Exercise 10 (2025) — Advanced Methods for Regression and Classification

Olesia Galynskaia 12321492

2025-12-16

Loading and observing data

```
data("Diabetes")

str(Diabetes)
```

```
## 'data.frame':    403 obs. of  22 variables:
##  $ id      : int  1000 1001 1002 1003 1005 1008 1011 1015 1016 1022 ...
##  $ chol    : int  203 165 228 78 249 248 195 227 177 263 ...
##  $ stab.glu: int  82 97 92 93 90 94 92 75 87 89 ...
##  $ hdl     : int  56 24 37 12 28 69 41 44 49 40 ...
##  $ ratio   : num  3.6 6.9 6.2 6.5 8.9 ...
##  $ glyhb   : num  4.31 4.44 4.64 4.63 7.72 ...
##  $ location: Factor w/ 2 levels "Buckingham","Louisa": 1 1 1 1 1 1 1 1 1 1 ...
##  $ age     : int  46 29 58 67 64 34 30 37 45 55 ...
##  $ gender  : Factor w/ 2 levels "female","male": 1 1 1 2 2 2 2 2 2 1 ...
##  $ height  : int  62 64 61 67 68 71 69 59 69 63 ...
##  $ weight  : int  121 218 256 119 183 190 191 170 166 202 ...
##  $ frame   : Factor w/ 3 levels "large","medium",..: 2 1 1 1 2 1 2 2 1 3 ...
##  $ bp.1s   : int  118 112 190 110 138 132 161 NA 160 108 ...
##  $ bp.1d   : int  59 68 92 50 80 86 112 NA 80 72 ...
##  $ bp.2s   : int  NA NA 185 NA NA NA 161 NA 128 NA ...
##  $ bp.2d   : int  NA NA 92 NA NA NA 112 NA 86 NA ...
##  $ waist   : int  29 46 49 33 44 36 46 34 34 45 ...
##  $ hip     : int  38 48 57 38 41 42 49 39 40 50 ...
##  $ time.ppn: int  720 360 180 480 300 195 720 1020 300 240 ...
##  $ bmi     : num  22.1 37.4 48.4 18.6 27.8 ...
##  $ dtest   : chr  "-" "-" "-" "-" ...
##  $ whr     : num  0.763 0.958 0.86 0.868 1.073 ...
```

```
Diabetes$y <- ifelse(Diabetes$dtest == "+", 1, 0)
table(Diabetes$y, useNA = "ifany")
```

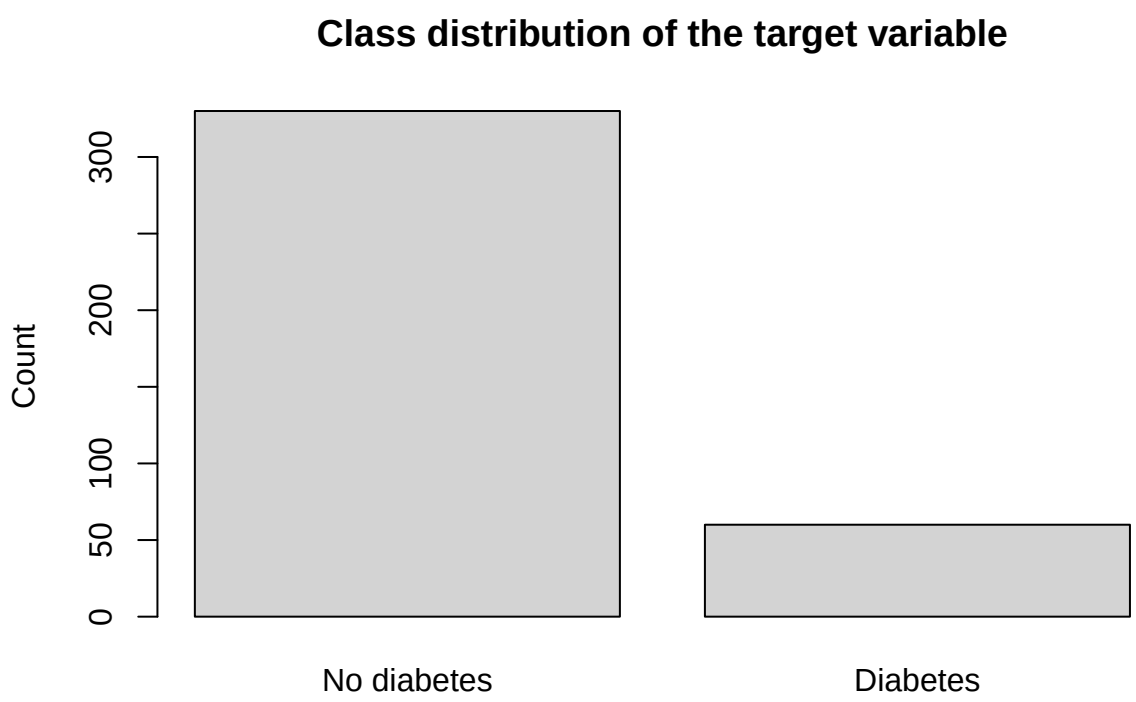
```
##
##    0    1 <NA>
##  330   60   13
```

```r
prop.table(table(Diabetes$y))
```

```
##
##         0         1
## 0.8461538 0.1538462
```

```r
taby <- table(Diabetes$y)

barplot(
  taby,
  names.arg = c("No diabetes", "Diabetes"),
  col = "lightgray",
  ylab = "Count",
  main = "Class distribution of the target variable"
)
```



Comment

Before fitting the models, I briefly explored the Diabetes dataset to understand its structure and the target variable.
The dataset contains 403 observations with clinical measurements such as cholesterol, stabilized glucose, HDL, BMI, age, and gender.

The original target variable dtest is a character variable indicating whether the diabetes test is positive or negative.
I converted it into a binary target y, where y = 1 represents a positive diabetes test and y = 0 a negative one. Missing values were kept as NA and handled later during model fitting.

The class distribution is imbalanced, with about 85% non-diabetic and 15% diabetic observations, as shown by the bar plot.
To avoid biased model evaluation, a stratified train/test split is used in the subsequent analysis.

Data preparation

```
vars <- c("y", "chol", "stab.glu", "hdl", "bmi", "age", "gender")
dat <- Diabetes[, vars]
dat$gender <- factor(dat$gender)

dat <- na.omit(dat)

set.seed(12321492)

idx0 <- which(dat$y == 0)
idx1 <- which(dat$y == 1)

train0 <- sample(idx0, size = floor(0.75 * length(idx0)))
train1 <- sample(idx1, size = floor(0.75 * length(idx1)))

train_id <- c(train0, train1)

train <- dat[train_id, ]
test  <- dat[-train_id, ]
```

Comment

I selected the predictor variables specified in the assignment and removed observations with missing values to ensure consistent model fitting.
Because the target classes are imbalanced, I created a stratified train/test split by randomly selecting about three quarters of the observations from each class for the training set.
This ensures that both classes are represented in the training and test data.

Ex-1 Logistic regression

```
logit_fit <- glm(
  y ~ chol + stab.glu + hdl + bmi + age + gender,
  data = train,
  family = "binomial"
)

summary(logit_fit)
```

```
##
## Call:
## glm(formula = y ~ chol + stab.glu + hdl + bmi + age + gender,
##     family = "binomial", data = train)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -10.837246   2.312841  -4.686 2.79e-06 ***
## chol          0.004053   0.006381   0.635  0.52531
## stab.glu      0.034151   0.005549   6.155 7.52e-10 ***
## hdl          -0.017442   0.015509  -1.125  0.26077
## bmi           0.076713   0.037461   2.048  0.04058 *
## age           0.052422   0.017148   3.057  0.00224 **
## gendermale   -1.067321   0.618978  -1.724  0.08465 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 242.14  on 285  degrees of freedom
## Residual deviance: 110.58  on 279  degrees of freedom
## AIC: 124.58
##
## Number of Fisher Scoring iterations: 7
```

```
prob_test <- predict(logit_fit, newdata = test, type = "response")
pred_test <- ifelse(prob_test > 0.5, 1, 0)
```

```
cm <- table(
  Actual = test$y,
  Predicted = pred_test
)

cm
```

```
##       Predicted
```

```
## Actual  0  1
##      0 82  0
##      1  8  7
```

```
miscl <- mean(pred_test != test$y)
miscl
```

```
## [1] 0.08247423
```

Comment

I fitted a logistic regression model using chol, stab.glu, hdl, bmi, age, and gender.
Stabilized glucose has a strong and highly significant positive effect on the probability of diabetes. BMI and age are also significant, while cholesterol and HDL are not. Gender shows only a weak effect.

Using a threshold of 0.5, the model classifies all non-diabetic cases correctly but misses some diabetic cases.
The overall misclassification rate is about 8.2%, showing good accuracy, but weaker performance for the minority diabetes class.

Ex-2

2(a) Fit a GAM

```
library(mgcv)

gam_fit <- gam(
  y ~ s(chol) + s(stab.glu) + s(hdl) + s(bmi) + s(age) + gender,
  data = train,
  family = "binomial",
  method = "REML"
)

summary(gam_fit)
```

```
##
## Family: binomial
## Link function: logit
##
## Formula:
## y ~ s(chol) + s(stab.glu) + s(hdl) + s(bmi) + s(age) + gender
##
## Parametric coefficients:
```

```
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -4.0858     0.7775  -5.255 1.48e-07 ***
## gendermale   -0.8234     0.7466  -1.103     0.27
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##               edf Ref.df Chi.sq  p-value
## s(chol)     2.976  3.722  8.094   0.0772 .
## s(stab.glu) 2.846  3.466 36.554 3.58e-07 ***
## s(hdl)      2.213  2.773  3.049   0.3506
## s(bmi)      1.000  1.000  2.480   0.1153
## s(age)      2.397  3.001  6.119   0.1067
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.679   Deviance explained = 69.8%
## -REML = 46.332  Scale est. = 1         n = 286
```

Comment

From the model output, the smooth term for stab.glu is clearly significant ($p < 0.001$) and has an effective degrees of freedom (edf) above 1, suggesting a non-linear relationship with diabetes risk.
The other smooth terms (chol, hdl, bmi, age) are not significant at the 5% level, although chol is borderline (p 0.08).
The parametric effect of gender is also not significant. Overall, the model explains about 70% of the deviance (deviance explained 69.8%).

2(b) Significance and complexity of smooth terms

Comment

In the GAM, the only clearly significant smooth term is s(stab.glu) ($p < 0.001$).
Its effective degrees of freedom (edf 2.85) indicate a moderately non-linear effect on the probability of diabetes.

The smooth terms s(chol), s(hdl), s(bmi), and s(age) are not statistically significant at the 5% level. Among them, s(chol) is borderline significant (p 0.08), while the others show weaker evidence of an effect.
The effective degrees of freedom for these terms are close to or slightly above 1, suggesting mostly linear or only weakly non-linear relationships.

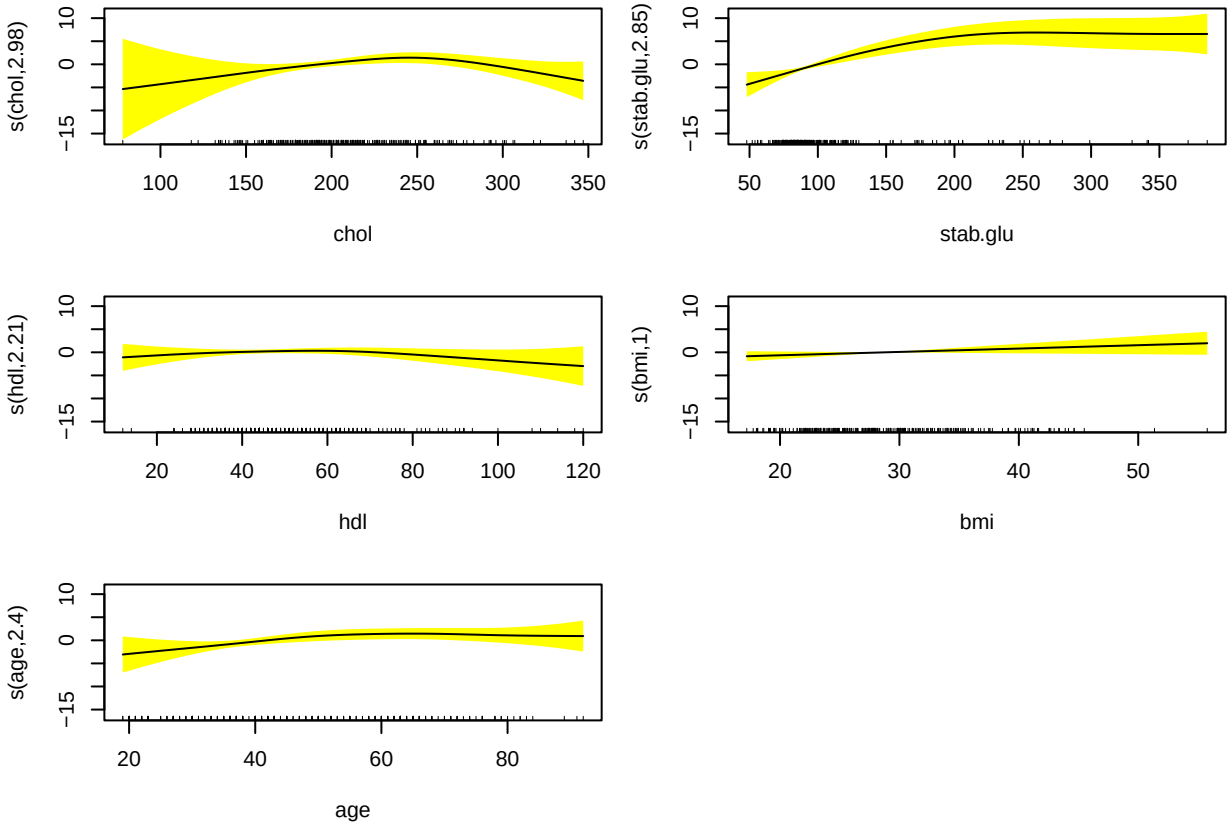
The parametric effect of gender is also not significant.
Overall, the model complexity is moderate, with only one predictor showing a clearly non-linear and important effect.

2(c) Smoothed effects plots and interpretation

```
par(mfrow = c(3, 2), mar = c(4, 4, 2, 1))

for (i in 1:5) {
  plot(gam_fit, select = i, shade = TRUE, shade.col = "yellow")
}

plot.new()
```




```
par(mfrow = c(1, 1))
```

Comment

The plots show the estimated smooth effects of the predictors on the log-odds of diabetes, together with confidence bands.

The smooth effect of stab.glu shows a clear and strong increasing pattern.
As stabilized glucose increases, the probability of diabetes rises sharply, especially at higher values. This confirms that glucose level is the most important predictor and that its effect is clearly non-linear.

For chol, the effect is weak and slightly curved, with wide confidence bands at the extremes. This indicates uncertainty and explains why this variable is only borderline significant in the model.

The smooth for hdl is mostly flat, suggesting no strong association with diabetes across its range. The effect of bmi shows a mild increasing trend, but the confidence bands overlap zero for most values, indicating a weak and uncertain effect.
For age, the smooth suggests a gradual increase in diabetes risk up to middle age, followed by a flattening at higher ages, but the overall effect remains modest.

Overall, the plots indicate that allowing for non-linearity mainly matters for stabilized glucose. For the other variables, the effects are small and uncertain, which is consistent with their lack of statistical significance in the model summary.

2(d) Confusion table, misclassification error, and comparison to (1)

```
# Predict probabilities on the test set
prob_test_gam <- predict(gam_fit, newdata = test, type = "response")

# Convert probabilities to class labels
pred_test_gam <- ifelse(prob_test_gam > 0.5, 1, 0)

# Confusion table
cm_gam <- table(
  Actual = test$y,
  Predicted = pred_test_gam
)
cm_gam
```

```
##        Predicted
## Actual  0  1
##      0 80  2
##      1  8  7
```

```
# Misclassification error
miscl_gam <- mean(pred_test_gam != test$y)
miscl_gam
```

```
## [1] 0.1030928
```

Comment

Using the GAM model, I predicted class membership for the test set and obtained the confusion table shown above.
The misclassification error of the GAM is about 10.3%, which is higher than the misclassification error of the logistic regression model from Exercise (1), which was about 8.2%.

This indicates that the GAM does not improve classification performance compared to the simpler logistic regression model.
Although the GAM captures non-linear effects, especially for stabilized glucose, this additional flexibility does not translate into better predictive accuracy on the test set.
As in Exercise (1), the model still misclassifies a noticeable fraction of diabetic cases.

Overall, in this application the GAM is mainly useful for understanding non-linear relationships rather than for improving classification accuracy.

2(e) Controlling smoothness with k

```
gam_fit_k <- gam(
  y ~ s(chol, k = 5) +
       s(stab.glu, k = 5) +
       s(hdl, k = 5) +
       s(bmi, k = 5) +
       s(age, k = 5) +
       gender,
  data = train,
  family = "binomial",
  method = "REML"
)

summary(gam_fit_k)
```

```
##
## Family: binomial
## Link function: logit
##
## Formula:
## y ~ s(chol, k = 5) + s(stab.glu, k = 5) + s(hdl, k = 5) + s(bmi,
##     k = 5) + s(age, k = 5) + gender
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -4.1048     0.7798  -5.264 1.41e-07 ***
## gendermale   -0.7790     0.7411  -1.051    0.293
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##               edf Ref.df Chi.sq p-value
## s(chol)     2.638  3.082  7.995  0.0505 .
## s(stab.glu) 2.661  3.121 36.238  <2e-16 ***
## s(hdl)      2.063  2.530  3.216  0.2917
## s(bmi)      1.000  1.000  2.564  0.1093
```

```
## s(age)      2.209  2.670  5.918  0.0849 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.672   Deviance explained = 69.3%
## -REML = 46.118  Scale est. = 1         n = 286
```

```
prob_test_gam_k <- predict(gam_fit_k, newdata = test, type = "response")
pred_test_gam_k <- ifelse(prob_test_gam_k > 0.5, 1, 0)

cm_gam_k <- table(
  Actual = test$y,
  Predicted = pred_test_gam_k
)
cm_gam_k
```

```
##       Predicted
## Actual  0  1
##      0 80  2
##      1  8  7
```

```
miscl_gam_k <- mean(pred_test_gam_k != test$y)
miscl_gam_k
```

```
## [1] 0.1030928
```

Comment

To control the complexity of the smooth terms, I set k = 5 for all smooth functions.
This choice allows the model to capture moderate non-linear patterns while avoiding overly flexible curves.
Given the relatively small sample size, a small value of k is sufficient and helps reduce the risk of overfitting, while still allowing meaningful non-linear effects to be detected.

After applying this restriction to GAM, stab.glu remains the only clearly significant smooth term ($p < 2e\text{-}16$) with an effective degrees of freedom of about 2.66, indicating a stable non-linear effect. The smooth terms for chol and age become borderline significant ($p = 0.05$ and $p = 0.085$), while hdl and bmi remain non-significant. The parametric effect of gender is still not significant.
The overall model fit remains almost unchanged, with about 69% deviance explained, very similar to the unrestricted GAM.

On the test set, the confusion table and the misclassification error (10.3%) are exactly the same as for the previous GAM and worse than for the logistic regression model from Exercise (1).
This shows that restricting the smoothness does not improve classification performance.

Overall, limiting the complexity confirms that the main conclusions are stable: stabilized glucose is the dominant predictor, and additional flexibility or restriction of the smooth terms does not lead to better predictive accuracy in this case.