Compulsory exercise 1: Group X

TMA4268 Statistical Learning V2018

NN1, NN2 and NN3

Date when you hand in

Problem 2 - Linear regression

```
##
## Call:
## lm(formula = -1/sqrt(SYSBP) ~ ., data = data)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0207366 -0.0039157 -0.0000304  0.0038293  0.0189747
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.103e-01  1.383e-03 -79.745  < 2e-16 ***
## SEX         -2.989e-04  2.390e-04  -1.251 0.211176
## AGE          2.378e-04  1.434e-05  16.586  < 2e-16 ***
## CURSMOKE    -2.504e-04  2.527e-04  -0.991 0.321723
## BMI          3.087e-04  2.955e-05  10.447  < 2e-16 ***
## TOTCHOL      9.288e-06  2.602e-06   3.569 0.000365 ***
## BPMEDS       5.469e-03  3.265e-04  16.748  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.005819 on 2593 degrees of freedom
## Multiple R-squared:  0.2494, Adjusted R-squared:  0.2476
## F-statistic: 143.6 on 6 and 2593 DF,  p-value: < 2.2e-16
```

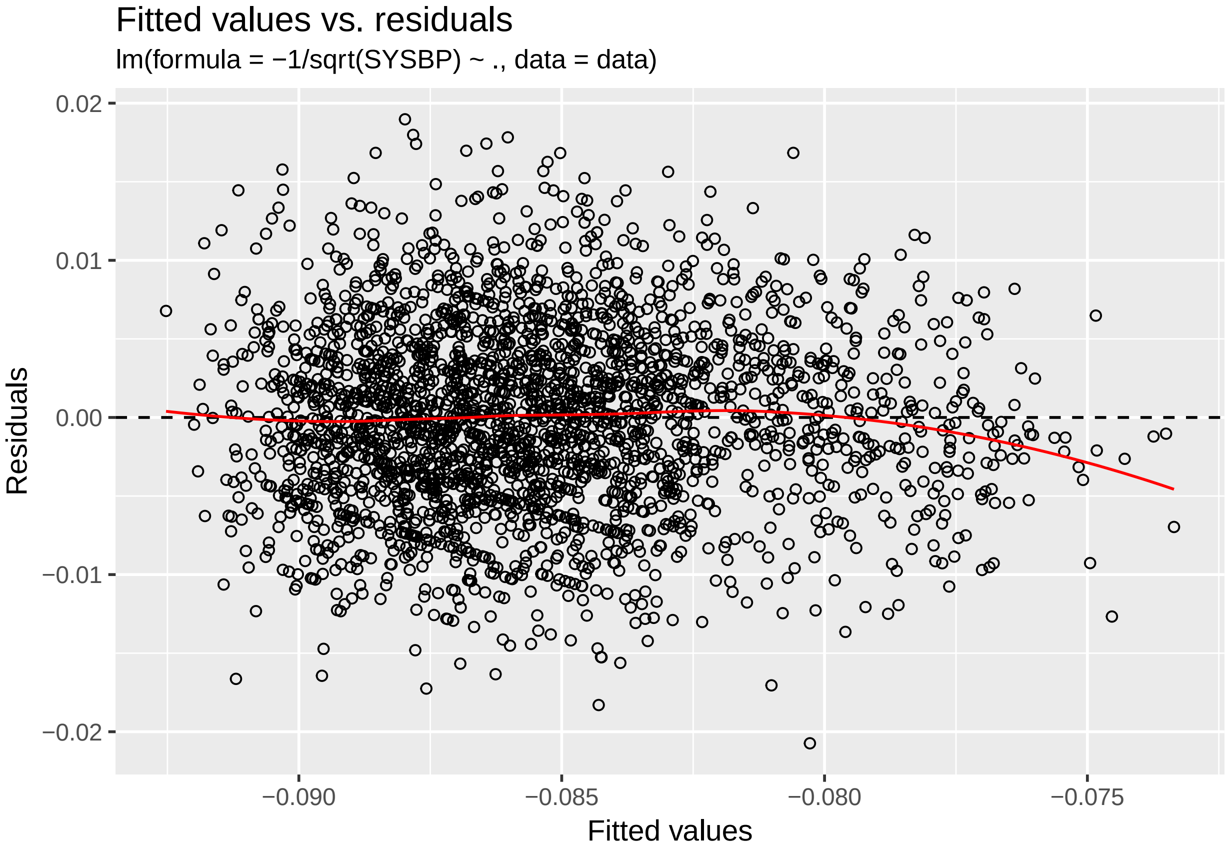
a)

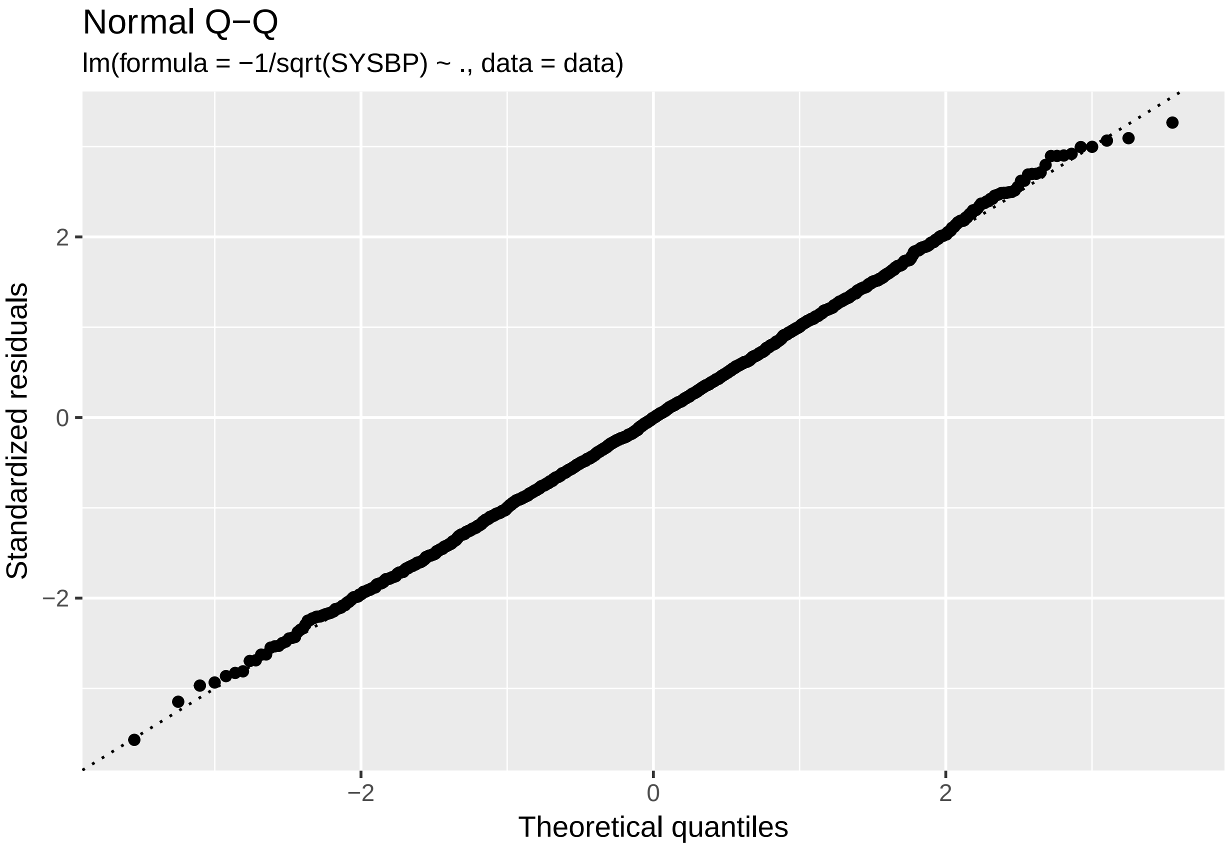
- The fittet model has the equation $\hat{Y} = X\hat{\beta}$, where $\hat{\beta} = (X^T X)^{-1} X^T Y$, X is the $n \times (p+1)$ design matrix, and Y is the corresponding response values.
- "Estimate" is the estimated coeffictients obtaining the minimum residual square error with the data set. The "intercept" is the constant term in the regression model.
- The "standard error" is the estimated standard deviation in the estimated coefficients. It is given as the square root of $V\hat{ar}(\hat{\beta_j}) = c_{jj}\hat{\sigma}^2$, where $c_{ij} = ((X^T X)^{-1})_{ij}$ and $\hat{\sigma}^2 = (Y - \hat{Y})^T (Y - \hat{Y})/(n-p-1)$.
- The "t value" is for every coefficent j, $\frac{\hat{\beta}_j}{\sqrt{c_{jj}}\hat{\sigma}}$ which is t distributed with $n-p-1$ degrees of freedom under the assumtion that H_0 is true, that is, β_j truly is 0. "Pr(t > |t|)" is then the probability of obeserving such an extreme t value given that H_0 is true. Hence Pr(>|t|) $:= P(|T_{n-p-1}| \geq |\frac{\hat{\beta}_j}{\sqrt{c_{jj}}\hat{\sigma}}|) = 2P(T_{n-p-1} \geq |\frac{\hat{\beta}_j}{\sqrt{c_{jj}}\hat{\sigma}}|)$.

- The "Residual standard error" is our estimate for the standard deviation of Y. The standard error squared is given as $\hat{\sigma}^2 = (Y - \hat{Y})^T(Y - \hat{Y})/(n - p - 1)$.
- The "F - statistic" is used to check the hypothesis of all betas being 0. In the table it is given as $\frac{(TSS-RSS)/p}{RSS/(n-p-1)}$, which is Fisher distributed with degrees of freedom p and $n - p - 1$, where $TSS := \sum_{i=1}^{n}(y_i - \bar{y})^2$, and $RSS := \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$.

b)

- The proportion of variability explained by the model is given by the R^2- statistic $:= (TSS-RSS)/TSS$, here being equal to 0.2494. Hence our model explains approximately 25% of the variance in the response value.

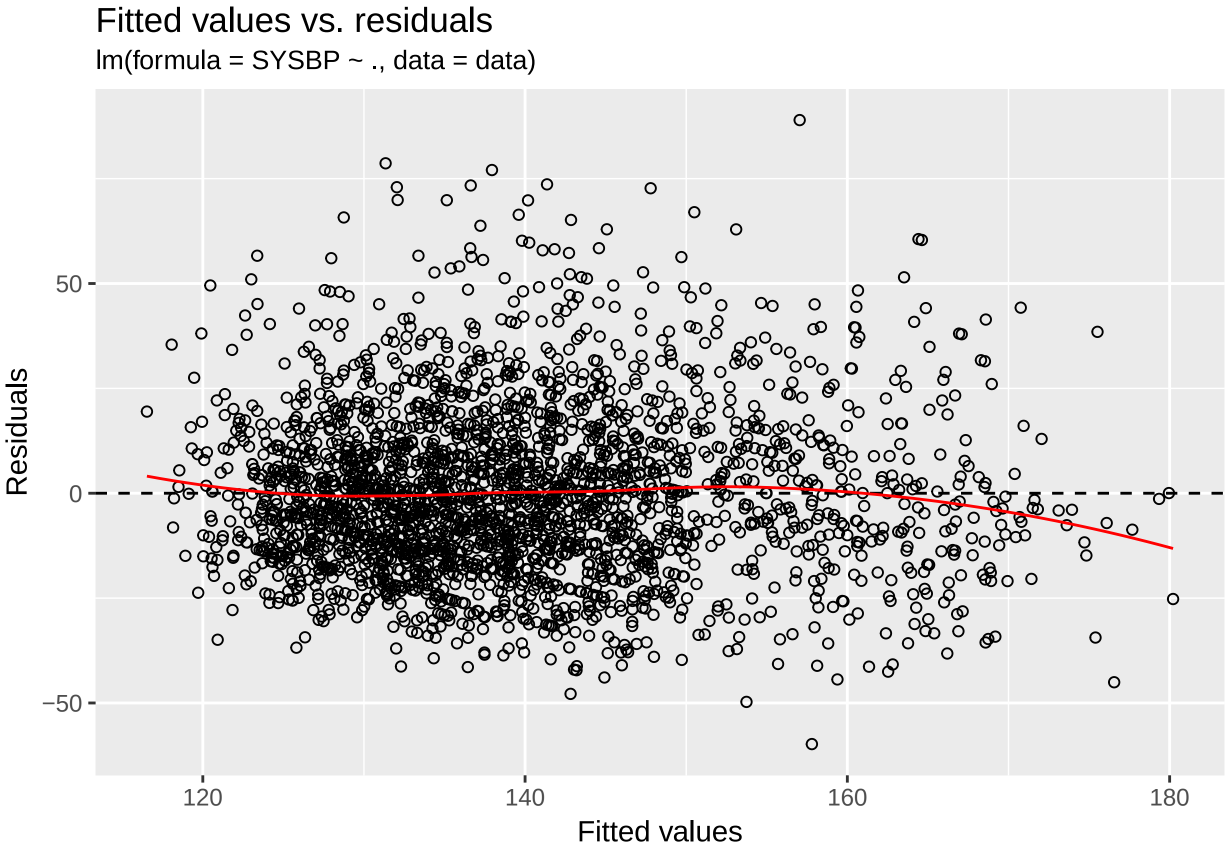


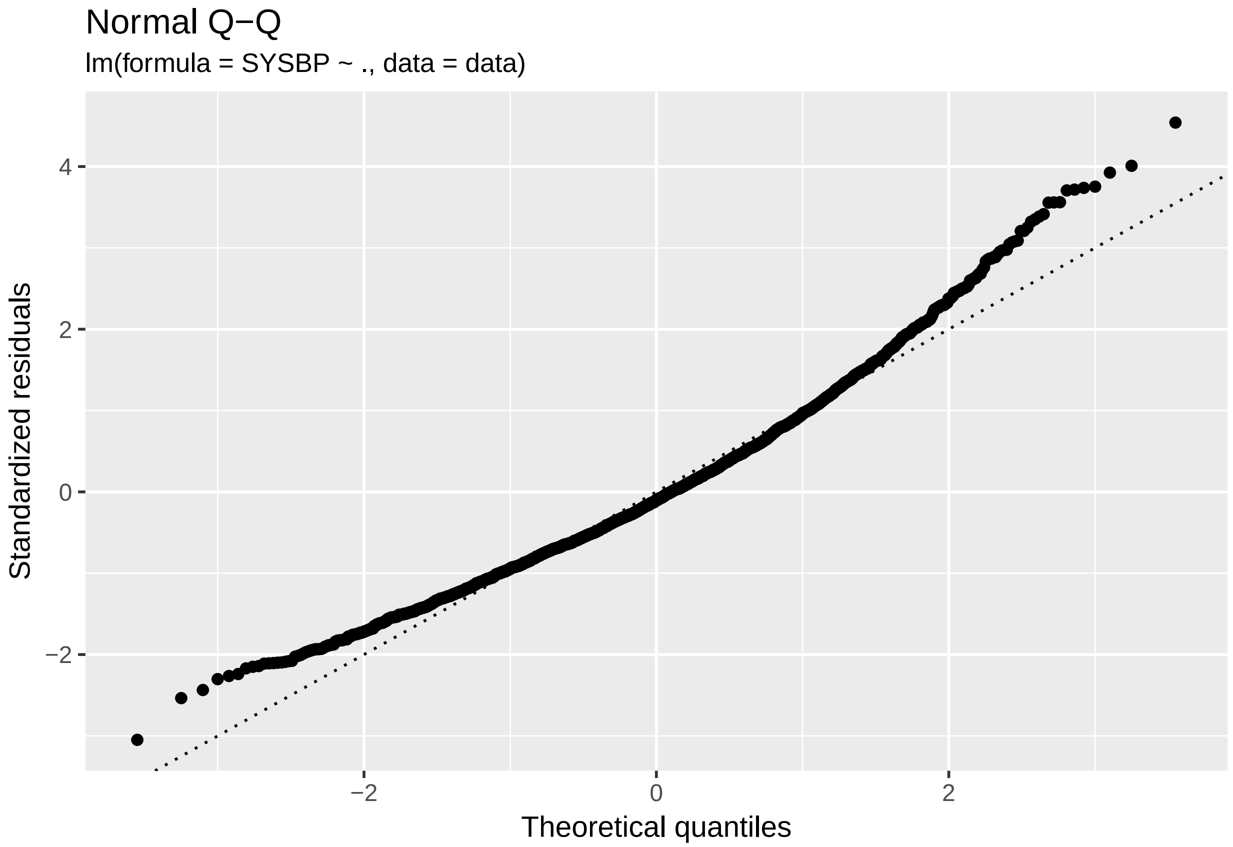


- Looking at the plot of residuals vs. fitted values we note that it does not appear to be a correlation between the value of the response and the variance of the response, and the mean appears to be 0. This fits well with the assumtion of the noise being normally distributed with mean 0 and constant variance.

  The QQ-plots strengthens our belief in this assumtion, as the points form a linear line.

```
##
## Call:
## lm(formula = SYSBP ~ ., data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -59.800 -13.471  -1.982  11.063  88.959
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 56.505170   4.668798  12.103  < 2e-16 ***
## SEX         -0.429973   0.807048  -0.533  0.59424
## AGE          0.795810   0.048413  16.438  < 2e-16 ***
## CURSMOKE    -0.518742   0.853190  -0.608  0.54324
## BMI          1.010550   0.099770  10.129  < 2e-16 ***
## TOTCHOL      0.028786   0.008787   3.276  0.00107 **
## BPMEDS      19.203706   1.102547  17.418  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19.65 on 2593 degrees of freedom
## Multiple R-squared:  0.2508, Adjusted R-squared:  0.249
## F-statistic: 144.6 on 6 and 2593 DF,  p-value: < 2.2e-16
```





- Looking at the diagnostic plots of model B we note that the values in the residuals vs. fitted values plot does not appear to be scattered evenly around the x-axis, where the greatest deviances appear to be in the postitive half plane. The QQ-plot also suggests that these residuals are not normally distributed, and thus the regression model fails, and the inference in the summary is not valid. We note that the RSE is considerably larger as well, but since the response is different, it is hard to directly compare. Clearly we prefer model A to make inference about systolic blood pressure, for this model follows the regression error assumptions, where model B fails.

c)

- The estimate for $\hat{\beta}_{BMI}$ is $3.087 \cdot 10^{-4}$.
- We interpret the estimated coefficient $\hat{\beta}_{BMI}$ as the coefficient of the variable containing the value of BMI in the linear expression for $-1/\sqrt{SYSBP}$, that is, the impact of change in BMI on the response

$$\hat{\beta}_{BMI} = \frac{\partial(-1/\sqrt{SYSBP})}{\partial BMI}$$

- Since $\hat{\beta}_{BMI} \sim N(\beta_{BMI}, \sigma^2 c_{BMI})$, where c_{BMI} := diagonal entry corresponding to BMI of $(X^T X)^{-1}$ we have

$$\frac{(\hat{\beta}_{BMI} - \beta_{BMI})/(\sigma\sqrt{c_{BMI}})}{\sqrt{\frac{1}{\sigma^2}RSS/(n-p-1)}} = \frac{\hat{\beta}_{BMI} - \beta_{BMI}}{\sqrt{\frac{RSS}{n-p-1}c_{BMI}}} \sim T_{n-p-1}$$

It follows that

$$Pr(\beta_{BMI} \in (\hat{\beta}_{BMI} - \hat{\sigma}\sqrt{c_{BMI}}t_{0.995,2593}, \hat{\beta}_{BMI} - \hat{\sigma}\sqrt{c_{BMI}}t_{0.005,2593})) = 0.99$$

Setting $t_{0.005,2593} = -2.577727$ and $t_{0.995,2593} = 2.577727$, we compute the interval to be $(2.325282 \cdot 10^{-4}, 3.848718 \cdot 10^{-4})$. This interval tells us that with probability 0.99, the true value of the coefficient is contained in this interval.

- We note that if H_0 is true, the center of the t distribution for prediction of $\hat{\beta}_{BMI}$ would be 0, but the degrees of freedom the same as for this prediction. Hence, a 99% prediction interval for the estimated coefficient would in this case be $(-|2.325282 \cdot 10^{-4} - 3.087 \cdot 10^{-4}|, |3.848718 \cdot 10^{-4} - 3.087 \cdot 10^{-4}|) = (-7.61718 \cdot 10^{-5}, 7.61718 \cdot 10^{-5})$. Clearly our observed value is outside the interval, meaning that the p value must be less than or equal to 0.01.

d)

- Model A predicts the response of these values to be -0.08667246, which corresponds to a SYSBP of 133.1183.
- Let $\tilde{Y_0}$ be a new observation of $-1/\sqrt{SYSBP}$ corresponding to the point x_0. Since we have $\tilde{Y_0} - {x_0}^T\beta \sim N(0, \sigma^2(1 + {x_0}^T(X^TX)^{-1}x_0))$ we get

$$\frac{(\tilde{Y_0} - {x_0}^T\hat{\beta})/(\sigma\sqrt{1 + {x_0}^T(X^TX)^{-1}x_0})}{\sqrt{\frac{1}{\sigma^2}RSS/(n-p-1)}} = \frac{\tilde{Y_0} - {x_0}^T\hat{\beta}}{\hat{\sigma}\sqrt{1 + {x_0}^T(X^TX)^{-1}x_0}} \sim T_{n-p-1}$$

letting $\tilde{Y_0} = -\frac{1}{\sqrt{Y_0}}$ we obtain the following prediction interval for SYSBP at x_0

$$Pr(Y_0 \in \left(\frac{1}{({x_0}^T\hat{\beta} + \hat{\sigma}kt_{0.05,2593})^2}, \frac{1}{({x_0}^T\hat{\beta} + \hat{\sigma}kt_{0.95,2593})^2}\right)) = 0.90, k = \sqrt{1 + {x_0}^T(X^TX)^{-1}x_0}$$

Setting $t_{0.05,2593} = -1.645441$ and $t_{0.95,2593} = 1.645441$ we compute the following prediction interval $(107.9250, 168.2845)$.

- This interval is very large numericaly but also in the state of the person having this blood pressure. It ranges from healty to close to lethal, and it is just a 90% prediction interval. In other words it is not particularly useful.

Problem 3 - Classification

a)

- We want to show that $\text{logit}(p_i) = \log(\frac{p_i}{1-p_i})$ is a linear function, where $p_i = \frac{e^{\beta_0+\beta_1x_{i1}+\beta_2x_{i2}}}{1+e^{\beta_0+\beta_1x_{i1}+\beta_2x_{i2}}}$. We see that

$$1 - p_i = 1 - \frac{e^{\beta_0+\beta_1x_{i1}+\beta_2x_{i2}}}{1+e^{\beta_0+\beta_1x_{i1}+\beta_2x_{i2}}} = \frac{1}{1+e^{\beta_0+\beta_1x_{i1}+\beta_2x_{i2}}}.$$

and thus

$$\text{logit}(p_i) = \log\left(\frac{p_i}{1-p_i}\right) = \log\left(\frac{\frac{e^{\beta_0+\beta_1 x_{i1}+\beta_2 x_{i2}}}{1+e^{\beta_0+\beta_1 x_{i1}+\beta_2 x_{i2}}}}{\frac{1}{1+e^{\beta_0+\beta_1 x_{i1}+\beta_2 x_{i2}}}}\right) = \log(e^{\beta_0+\beta_1 x_{i1}+\beta_2 x_{i2}}) = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2}.$$

So $\text{logit}(p_i)$ is linear.

```
##
## Call:
## glm(formula = y ~ ., family = "binomial", data = train)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.33682  -0.30197   0.07588   0.31184   2.64030
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   2.1431     4.4834   0.478 0.632645
## x1            0.3245     0.2156   1.505 0.132290
## x2           -1.9216     0.5165  -3.721 0.000199 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 90.094  on 64  degrees of freedom
## Residual deviance: 35.376  on 62  degrees of freedom
## AIC: 41.376
##
## Number of Fisher Scoring iterations: 6
```

- $\hat{\beta}_1$ and $\hat{\beta}_2$ can be interpreted as how the odds vary with x_{i1}, x_{i2} respectively. The odds is given as $\frac{p_i}{1-p_i}$. If the covariate x_{i1} is increased by one unit, the odds is multiplied by $\exp(\beta_1)$. The same is true for x_{i2} and $\exp(\beta_2)$. $hat\beta_i,\ i = 0, 1, 2$ and are estimates for the parameters β in the model, and are estimated by maximum likelihood on the training data.
- We find the formula for the class boundary by solving $\hat{Pr}(Y = 1|X) = 0.5$. This gives

  $$\frac{e^{\hat{\beta}_0+\hat{\beta}_1 x_{i1}+\hat{\beta}_2 x_{i2}}}{1+e^{\hat{\beta}_0+\hat{\beta}_1 x_{i1}+\hat{\beta}_2 x_{i2}}} = 0.5,$$

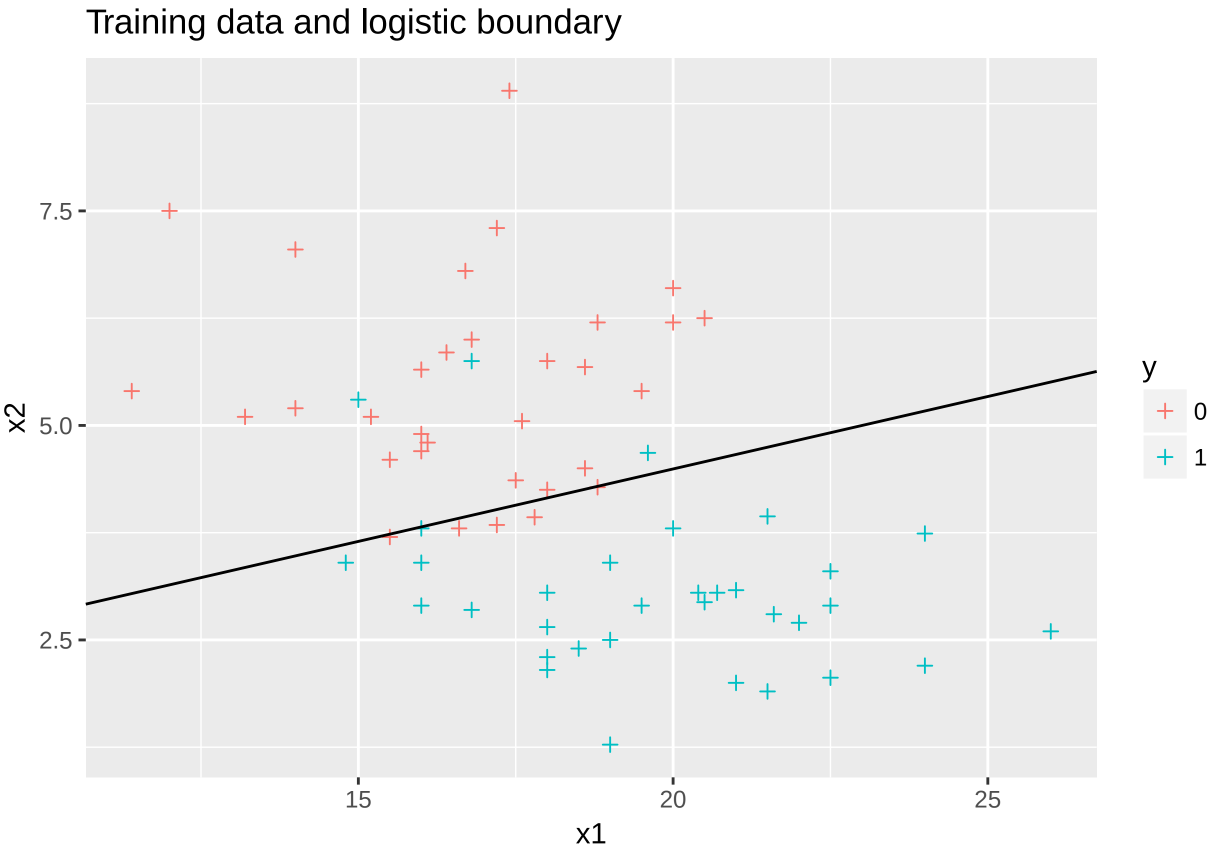
  so

  $$0.5e^{\hat{\beta}_0+\hat{\beta}_1 x_{i1}+\hat{\beta}_2 x_{i2}} = 0.5.$$

  This means that we need $\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \hat{\beta}_2 x_{i2} = 0$. Thus

  $$x_2 = -\frac{\hat{\beta}_0}{\hat{\beta}_2} - \frac{\hat{\beta}_1}{\hat{\beta}_2} x_1,$$

  and we see that the boundary is linear.
- The training data is plotted with the class boundary.



- From the summary we find that $\hat{\beta}_0 = 2.1431$, $\hat{\beta}_1 = 0.3245$, $\hat{\beta}_2 = -1.9216$. The probability of class 1 given $x_1 = 17$ and $x_2 = 3$ is then

$$p = \frac{e^{\hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2}}{1 + e^{\hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2}} = 0.8693.$$

  The interpretation of this is that based on the model, the probability of this point belonging to class one is 86.9%.

- The predicted probabilities for the test set is visualized in a confusion matrix with a cut-off of 0.5. The sensitivity is then $22/27 = 0.8148$, and specificity is $33/38 = 0.8684$. Since the sensitivity and specificity is high, the classification model appears to fit the dataset well. Perhaps a linear boundary is fitting.

```
##    testclass
##      0  1
##   0 22  5
##   1  5 33
```

b)

- The expression

$$P(Y = j|X = x_0) = \frac{1}{K} \sum_{i \in N_0} I(y_i = j)$$

returns the proporiton of the K nearest neighbours of x_0 belonging to class j. K is the number of neighbours we are considering, and the set N_0 contains all these neigbours. $I(y_i = j)$ is an indictor function taking the value 1 when the argument is true, and 0 otherwise.

- We have fitted the model below.
- For $K = 3$, the confusion table is shown below. The sensitivity and specificty is respectively $23/27 = 0.8519$ and $29/38 = 0.7632$. It scores a bit lower than classification by logistic regression, but it appears to perform ok.

```r
knn3 = knn(train = train[,-1], test = test[,-1], k = 3, cl = train$y, prob = FALSE)
t <- table(test$y, knn3)
t
```

```
##    knn3
##      0  1
##   0 23  4
##   1  9 29
```

- We repeat with $K = 9$. The confusion table is shown below. The sensitivity and specificity is now respectively $23/27 = 0.8519$ and $31/38 = 0.8158$. This model has higher sensitivity and specificity than the 3-nearest neighbour model, and we therefore prefer this one. If we choose K very small, we risk overfitting the data, and with very large K the model is possibly not flexible enough. We therefore search for the optimal K.

```r
knn9 = knn(train = train[,-1], test = test[,-1], k = 9, cl = train$y, prob = FALSE)
t <- table(test$y, knn9)
t
```

```
##    knn9
##      0  1
##   0 23  4
##   1  7 31
```

c)

- π_k is the probability of an observation being from class k, that is, the probability of getting a sample from a certain wine, wine 1 or 2 in this case. μ_k is the expected value of a point from class k, in our case the expected values of $(x1, x2)$ corresponding to wine 1 and wine 2. Σ is the variance matrix of the distribution of a class, here assumed to be equal for every class. Hence we assume that the variance in observations of $(x1, x2)$ are the same for both wine 1 and 2. $f_k(x)$ is the distribution of points $(x1, x2)$, coming from class k, i. e. wine 1 and 2, which we assume takes the form of the normal distribution with mean μ_k and variance Σ.
- To estimate π_k we consider the proportion of observations coming from class k, that is, $\hat{\pi}_k = \frac{n_k}{n}$. We compute $\hat{\pi}_1 = 32/65 = 0.4923$ and $\hat{pi}_2 = 33/65 = 0.5077$. To estimate μ_k we consider the estimated mean of points coming from class k, that is $\hat{\mu}_k = \frac{1}{n_k}\sum_{i,y_i=k} x_i$, which we compute to be: $\hat{\mu}_1 = (16.7781, 5.4575)^T$, $\hat{\mu}_2 = (19.6879, 3.0536)^T$. To estimate Σ we consider the estimated variance for each class, $\hat{\Sigma}_k := \frac{1}{n_k - 1}\sum_{i,y_i=k}(X_i - \hat{\mu}_k)(X_i - \hat{\mu}_k)^T$, and compute:

$$\hat{\Sigma} = \sum_{k=1}^{2} \frac{n_k - 1}{n - 2}\hat{\Sigma}_k = \begin{bmatrix} 6.2014 & -0.4447 \\ -0.4447 & 1.1678 \end{bmatrix}$$

- The desicion boundary is given by the equality $P(Y = 0|X) = P(Y = 1|X)$, that is

$$\frac{\pi_0 f_0(x)}{\sum_{i=0}^{1} \pi_k f_i(x)} = \frac{\pi_1 f_1(x)}{\sum_{i=0}^{1} \pi_k f_i(x)}$$

which simplifies to

$$\pi_0 \exp(-\frac{1}{2}(x-\mu_0)^T \Sigma^{-1}(x-\mu_0)) = \pi_1 \exp(-\frac{1}{2}(x-\mu_1)^T \Sigma^{-1}(x-\mu_1))$$

taking the logarithm on both sides yields

$$\log(\pi_0) - \frac{1}{2}({\mu_0}^T \Sigma^{-1} \mu_0 - 2{\mu_0}^T \Sigma^{-1} x + x^T \Sigma^{-1} x) = \log(\pi_1) - \frac{1}{2}({\mu_1}^T \Sigma^{-1} \mu_1 - 2{\mu_1}^T \Sigma^{-1} x + x^T \Sigma^{-1} x)$$

and finally

$$\log(\pi_0) - \frac{1}{2}{\mu_0}^T \Sigma^{-1} \mu_0 + {\mu_0}^T \Sigma^{-1} x = \delta_0(x) = \log(\pi_1) - \frac{1}{2}{\mu_1}^T \Sigma^{-1} \mu_1 + {\mu_1}^T \Sigma^{-1} x = \delta_1(x)$$

- We note that this descision rule is the same as classifying to the class with highest probability. Let $\hat{\delta}_k(x) := \log(\hat{\pi_k}) - \frac{1}{2}{\hat{\mu_k}}^T \Sigma^{-1} \hat{\mu_k} + {\hat{\mu_k}}^T \hat{\Sigma}^{-1} x$. And so, by the previous task, we have

  $$\hat{\delta}_0(x) = \hat{\delta}_1(x)$$

  and the boundary becomes

  $$\log(\hat{\pi_0}) + \frac{1}{2}{\hat{\mu_0}}^T \Sigma^{-1} \hat{\mu_0} - {\hat{\mu_0}}^T \hat{\Sigma}^{-1} x = \log(\hat{\pi_1}) + \frac{1}{2}{\hat{\mu_1}}^T \Sigma^{-1} \hat{\mu_1} - {\hat{\mu_1}}^T \hat{\Sigma}^{-1} x$$

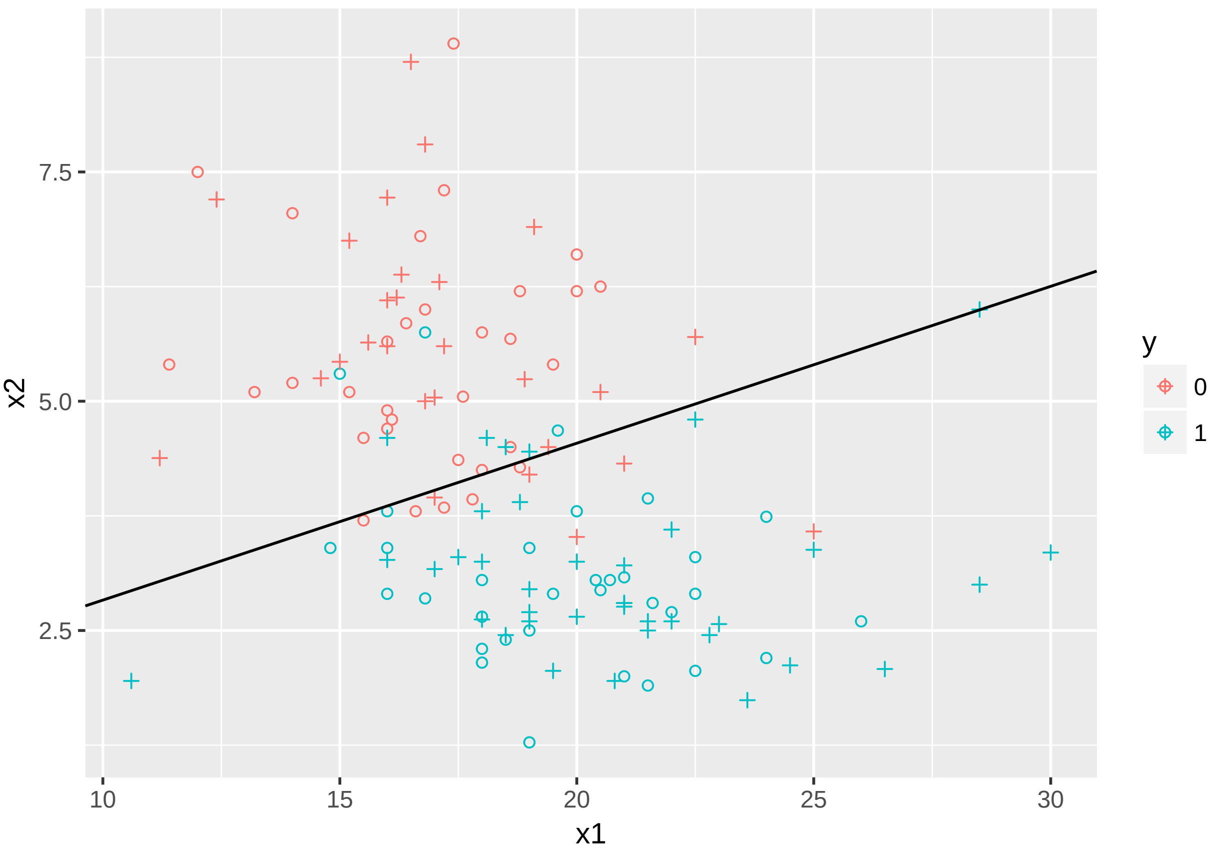
  that is

  $$\frac{1}{2}(\hat{\mu}_0^T \hat{\Sigma}^{-1} \hat{\mu}_0 - \hat{\mu}_1^T \hat{\Sigma}^{-1} \hat{\mu}_1) + \log\left(\frac{\hat{\pi}_0}{\hat{\pi}_1}\right) + (\hat{\mu}_1^T \hat{\Sigma}^{-1} - \hat{\mu}_0^T \hat{\Sigma}^{-1})x = 0$$

  Inserting our estimates from the training set we get the boudary

  $$x_2 = 0.1711 x_1 + 1.1205$$

- The descision boundary with both the training and test observations is shown below (circles are from the training set)



- Done below

```
wine_lda <- lda(y ~ x1 + x2, data = train)
```

- The confusion table is show below. We get a sesitivity of $22/27 = 0.8148$, and a specificity of $34/38 = 0.8947$. The performance is good compared to the logistic regression and KNN. Perhaps the data fit a linear model, and the normal distribution assumtion is not so far off.

```
##    predicted
##      0  1
##   0 22  5
##   1  4 34
```

- The most important diffence in regard to using LDA or QDA would be that with QDA we expect the variance of the classes to be different, and hence use different covariance matrices in their distributions. This allows for a more flexible fit to the data.

d)

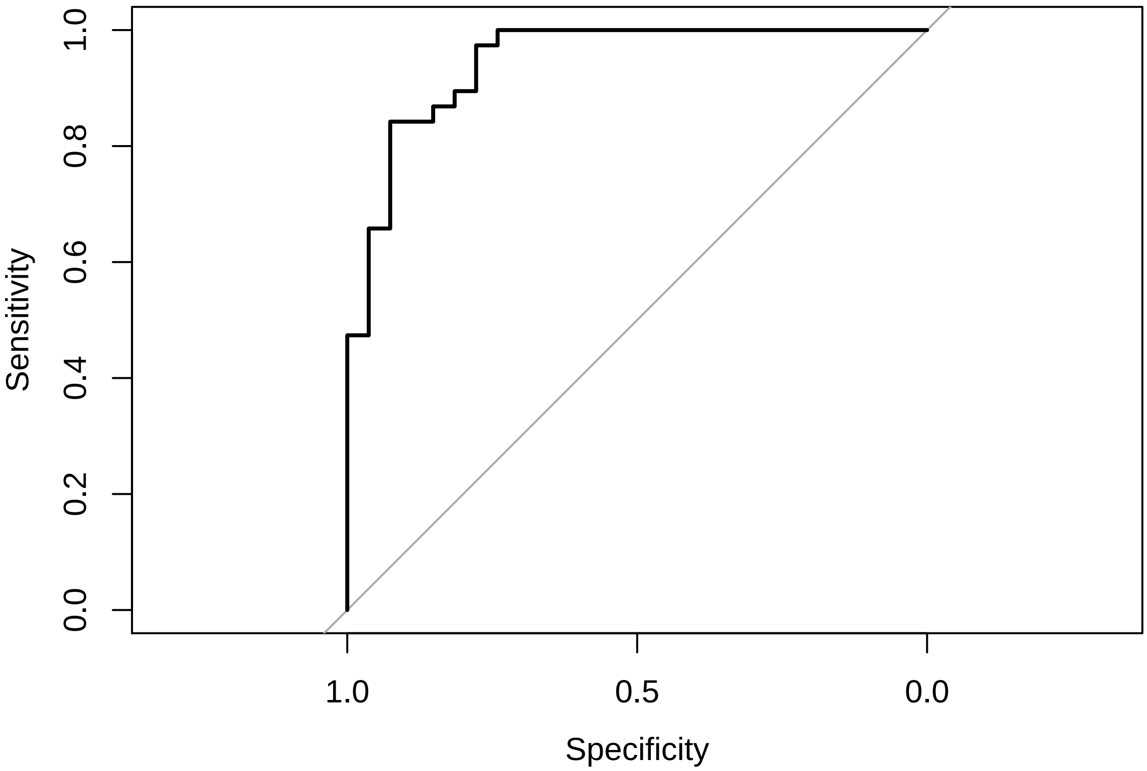
- To get in indication of the performance of the different classification methods, we list their sensitivity and specificity

```
##                       Sensitivity Specificity
## Logistic regression     0.8148148   0.8684211
## KNN (K = 9)             0.8518519   0.8157895
## LDA                     0.8148148   0.8947368
```

We note that over all, the methods based on a linear descision boundary scores the best, but the 9-nearest neighbour is not far behind. The highest scoring is the LDA, and based on this table, this would be the preferred method. But since the race is so even it would be wrong to take any stong stance.

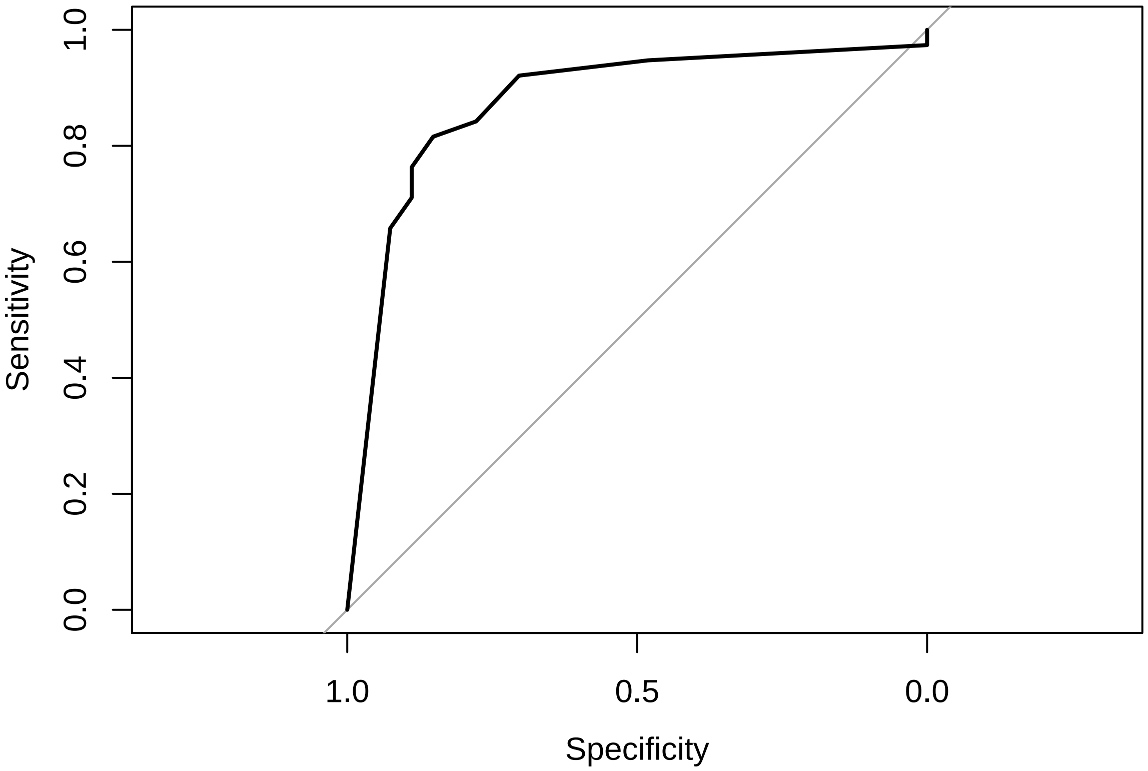
- A reciever operating characteristics curve is plot of sensitivity vs. specificity for a classifier, as we let the cut-off take on values in $[0, 1]$. An ROC curve is useful, since it indicates if there excist a particular cut-off giving both high sensitivity and specificity, that is, if the curve takes on values in the top left corner of the plot. Below are ROC plots for the three classifiers. We note that all three classifiers have points on their ROC curve close to the top left corner, but the two linear descision boundaries get closest. These methods also have the highest AUC value, that is, the area under the ROC curve, which suggests they are the over-all best classifiers for general cut-off.

```
## [1] "Logistic regression"
```

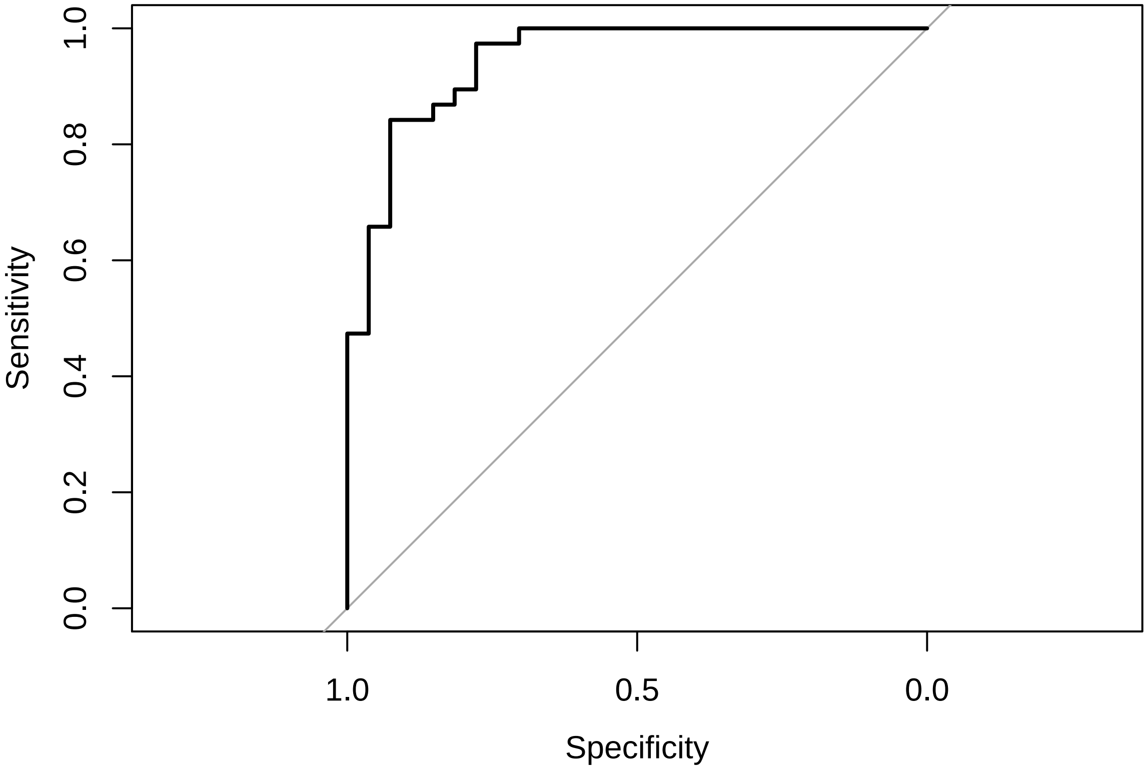


```
## Area under the curve: 0.9464
```

```
## [1] "KNN (K = 9)"
```



```
## Area under the curve: 0.8757
## [1] "LDA"
```



```
## Area under the curve: 0.9454
```