could be the reason.

A simple solution is to calculate correlation between *yrs.service* and *yrs.since.phd* (it is 0.91 in our data) and/or draw a scatter plot for the two variables:



Knowing the collinearity, we would possibly remove one of the two variables from the model (if we do so and remove *yrs.since.phd* in the first call and *yrs.service* in the second call, any of the two variables comes out insignificant then):

```
lm(salary ∼ rank + discipline + yrs.since.phd + sex, data = df)

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   67884.32    4536.89  14.963  < 2e-16 ***
rankAssocProf 13104.15    4167.31   3.145  0.00179 **
rankProf      46032.55    4240.12  10.856  < 2e-16 ***
disciplineB   13937.47    2346.53   5.940 6.32e-09 ***
yrs.since.phd    61.01     127.01   0.480  0.63124
sexMale        4349.37    3875.39   1.122  0.26242
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 22660 on 391 degrees of freedom
Multiple R-squared:  0.4472,       Adjusted R-squared:  0.4401
F-statistic: 63.27 on 5 and 391 DF,  p-value: < 2.2e-16
```



```
lm(salary ~ rank + discipline + yrs.service + sex, data = df)

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)    68351.67    4482.20  15.250  < 2e-16 ***
rankAssocProf  14560.40    4098.32   3.553 0.000428 ***
rankProf       49159.64    3834.49  12.820  < 2e-16 ***
disciplineB    13473.38    2315.50   5.819 1.24e-08 ***
yrs.service      -88.78     111.64  -0.795 0.426958
sexMale         4771.25    3878.00   1.230 0.219311
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 22650 on 391 degrees of freedom
Multiple R-squared:  0.4478,        Adjusted R-squared:  0.4407
F-statistic: 63.41 on 5 and 391 DF,  p-value: < 2.2e-16
```

If we look at the model scores and compare the simplified models with lm Call 2, it is obvious that shrinkage is beneficial (no decrease in Adjusted R-squared, slightly tighter coefficient estimates).

3 Generalized linear models (GLMs)

Problem 6. *(10 p) Your task is to predict household size in the Philippines. In particular, you are supposed to predict the number of people sharing a house (the variable denoted as total) as a function of the age of the household head (age) and location island (location). We have $m = 1,500$ observations, they are stored in the data frame named fHH1. The target variable is obviously a count variable, we will work with generalized linear models and Poisson regression in particular.*

(a) *(2 p)* Explain the terms *saturated model* and *null model*.

A saturated model has as many estimated parameters as observations. It is over-parametrized and just interpolates the data. We can implement it with the aid of a categorical independent variable that has as many different values as observations. This variable will be transformed into $m-1$ dummy binary variables, the total number of β parameters will be equal to m (including the intercept).

The null model is the model that matches the null hypothesis. In regression it commonly means that there is no relationship between the dependent variable and predictors. The predictors are thus unimportant and the model contains only intercept.

In GLMs, these two models commonly serve to calibrate the fit of any reasonable model. The fit can be quantified in terms of likelihood, log-likelihood or deviance. The closer our model is to the saturated model, the better. The further our model is from the null model, the better.

(b) *(2 p)* Show how you would construct these two models in the household size task.

We deal with the Poisson models $y_i \sim Poisson(\mu_i)$, their μ_i can be defined as follows:

null model: $\log \mu_i = \beta_0$,

saturated model: $\log \mu_i = \beta_0 + \beta_1 I(i = 1) + \beta_2 I(i = 2) + \cdots + \beta_{m-1} I(i = m - 1)$,

where $I(.)$ is an indicator function (takes 1 if its condition is met, 0 otherwise).

In R the models can easily be constructed as follows:

```
glm.sat<-glm(total ~ as.factor(1:nrow(fHH1)), data = fHH1, family=poisson)
glm.null<-glm(total ~ 1, data = fHH1, family=poisson)
```

(c) *(2 p)* Create a reasonable Poisson model based on the variables contained in *fHH1*. Interpret the model.

The most straightforward model just includes both the independent variables:

```
glm.m<-glm(total ~ age + location, data = fHH1, family=poisson)
summary(glm.m)

Coefficients:
                      Estimate Std. Error z value Pr(>|z|)
(Intercept)           1.472332   0.061922  23.777  < 2e-16 ***
age                  -0.004598   0.000940  -4.891    1e-06 ***
locationDavaoRegion  -0.014191   0.053800  -0.264  0.79196
locationIlocosRegion  0.052387   0.052652   0.995  0.31975
locationMetroManila   0.072806   0.047195   1.543  0.12291
locationVisayas       0.127532   0.041742   3.055  0.00225 **
---

    Null deviance: 2362.5  on 1499  degrees of freedom
Residual deviance: 2320.8  on 1494  degrees of freedom
AIC: 6705.7
```

The number of variables is small wrt the sample size. Therefore, knowing the p-values for age and $Visayas\ island$ we may prejudge that both of the independent variables are important for the household size prediction. The model suggests that the mean household size decreases by the multiplicative factor of $e^{-0.004598} = 0.9954$ for each year that the head of household is older. Similarly, the location Visayas has a larger mean household size than the referential island (its name is not shown), the ratio is $e^{0.1275} = 1.136$.

(d) *(2 p)* How would you statistically evaluate the model, i.e., how would you decide whether any of the predictors is useful?

GLMs can be evaluated in terms of their deviance (similar role as the residual sum of squares in linear regression). Vaguely, the null deviance shows the difference between the perfect fit (saturated model) and the worst fit (the null model). The residual deviance shows the difference between the perfect fit and the model. A criterion technically identical to R^2 can be used:

$$R^2 = 1 - \frac{D_{res}}{D_{null}} = 0.018$$

In linear regression, R^2 quantifies the percentage of variance explained. In here, R^2 quantifies the percentage of deviance explained. Our model explains less than 2% of deviance contained in the null model. This is not much. At the same time, p-values related to two of the predictors seem to be high. Is it a contradiction? Not necessarily. We can statistically test the difference in deviances of two nested models:

```
anova(glm.null,glm.m,test="Chisq")

Analysis of Deviance Table
  Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
1      1499     2362.5
2      1494     2320.8  5   41.646 6.946e-08 ***
```

Consequently, we can reject the null hypothesis that our model fits the data equally well as the null model in favor of the alternative hypothesis that our model significantly outperforms the null model. Despite the small relative difference in deviances, our model (very likely) helps.

(e) *(2 p)* Describe the method that serves to estimate parameters in GLMs. Specify for Poisson regression and saturated model.

GLMs tune their parameters by maximizing model likelihood. To do so, they employ the method of iteratively reweighted least squares. For Poisson regression, the optimization formula is:

$$Poisson(\mu) = \frac{\mu^y e^{-\mu}}{y!} = \exp\left(y \log \mu - \mu - \log(y!)\right)$$

(knowing that: $\mu^y = \exp(y \log \mu)$)

Log-likelihood to be maximized: $\ell(\beta) = \sum_{i=1}^{m} (y_i \log \mu_i - \mu_i - \log(y_i!))$

$$\ell(\beta) = \sum_{i=1}^{m} \big(y_i(\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}) - \exp(\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}) - \log(y_i!)\big)$$

(knowing that: $\log \mu_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}$ for canonical link)

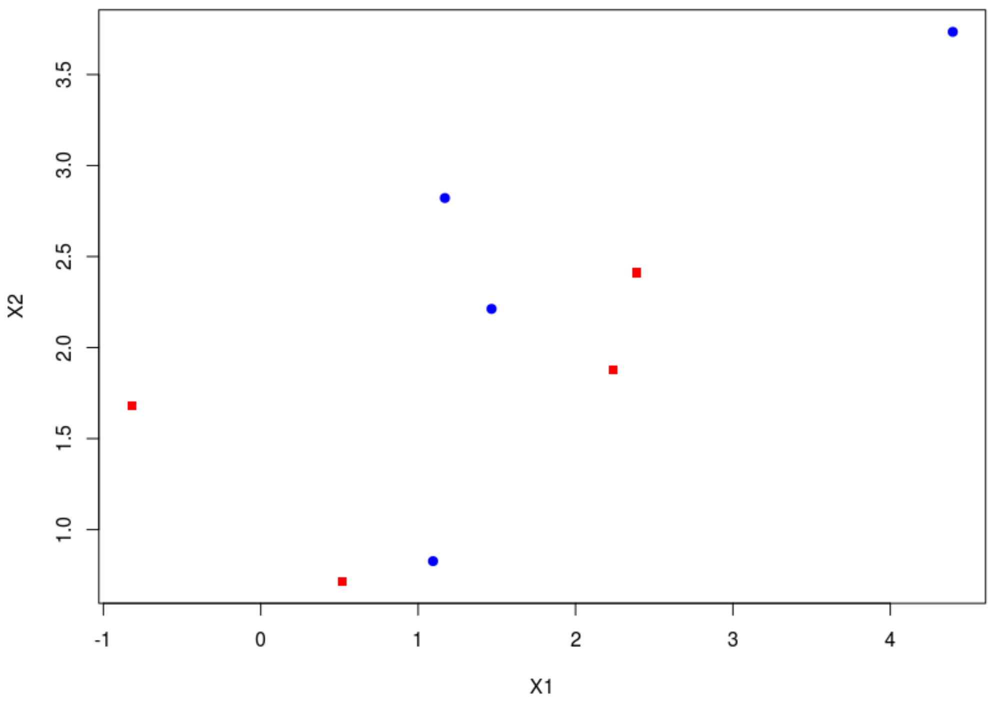
For the saturated model in Poisson regression it is clear that $\forall i : \mu_i = y_i$. In other words, interpolation and over-parametrized model fit the targets with means perfectly. At the same time it holds that the variance will also be equal to μ_i as it must be equal to mean in Poisson distribution. Consequently, the saturated model will not have zero log-likelihood, its log-likelihood will be:

$$\ell(\beta_{sat}) = \sum_{i=1}^{m} (y_i \log y_i - y_i - \log(y_i!))$$

The log-likelihood is 0 only for targets where $y_i = 0$. Otherwise, it takes non-zero values (-1 for $y_i = 1$, -1.31 for $y_i = 2$, -1.50 for $y_i = 3$, etc.)

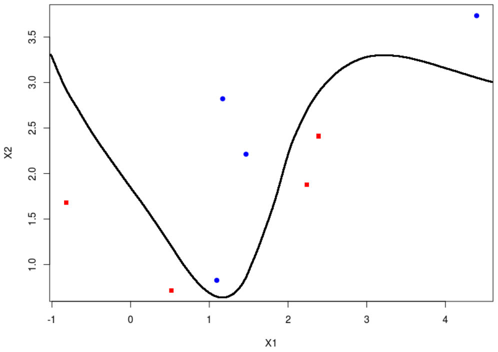
4 Classification

Problem 7. *(10 p) Let us consider a 2-class classification problem represented by a training set visualized below (there are two independent variables $X_1 \in R$ and $X_2 \in R$, the class $Y \in \{C_1, C_2\}$ is shown in colors and point shapes). The main goal is to propose a classifier $f : \mathcal{X} \to Y$.*



(a) *(2 p)* Draw a decision boundary that perfectly splits the training data (as simple as you can). Write down a formula for such a decision boundary (only its mathematical form is important, use symbols for coefficients, not numbers).

Obviously, the training examples are not linearly separable. As for polynomial decision boundaries, the third degree polynomial represents the least flexible split. One of its analytical forms could be: $X_2 = \theta_0 + \theta_1 X_1 + \theta_2 X_1^2 + \theta_3 X_1^3$.

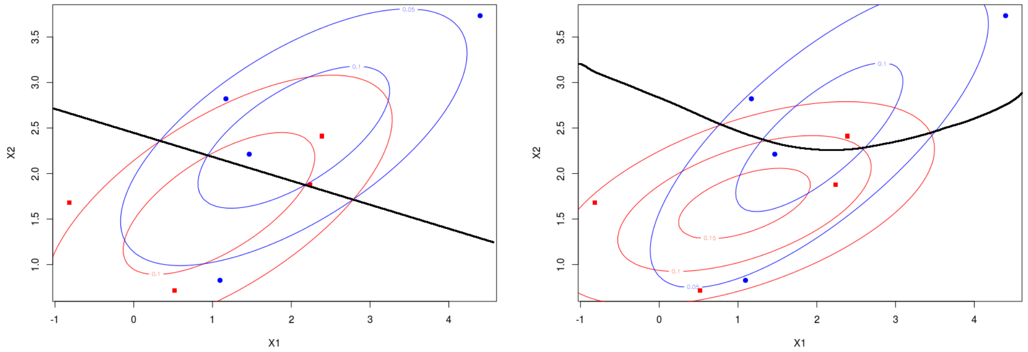


(b) *(2 p)* Is linear discriminant analysis (LDA) able to reach 100% accuracy on this training set? Quadratic discriminant analysis (QDA)? Logistic regression (LR)? Explain.

The training examples are not linearly separable. LDA and logistic regression have a linear decision boundary, they cannot be 100% accurate. In fact, they classify correctly only 5 out of 8 training examples (the best linear classifier makes 2 training errors, it does not have to follow any assumptions). QDA works with conic section boundaries, its flexibility is also insufficient to reach 100% accuracy on this training set. In this training set, it classifies correctly 5 out of 8 training examples too (the best conic section classifies all the training examples correctly).

(c) *(3 p)* Explain the difference between LDA and QDA application to this training set? Which method is more likely to better fit future test data?

Both LDA and QDA perform discriminative analysis. They both stem from the Bayes' theorem and assume that each class can be modeled by a Gaussian distribution. LDA also assumes that all the classes share the same covariance matrix. The decision boundaries they construct for the given training set are shown below (LDA left, QDA right).



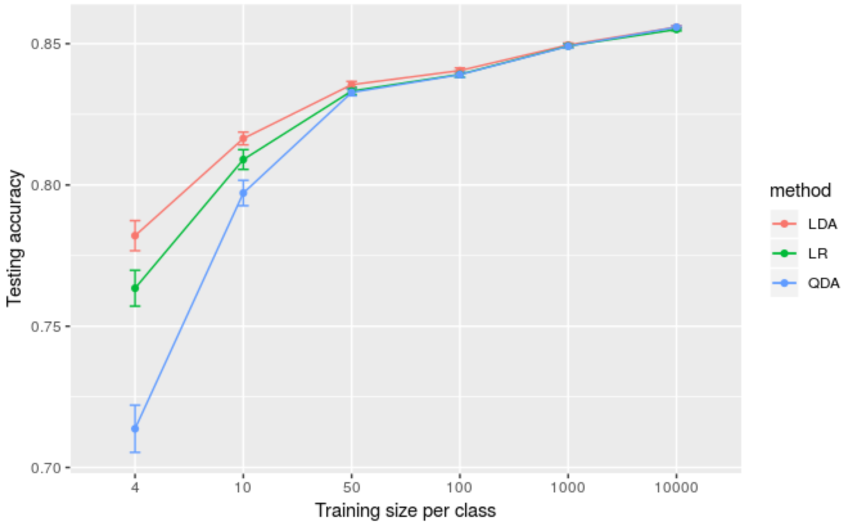
Without any knowledge about data distributions we cannot decide which method will perform better. However, we deal with an extremely small training set. For such a small training set it is reasonable to employ the simpler method with a smaller number of parameters to set, which is LDA. This method is less likely to suffer from variance that makes a bigger issue than bias here.

(d) *(3 p)* Assume that the underlying data distribution is as follows: the samples have bivariate normal distribution given their class, the class means are different, however, the class covariance matrices match. Decide and explain, which method (LDA, QDA or logistic regression) will best classify test samples when using the given training set. Then, make the same decision for a sufficiently large training set having e.g., 10,000 samples per class. Explain both the decisions.

The task definition obviously matches the LDA assumptions. This method will benefit from the right flexibility as well as its normality assumption. It is going to work best, especially for small sample sizes. For very large sample sizes, all the considered methods will perform equally.

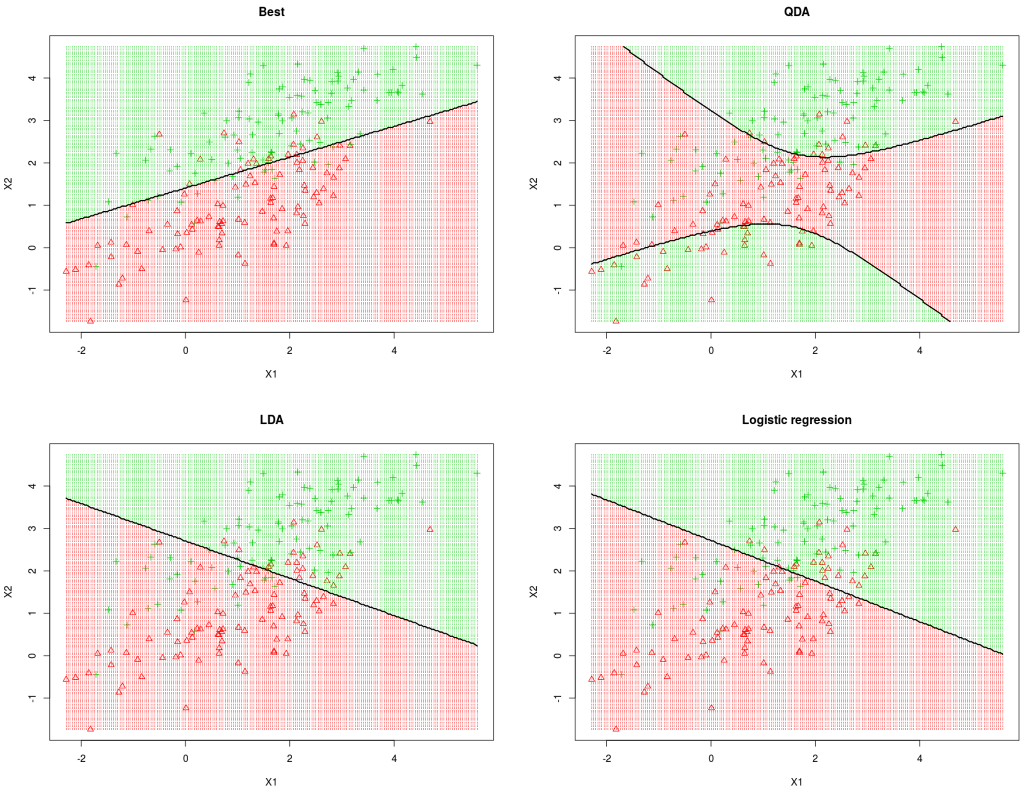
QDA and LR are more difficult to theoretically compare. QDA will suffer from higher variance as it is too flexible for the given task, on the other hand, it may benefit from the knowledge of within-class normality.

The figure below shows an empirical comparison of the three methods for the setting S: $\mu_{C1} = c(1,1)$, $\mu_{C2} = c(2,3)$, $\Sigma_{C1} = \Sigma_{C2} = matrix(c(2,1,1,1),2,2)$.



The plot summarizes relationship between train set size and testing accuracy. In this experiment, a train set of each size was randomly generated 100 times. Testing accuracy comes from a large test set that remained constant during the experiment. The comparison confirms our reasoning about LDA dominance. It also shows that LR outperforms QDA for small training sets. The negative role of QDA flexibility outweighs the positive role of its knowledge of within-class normality.

Deeper understanding can be gained from the plots below. They illustrate how LDA, QDA and LR work with the small training set presented in the beginning of this exercise (this set was actually generated with the setting S too, red $\sim$ C1, blue $\sim$ C2). The plots show a large test set where red triangles represent C1 and green crosses represent C2. The first benchmark plot demonstrates the best decision boundary, it can be easily approached by e.g., LDA with a large train set.



5 Dimensionality reduction

Problem 8. *(10 p) Let us consider a real data matrix* $\mathbf{X}$ *that contains 1,000 samples and 100 variables. The matrix has been centered, the variables in the matrix columns have zero means. Your goal is to visualize the data in 2 dimensional space in order to understand them.*

(a) *(3 p)* Your first option to reduce the dimension is principal component analysis (PCA). Explain how the method works, provide a mathematical description of principal components.

PCA is a linear dimension reduction method that transforms the original data in such a way that the maximum amount of variance contained in the original data remains contained in a small number of principal components. Principal components can be seen as mutually orthogonal lines that minimize the average squared distance from the original points to the line.

In our case: $m = 1,000$, $D = 100$, $L = 2$. The principal components can be reached as eigenvectors of covariance matrix: $\mathbf{C_X} = \frac{1}{m}\mathbf{X^T X}$. The jth principal component is the eigenvector with the jth largest eigenvalue λ_j: $\mathbf{C_X p_{\cdot j}} = \lambda_j \mathbf{p_{\cdot j}}$, $\mathbf{p_{\cdot j}} = \langle p_{1j}, p_{2j}, \ldots, p_{Dj} \rangle$, $\sum_{i=1}^{D} p_{ij}^2 = 1$. The vector of jth principal component scores is a vector of length m that is a normalized linear combination of the original variables weighted by $\mathbf{p_{\cdot j}}$: $\mathbf{t_{\cdot j}} = p_{1j}\mathbf{x_{\cdot 1}} + p_{2j}\mathbf{x_{\cdot 2}} + \cdots + p_{Dj}\mathbf{x_{\cdot D}}$. The transformation has a matrix form, the transformation matrix $\mathbf{P}$ contains $\mathbf{p_{\cdot j}}$ in its jth column.

The projection to the output space is reached as: $\mathbf{T} = \mathbf{XP}$.

(b) *(2 p)* Mathematically define dimension reduction mapping F and reconstruction mapping f in our task.

PCA generates a transformation matrix: $\mathbf{P}_{\mathbf{DxD}}$ (columns represent principal components),

dimensionality reduction mapping: $F : \mathbf{T}_{\mathbf{mxL}} = \mathbf{X}_{\mathbf{mxD}}\mathbf{P}_{\mathbf{DxL}}$ (only L first columns used),

reconstruction mapping: $f : \mathbf{X}^{\mathbf{r}}_{\mathbf{mxD}} = \mathbf{T}_{\mathbf{mxL}}\mathbf{P}^{\mathbf{T}}_{\mathbf{LxD}} = \mathbf{X}_{\mathbf{mxD}}\mathbf{P}_{\mathbf{DxL}}\mathbf{P}^{\mathbf{T}}_{\mathbf{LxD}}$.

(c) *(3 p)* You already performed PCA. Is there any way to guess whether PCA is a good visualization method in our case?

Unsupervised learning has rather vague goals. It is not easy to say that a certain method works well for the given data without deeper knowledge of the domain and comparative tests of several dimension reduction methods.

However, the proportion of explained variance provides a simple test that helps us to decide whether PCA could potentially be useful. A small proportion of explained variance means that the reconstruction error is large and a lot of information gets lost.

In our case, it is unlikely to observe meaningful patterns in the final 2D visualization if the variance captured by the first two principal components is small. If they captured only 2% variance contained in our dataset (the ration between the new dimension and the original dimension) or slightly more, PCA failed to effectively compress the data. The visualization can easily miss patterns contained in the original data. On the other hand, if the variance captured by the first two principal components approaches 100% of the variance in the original data, PCA is able to approach zero reconstruction error and can hardly be overcome by its competitors. In general, uneven variance distribution between the principal components suggests that PCA might be helpful.

Mathematically:

variance in the original data: $var(\mathbf{X}) = \sum_{j=1}^{D} var(\mathbf{x}_{\cdot\mathbf{j}})$ (the sum of original variable variances),

explained variance: $var(\mathbf{T}) = \sum_{j=1}^{L} var(\mathbf{t}_{\cdot\mathbf{j}})$ (the sum of reduced variable variances),

decision making from proportion of explained variance: $\frac{var(\mathbf{T})}{var(\mathbf{X})} \gg \frac{L}{D}$?,

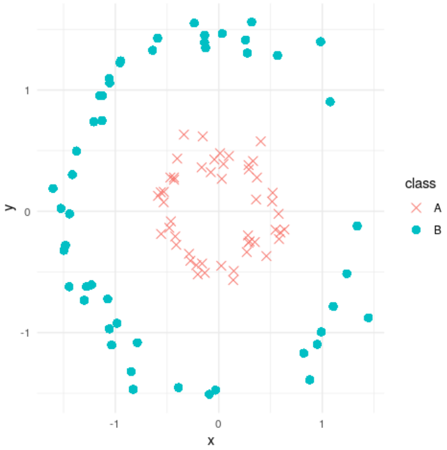
there is a couple of other ways to calculate the matrix variance too: $var(\mathbf{X}) = \frac{1}{m-1}||\mathbf{X}||_2^2$ (normalized squared L2 matrix norm), or $var(\mathbf{X}) = \sum_{j=1}^{D} \text{eigenvalues}(\mathbf{C_X})$) (sum of eigenvalues of the covariance matrix of $\mathbf{X}$), then the same decision making can be made as: $\frac{\sum_{j=1}^{L} \text{eigenvalues}(\mathbf{C_X})}{\sum_{j=1}^{D} \text{eigenvalues}(\mathbf{C_X})} \gg \frac{L}{D}$?

(d) *(2 p)* Explain how we can run PCA with 1,000 samples and 100 variables when knowing the curse of dimensionality principle. Is the sample size sufficient?

The curse of dimensionality principle says: in the absence of simplifying assumptions, the sample size needed to estimate a function with D variables to a given degree of accuracy grows exponentially with D. A general dimensionality reduction method requires a sample size exponential in the data dimension to be able to capture the manifold shape.

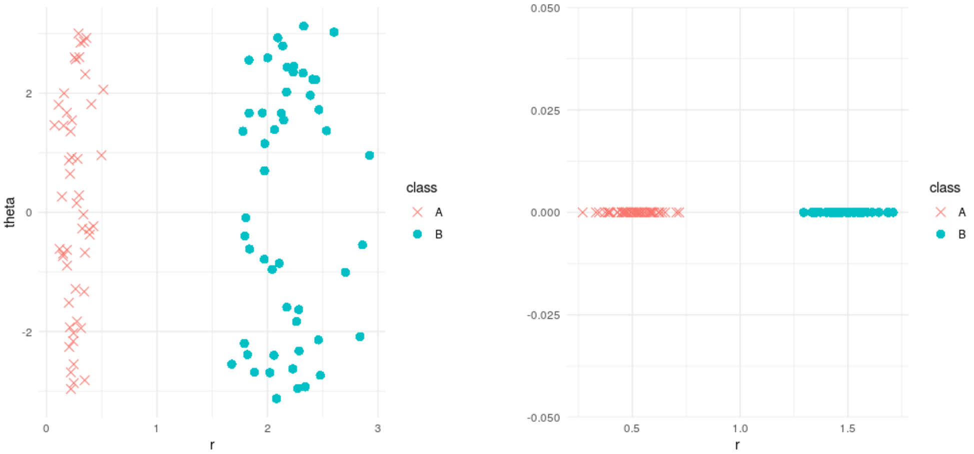
PCA is a linear dimensionality reduction method. The data is projected onto a lower dimensional linear subspace (hyperplane) and this way of projection represents the above mentioned simplifying assumption. The hyperplane should approximately match the observations. If the assumption fails, PCA does not work properly even with an arbitrarily large amount of input samples. If it holds the necessary sample size depends on: 1) the intrinsic dimension d of the assumed hyperplane (the manifold we learn could be a line, plane, . . . , D-dimensional hyperplane), 2) the amount of noise in the dataset (how precisely the samples hit the hyperplane). For example, in the case of zero noise, the necessary sample size only grows linearly with the intrinsic dimension of the assumed hyperplane. A plane with no noise embedded in 100-dimensional space only needs 3 samples to get identified and PCA finds the embedding from these 3 samples. For these reasons we can run PCA even though $m < D$. Of course, in presence of noise we can easily overfit the training data and we need more than $d + 1$ observations. At the same time, d is rarely known.

Problem 9. *(10 p) Let us consider a problem with two concentric circles in a two-dimensional space. The circles represent two different classes. There are 100 instances, with half belonging to class A and the other half to class B. See the picture below. Your task is to reduce the problem to an one-dimensional space, where the classes are linearly separable.*



(a) *(1 p)* Start with your own custom mapping avoiding traditional methods such as PCA, kernel PCA, multidimensional scaling, etc.

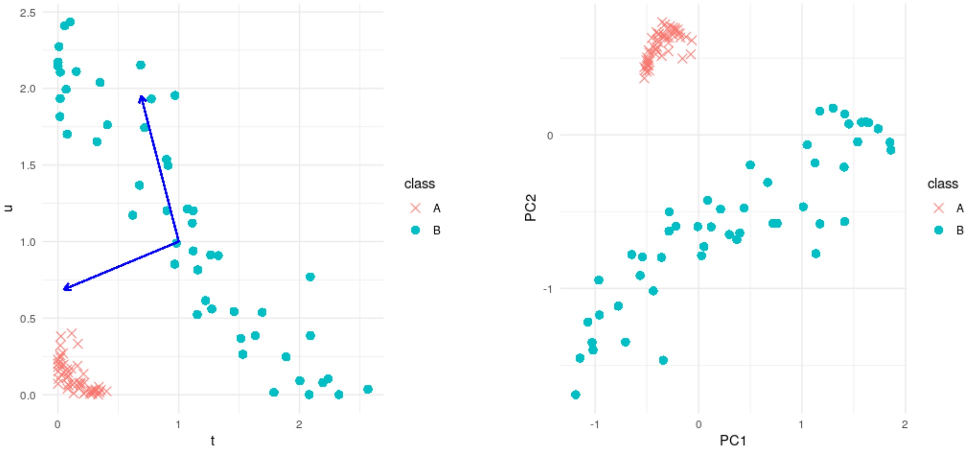
Obviously, the data have the radial structure. The conversion to polar coordinates can capture it: $r = \sqrt{x^2 + y^2}$, $\theta = atan2(y, x)$ (the left figure below). The radial distance from the origin is the coordinate that clearly splits the classes (the right figure below).



(b) *(1 p)* If we apply PCA directly, it will fail in our task. Propose a simple explicit transformation that will help PCA to solve the task.

The simplest transformation is: $t = x^2$, $u = y^2$. While the classes are already linearly separable in two-dimensional space (as shown in the left figure below), they are not separable in t nor u independently. After PCA, the main principal component captures the variance within class B, whereas the second principal component separates the classes.

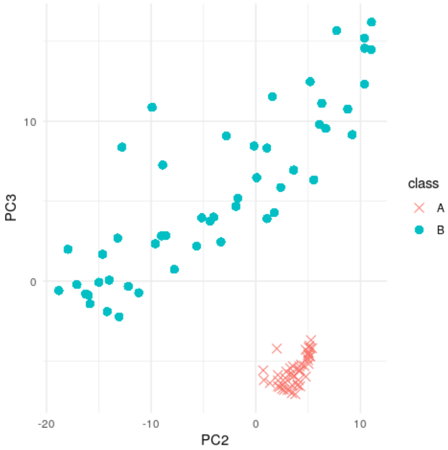
Is is important to note that PCA does not directly solve our task of class separation when reducing to 1D, as it is not a classification algorithm. However, as a transformation method, it helps create dimensions that facilitate finding a solution.



(c) *(2 p)* Apply kernel PCA (kPCA) with the quadratic kernel. Define the kernel and explain the transformation that occurs. Will this approach achieve our goal?

A kernel computes a similarity measure between two input vectors $\mathbf{x} = (x_1, y_1)$ and $\mathbf{x}' = (x_2, y_2)$ without explicitly mapping them into a higher-dimensional space. The quadratic kernel is defined as: $K(\mathbf{x}, \mathbf{x}') = \langle \mathbf{x}, \mathbf{x}' \rangle^2$. The kernel matrix $\mathbf{K}_{100\times100}$ contains the similarity among all the instance pairs. kPCA is performed through eigendecomposition of the kernel matrix.

The quadratic kernel implicitly maps data to a higher-dimensional intermediate feature space Φ such that: $K(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle$. For a 2D input, this corresponds to the explicit mapping to a 3D space $\phi(\mathbf{x}) = (x^2, \sqrt{2}xy, y^2)$ (but that is not directly done, we only work with the kernel matrix). This approach is thus not much different from the procedure ad (b). In the figure below you can see that the third principal component linearly separates the classes.



(d) *(2 p)* Apply kPCA with the RBF kernel. Define the kernel and explain the transformation that occurs. Will this approach achieve our goal?

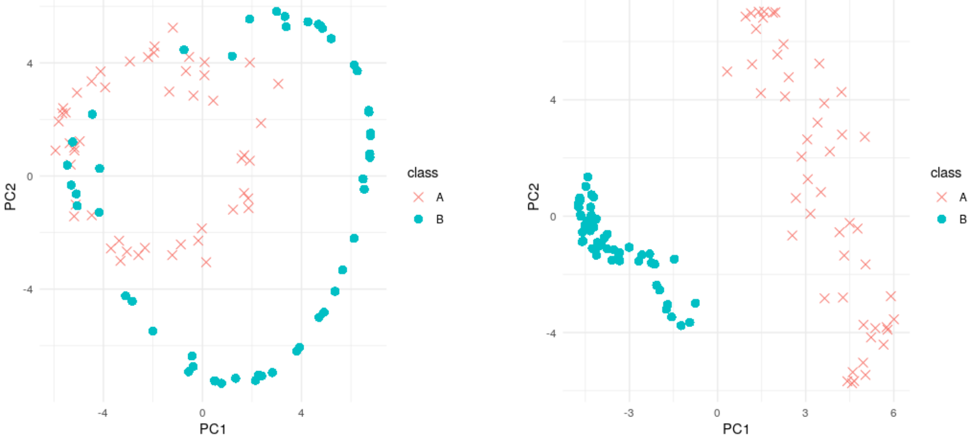
The Radial Basis Function (RBF) kernel, also known as the Gaussian kernel, is defined as:

$$K(\mathbf{x}, \mathbf{x}') = \exp\left(\frac{-\|\mathbf{x} - \mathbf{x}'\|^2}{2\sigma^2}\right)$$

The kernel measures the similarity between two points. Points closer in the input space will have a higher kernel value (closer to 1), while distant points will have a smaller kernel value (closer to 0). The parameter σ controls how "localized" the kernel is. Small values focus on local structures, capturing finer details, while large values blur differences, making the kernel capture broader relationships. The RBF kernel is an ideal choice for tasks like the concentric circles dataset. A well-tuned σ is crucial.

The RBF kernel implicitly maps data to an intermediate feature space with infinite dimension. In practice, the dimensionality of the transformed space is limited to the number of data points. For concentric circles, most variance will be captured by a small number of principal components, the rest will have zero or close to zero eigenvalues and can be discarded.

The figure in the bottom left shows the solution for $\sigma = 0.5$. Its value is too small, which leads to overfitting. The figure in the bottom right shows the solution for $\sigma = 2$. The main pattern is captured, the classes are separated in the first component, we achieved the optimal solution.



(e) *(2 p)* Generalize the problem of concentric circles and explain how kPCA might improve class separability.

kPCA transforms data into a higher-dimensional feature space where non-linear relationships between data points can become linear. By selecting an appropriate kernel (e.g., an RBF kernel), kPCA can project the data in such a way that the overlap between classes in the original space reduces or disappears in the transformed space. kPCA finds the principal components in the transformed space that maximize variance. These components capture the intrinsic geometry of the data better than the original axes and may also separate the classes. Different kernel functions (e.g., polynomial, RBF, or sigmoid kernels) allow customization to fit various non-linear patterns.

(f) *(2 p)* Explain how you would project a new instance $\mathbf{x_{new}}$ to the transformed space without the need to rerun the complete kPCA.

We have already run kPCA. Assume than m training points were available. The kernel matrix $\mathbf{K_{m\times m}}$, its eigenvectors α_k and eigenvalues λ_k are available ($k = 1 \dots m$). We only need to calculate the vector of kernel values between the new instance and each of the training points $\mathbf{x_i}$: $\mathbf{K_{new}} = (K(\mathbf{x_{new}}, \mathbf{x_1}), K(\mathbf{x_{new}}, \mathbf{x_2}), \dots, K(\mathbf{x_{new}}, \mathbf{x_m}))^T$. The projection on the k-th eigenvector gives the k-th coordinate in the transformed space: $t_{new,k} = \sum_{i=1}^{m} \alpha_k[i]\mathbf{K_{new}}[\mathbf{i}]$.

6 Robust statistics

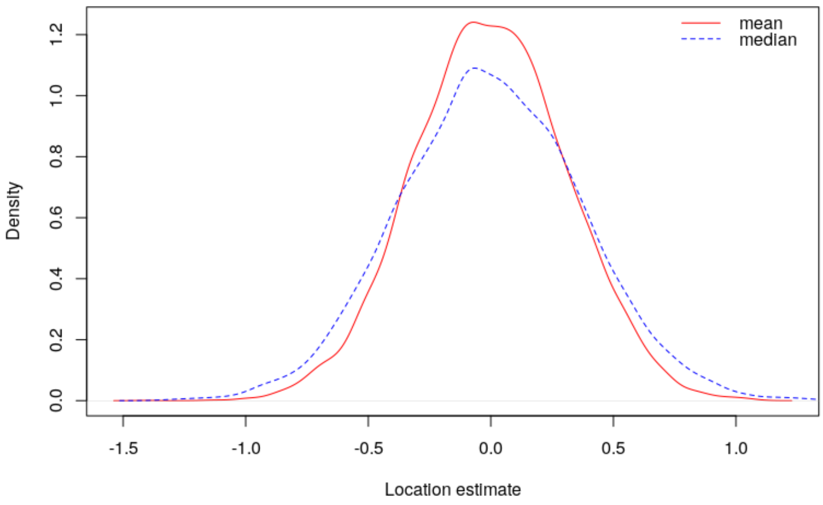
Problem 10. *(10 p) Answer the questions below.*

(a) *(3 p)* There is a sample of 10 observations $X_{10} = \{x_1, \dots, x_{10}\}$ drawn from a normal distribution $N(\mu, \sigma)$ with no outliers. You estimated the parameter μ with the aid of median, i.e. $\hat{\mu} = med(X_{10})$. Explain whether it is a good or bad estimator and why.

Sample median is an unbiased and consistent estimator of the population mean μ. It means that: 1) there is no difference between this estimator's expected value and the true value of μ, 2) for large samples, it will give the right value of μ. These are good properties.

At the same time, median is a location estimate that has a large breakdown point and low efficiency for normal distributions. The first characteristic says that it is robust towards outliers, however, there are no outliers here. The second property means that variance of the median is larger than variance of ML estimate for this type of distribution, which is sample mean $\frac{1}{n}\sum_{i=1}^{10} x_i$. To conclude, it is better to use sample mean, which is unbiased and consistent too and most efficient.

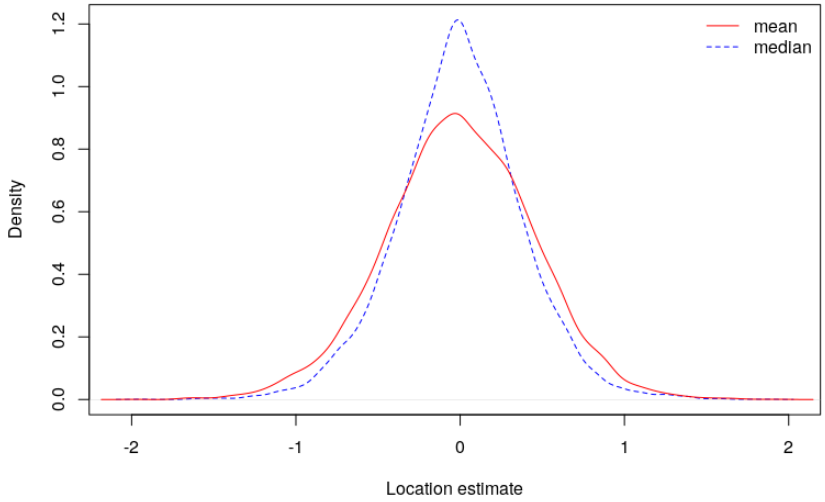
To compare the efficiency of the mean and the median, let us conduct the following experiment. We will draw a sample of size 10 from $N(0, 1)$ and calculate both the sample mean and the sample median. We will repeat this experiment 10,000 times and compare the distributions of the means and medians. The plot below demonstrates that both the distributions have a mean of zero (indicating they are unbiased), but the variance of the sample means is only about 3/4 of the variance of the medians. This indicates that smaller samples are needed to estimate μ using the sample mean as accurately as when using the sample median (accuracy includes both bias and variance).



(b) *(2 p)* There is a sample of 10 observations $X_{10} = \{x_1, \ldots, x_{10}\}$ drawn from a Laplace distribution $L(\mu, \sigma)$ with no outliers. You estimated the parameter μ with the aid of sample mean, i.e. $\hat{\mu} = \frac{1}{n}\sum_{i=1}^{10} x_i$. Explain whether it is a good or bad estimator and why.

The relationship between sample mean and sample median is inverse wrt the previous task. Sample mean is an unbiased and consistent location estimator, however, sample median is an unbiased and consistent location estimator too and it is most efficient on top of that.

Let us perform exactly the same experiment as we did in the previous task, we will only work with $L(0, 1)$. The plot below demonstrates that both the distributions have zero mean (they are unbiased), but the variance in sample medians is only about 3/4 of variance in sample means (median is in a blue dashed line). We need smaller samples to estimate μ with sample median as accurately as with sample mean.



(c) *(2 p)* Explain the term correlation. What is the difference between correlation and covariance? Which measures of correlation do you know?

Correlation is any statistical relationship between two random variables. It is synonymous with dependence. If we consider only linear relationship (the relationship can generally be non-linear too), defined by the Pearson's correlation coefficient ρ, correlation becomes very similar to covariance. It refers to the scaled form of covariance:

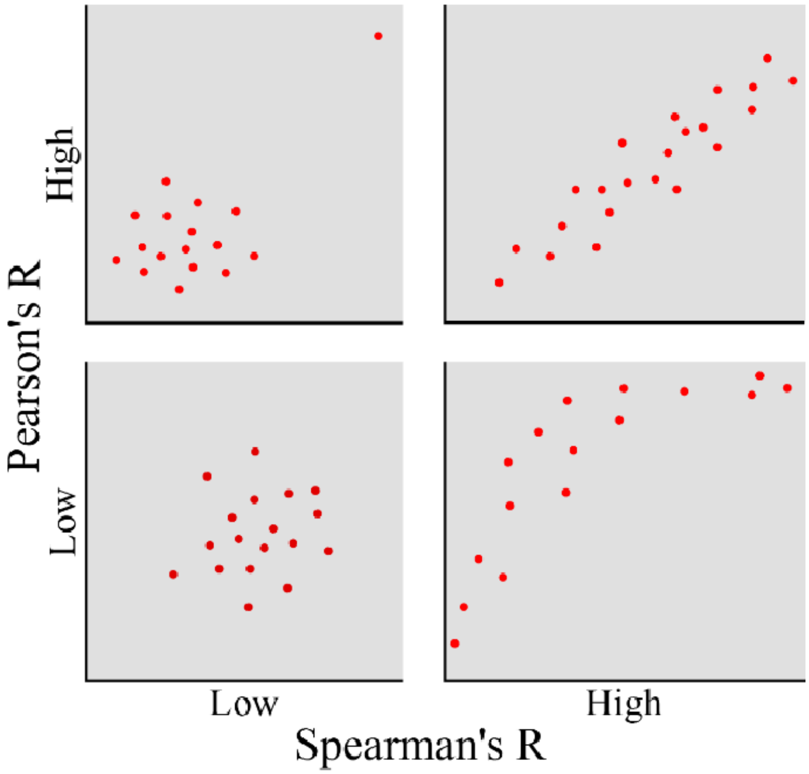
$cov(X, Y) = \sigma_{XY} = E[(X - \mu_X)(Y - \mu_Y)]$

$corr(X, Y) = \rho_{XY} = E[(X - \mu_X)(Y - \mu_Y)]/(\sigma_X \sigma_Y)$

The other well-known correlation coefficients (Spearman's, Kendall's) assess how well the relationship between two variables can be described using a monotonic function. They further generalize the Pearson's coefficient. Spearman's correlation can be calculated with the aid of ρ, we only deal with ranks instead of the raw values of random variables:

$r(X, Y) = \rho_{R(X)R(Y)}$

(d) *(3 p)* In the four bivariate plots below gradually show small sample problems where Pearson's correlation is large and Spearman's correlation is large, Pearson's correlation is small and Spearman's correlation is large, . . . Briefly describe each plot.



Non-parametric Spearman's correlation is more robust in case of outliers (the upper-left plot). It is also able to detect non-linear correlation, in addition to standard linear correlation (the bottom-right plot). Both the coefficients agree if there is no relationship between variables (the bottom-left plot), or there is a strong linear relationship (the upper-right plot). In the latter case, Pearson's coefficient could be advantageous as it has more statistical power than Spearman's coefficient (it needs a smaller sample to detect the linear relationship = reject the null hypothesis that there is no linear relationship between the variables, etc.).

7 Clustering

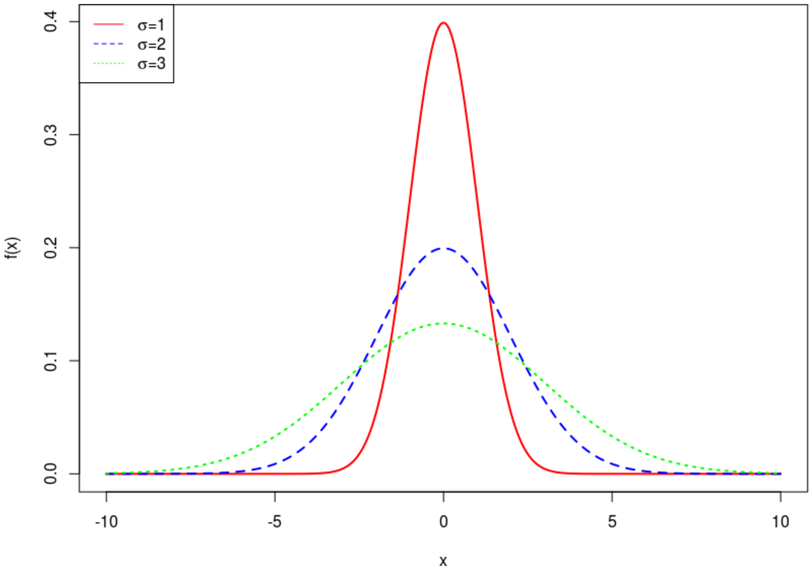
Problem 11. *(10 p) You have a blood test result for six people. The result of the test is a real number, here are the results: 10, 12, 13, 17, 18, 21. There are healthy and sick people, but you do not know who is who. All you know is that the test result has a normal distribution in both the healthy and sick populations. You also know that sick people have higher blood test values, both the populations have the same size. You want to decide on the statistical properties of the blood test and the division of people into groups using Expectation-Maximization (EM) clustering with the Gaussian Mixture Model (GMM).*

(a) *(1 p)* Initialize the EM clustering algorithm. Work with the values presented above.

EM clustering works with a latent variable that gives the probability of group given measurement. The first option to initialize EM clustering is to initialize the latent variable vector, for example: $p(C_h|x_i) = \langle 1,1,1,0,0,0 \rangle$ (the first three individuals are healthy, the rest of them are sick). In fact, we start with the E-step and do the first estimate of the expected value for each latent variable.

The algorithm also works with a GMM model that explains the data. We could propose the parameters of this model too, for example: $\hat{\mu}_h = 13$, $\hat{\sigma}_h = 2$, $\hat{\mu}_s = 17$, $\hat{\sigma}_s = 3$, $\alpha_h = \alpha_s = 0.5$ (the healthy group has the mean value of blood test 13 with standard deviation 2, the sick group has the mean value of blood test 17 with standard deviation 3, the populations are equiprobable). In this option, we imitate the first M-step that finds the model parameters maximizing the expected likelihood.

(b) *(3 p)* Perform the first step of the EM algorithm. Depending on the method of initialization, start with either E-step – the output will be a soft split of people into groups, or M-step – the output will be a model. Use the normal distributions in the figure below to estimate the probabilities. The calculation does not have to be detailed for all 6 samples . . .



Let us assume that we started with the model initialization. Then, the first E-step is as follows:

$x_1 = 10$, from the normPDF plots it follows that $f(x_1|C_h) \sim 0.065$ (the blue curve, the distance from mean is 3) and $f(x_1|C_s) \sim 0.009$ (the green curve, the distance from mean is 7),

$p(C_h|x_1) = f(x_1|C_h)/(f(x_1|C_h) + f(x_1|C_s)) = 0.065/0.074 = 0.88$

$p(C_s|x_1) = 1 - p(C_h|x_1) = 0.12$

$x_4 = 17$, from the normPDF plots it follows that $f(x_4|C_h) \sim 0.03$, (the blue curve, the distance from mean is 4) and $f(x_4|C_s) \sim 0.13$ (the green curve, the distance from mean is 0),

$p(C_h|x_4) = f(x_4|C_h)/(f(x_4|C_h) + f(x_4|C_s)) = 0.03/0.16 = 0.17$

$p(C_s|x_4) = 1 - p(C_h|x_4) = 0.83$

The full outcome is: $p(C_h|x_i) = \langle 0.88, 0.84, 0.78, 0.17, 0.07, 0 \rangle$.

(c) *(3 p)* Perform the second step of the EM algorithm (depending on the previous step, you will continue with either the M-step – the output will be a vector of model parameters, or the E-step – the output will be the soft split of people into groups).

We will carry on with the M-step, the model parameters will be calculated as the weighted average of the observations:

$\hat{\mu}_h = \frac{\sum_{i=1}^{6} p(C_h|x_i)x_i}{\sum_{i=1}^{6} p(C_h|x_i)} = \frac{0.88*10+0.84*12+0.78*13+0.17*17+0.07*18+0*21}{0.88+0.84+0.78+0.17+0.07+0} = 12.1,$

$\hat{\sigma}_h^2 = \frac{\sum_{i=1}^{6} p(C_h|x_i)(x_i-\hat{\mu}_h)^2}{\sum_{i=1}^{6} p(C_h|x_i)}$

$\hat{\sigma}_h^2 = \frac{0.88(10-12.1)^2+0.84(12-12.1)^2+0.78(13-12.1)^2+0.17(17-12.1)^2+0.07(18-12.1)^2}{0.88+0.84+0.78+0.17+0.07+0} = 4,$

$\hat{\mu}_s = \frac{\sum_{i=1}^{6} x_i p(C_s|x_i)}{\sum_{i=1}^{6} p(C_s|x_i)} = \frac{0.12*10+0.16*12+0.22*13+0.83*17+0.93*18+1*21}{0.12+0.16+0.22+0.83+0.93+1} = 17.7,$

$\hat{\sigma}_s^2 = \frac{\sum_{i=1}^{6} p(C_s|x_i)(x_i-\hat{\mu}_s)^2}{\sum_{i=1}^{6} p(C_s|x_i)}$

$\hat{\sigma}_s^2 = \frac{0.12(10-17.7)^2+0.16(12-17.7)^2+0.22(13-17.7)^2+0.83(17-17.7)^2+0.93(18-17.7)^2+1(21-17.7)^2}{0.12+0.16+0.22+0.83+0.93+1} = 8.8,$

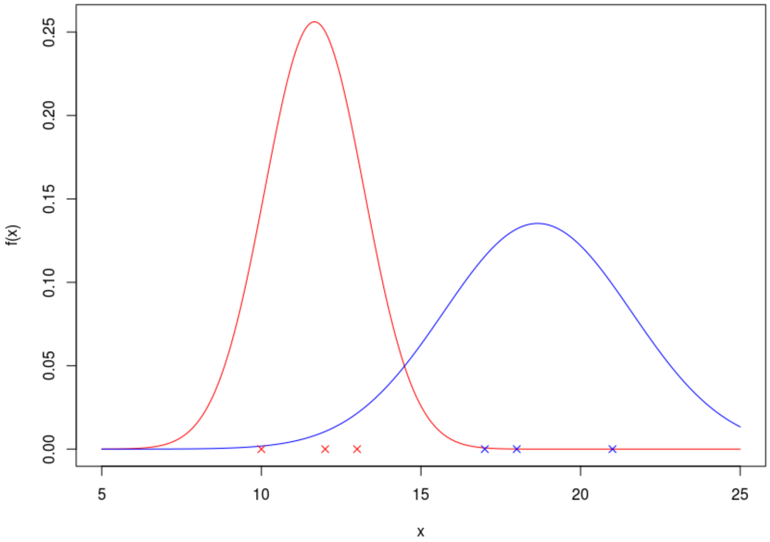
$\alpha_h = \alpha_s = 0.5$ remain unchanged.

(d) *(2 p)* Graphically and verbally describe the expected result of the EM algorithm at the end of its run (what the model will look like, how the objects will be divided into clusters).

The model is a mixture of two Gaussian distributions. The first element of the mixture represents healthy individuals, the second one stands for sick people. The model performs soft clustering, it splits the individuals between the groups probabilistically, see $p(C_h|x_i)$ and $p(C_s|x_i)$ calculated above. In order to perform hard clustering, we can simply compare $p(C_h|x_i)$ and $p(C_s|x_i)$ for each x_i and put x_i in the cluster with the higher posterior probability. Since $\alpha_h = \alpha_s = 0.5$, we can also compare $f(x_i|C_h)$ and $f(x_i|C_s)$ with the same result.

A very simple guess is that the first mixture element will represent the three smallest test outcomes (10, 12, 13), while the second element will stand for the three largest outcomes (17, 18, 21). Then, $\hat{\mu}_h$ would be 11.7 (the average of the three leftmost values), $\hat{\sigma}_h^2$ around 1.6 (the variance in the three leftmost values), $\hat{\mu}_s$ would be 18.7 (the average of the three rightmost values), $\hat{\sigma}_s^2$ around 2.9 (the variance in the three rightmost values).

The actual EM GMM outcome is as follows (calculated in R with Mclust function):



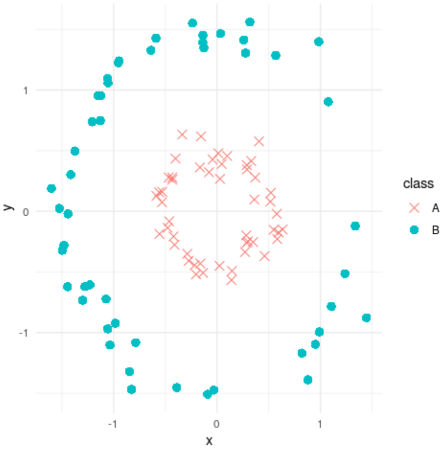
The real parameters are: $\hat{\mu}_h = 11.7$, $\hat{\sigma}_h^2 = 1.6$, $\hat{\mu}_s = 18.7$, $\hat{\sigma}_s^2 = 2.9$. Obviously, both the Gaussians consider the other test outcomes too, but their influence is insignificant due to negligible weights. The above-mentioned simple estimate was very accurate.

(e) *(1 p)* Explain the difference between EM GMM and quadratic discriminant analysis (QDA).

EM GMM learns in an unsupervised way. The input samples do not have to be annotated. The model is trained to optimally solve a density estimation task through maximization of the data likelihood. QDA learns in a supervised way, it assumes class assignments on its input and solves a discrimination task.

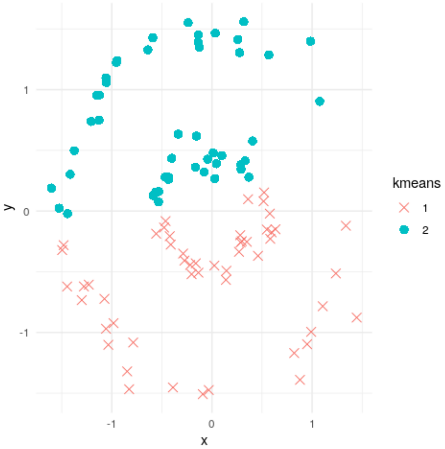
However, when having the GMM model reached with EM, we can use it in exactly the same way as we did in QDA. It has exactly the same form. In the task above, the decision boundary between classes would be around 14.5 (there is one more decision boundary for outcomes smaller than 3.5, but is uninteresting for the moment). In general, QDA is expected to show a higher classification accuracy, it deals with more information on its input. In our simple task, the outcome of both the methods (the mixture itself, the decision boundary) is nearly the same (assuming that the correct classification of our sample provided to QDA is ⟨healthy, healthy, healthy, sick, sick, sick⟩).

Problem 12. *(10 p) Let us consider a problem with two concentric circles in a two-dimensional space. The circles represent two different classes. There are 100 instances, with half belonging to class A and the other half to class B. See the picture below. Your task is to cluster the instances into two clusters that match the classes.*



(a) *(2 p)* Name a clustering method that will not solve the problem at all. Explain why.

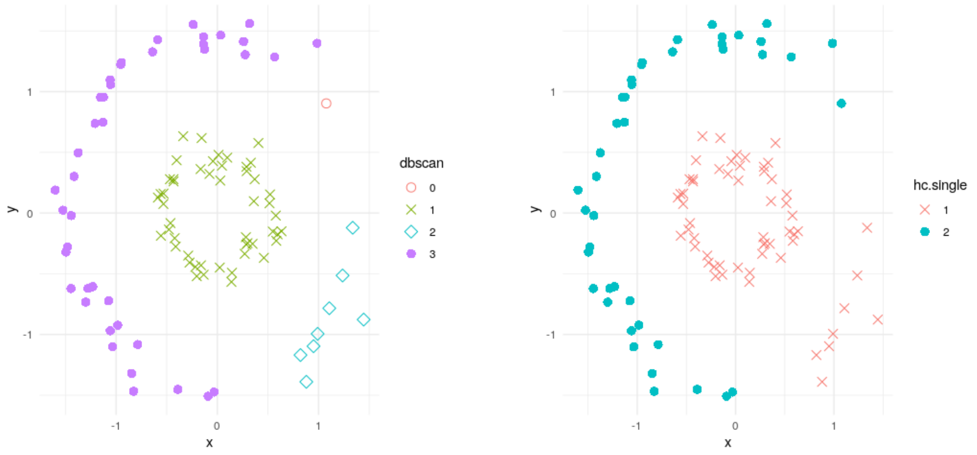
k-means relies on Euclidean distances and assigns points to clusters based on their proximity to cluster centroids. This implicitly assumes that clusters are approximately spherical or isotropic (equal variance in all directions) and separated by straight-line boundaries. The mismatch between k-means' assumptions and the data's true structure leads to poor performance. A typical k-means outcome is shown below. The data are commonly partitioned into spatially intuitive regions, such as left vs. right halves or top vs. bottom halves.



(b) *(2 p)* Name another clustering method that may solve the problem but the solution heavily depends on the method parameters and actual circle settings. Explain the relationships.

DBSCAN can correctly cluster two concentric circles under certain conditions. DBSCAN does not directly consider the geometric or global structure of the data, which is rather good in our specific problem. It operates based on local density and in our problem will work correctly if: 1) its parameters are properly selected (ϵ (neighborhood radius) parameter should be chosen such that points within each circle are considered neighbors but points from different circles are not) and the minPts (minimum number of points in a neighborhood to define a core point) is appropriately set, considering the density of the points in the circles, 2) there is clear density difference between circles (each circle forms a dense, well-separated cluster in the feature space, and the points between the circles are sparse), 3) data shows uniform distribution (the points within each circle are distributed evenly). One of the best possible solutions is shown in the left pane below (ϵ=0.5, minPts=4). It is considered as a partial solution to our problem, the main obstacle is that the outer circle is not dense enough. It is difficult if not impossible to find a parameter setting that generates two correct clusters with no noise (0 instances).

Single linkage hierarchical clustering determines clusters based on the smallest distance (or linkage) between points in different clusters. In our case it will work perfectly, if the distance between adjacent points within a circle is consistently smaller than the minimum distance between points from different circles. However, this condition is not met in our data set. The algorithm will either return three clusters similar to those returned by DBSCAN, or two clusters as shown in the right pane below.

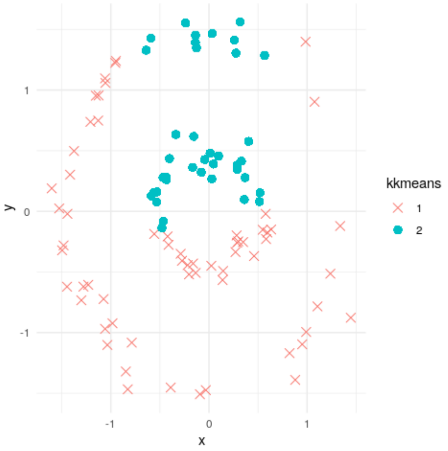


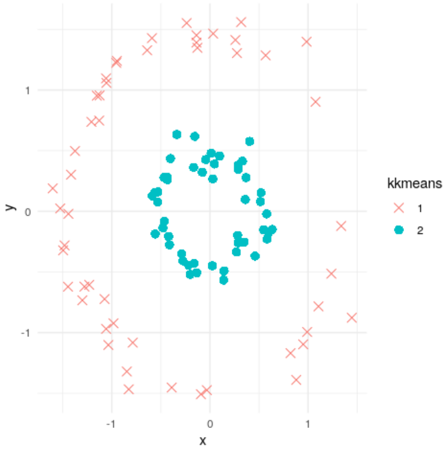
(c) *(2 p)* Would you recommend to apply kernel k-means in this task? Highlight its pros and cons. Which kernel would you use and what is the expected outcome?

In theory, **kernel k-means** can be a good choice for the task of concentric circles. A kernel may overcome the above-mentioned drawbacks of raw k-means. The main pros lie in non-linear separation (a suitable kernel allows the data to be mapped into a higher-dimensional space where the clusters become linearly separable), flexibility with kernel choice (we have already learned that the RBF kernel works well for tasks like this), and ability to capture complex cluster patterns (kernel k-means can model more complex relationships in the data compared to traditional k-means).

Cons of kernel k-means lie in kernel selection (the success of kernel k-means largely depends on selecting the right kernel), parameter tuning (the RBF kernel requires careful tuning of σ), computational complexity (the complexity of kernel matrix computation grows quadratically with the number of data points, in each iteration we need to sum up kernel values for all the data pairs from the same cluster) and sensitivity to outliers (similar to k-means).

In our task, the main difficulty is to find the optimal σ for the RBF kernel. The algorithm works well only for σs around 1.5 and only in around 50% of the runs (k-means algorithm is initialization dependent). The figure below exemplifies an incorrect outcome and the correct outcome for $\sigma = 1.5$.



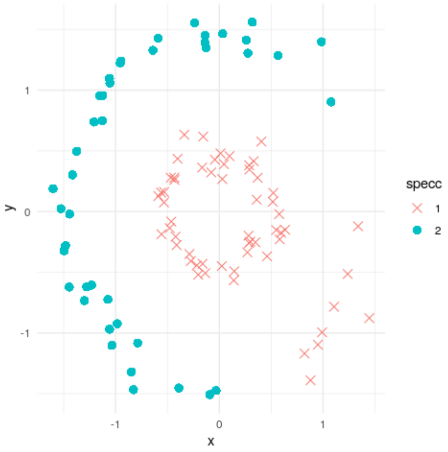


(d) *(2 p)* Would you recommend to apply spectral clustering in this task? Compare it with kernel k-means in terms of the algorithm as well as the outcome.

For two concentric circles, spectral clustering is generally recommended. Similarly to previous kernel methods in this task, it works well with the RBF kernel. The basic steps are: 1) construct the neighborhood graph (the goal is to densely connect the points within the same circle and keep the rest as much disconnected as possible), 2) construct the Laplacian matrix for the graph and find its eigenvectors, 3) run k-means in the space of a few smallest eigenvectors.

When compared to kernel k-means, spectral clustering is preferred due to its ability to leverage the graph structure to capture the global data topology. However, if computational efficiency is critical and you have a good understanding of kernel parameters, kernel k-means can also be a practical choice (as eigendecomposition is cubic with the number of data points).

The outcome of spectral clustering with the RBF kernel, 2 clusters and the default sigma parameter in the specc() R function is shown below.



(e) *(2 p)* Now, compare spectral clustering with kernel PCA. What are the similarities and differences? Propose one modification to make them more similar.

Both methods rely on a similarity matrix derived from a kernel (e.g., the RBF kernel) to capture non-linear relationships in the data. For the concentric circles, both methods can transform the data into a higher-dimensional space where the circles become linearly separable. Both methods excel in capturing the circular structure of the data, making them suitable for separating the two concentric circles. Both methods use eigenvalue decomposition of a matrix: kernel PCA decomposes the kernel matrix to find principal components in the transformed space, spectral clustering decomposes the Laplacian matrix (a function of the similarity matrix) to identify clusters.

The main difference is in their objectives. kernel PCA is a dimensionality reduction method it maps data to a new space to capture variance. It is a variance preserving algorithm. Spectral clustering groups data points into clusters and thus focuses on minimizing inter-cluster similarity and maximizing intra-cluster similarity. Technically, kernel PCA uses the largest eigenvectors of the kernel matrix while spectral clustering uses the smallest non-zero eigenvectors of the Laplacian matrix.

The application of k-means after kernel PCA would align the objectives of both the methods.