# Project description for report 2

Objective: The objective of this second report is to apply the methods you have learned in the second section of the course on *"Supervised learning: Classification and regression"* in order to solve both a relevant classification and regression problem for your data.

Material: You can use the 02450Toolbox on Inside to see how the various methods learned in the course are used in Matlab, R or Python. In particular, you should review exercise 5 to 8 in order to see how the various tasks can be carried out.

---

## Mandatory section

In order to have your report evaluated, it must contain the following two items:

- The report must be submitted by a group of 2-3 students unless explicit permission has been given by the teachers to work alone. According to the DTU regulations, a student's contribution to the report must be clearly specified. Therefore, for each section, specify (in a table on the frontpage) who was responsible for it. **A report must contain this documentation to be accepted**[1]

- Solutions, or attempted solutions, for at least four of the exam problems found at the end of this document[2] . The solutions do not have to be long (a couple of lines, perhaps a calculation) but must show the gist of your reasoning so as to verify you have worked independently on the problem. We suggest they are given in an itemized format:

[1]For reports made by 3 students: Each section must have a student who is 40% or more responsible. For reports made by 2 students: Each section must have a student who is 60% or more responsible. For exam problems students are expected to contribute equally, and a student will not get any credit if they only contribute to the exam problems!.

[2]We ask you to do this because it has been our experience some students are unfamiliar with the written exam format until days before the exam, and we think this is the best way to ensure the requirements of the written exam are made clear early on. We don't evaluate your answers for correctness because that aspect of the course will be tested at the exam and would be redundant here.

1. Option $A/B/C/D$: To see this ...
2. Option $A/B/C/D$: We solve this by using..

*Don't know* is obviously not allowed, but you can take inspiration from the homework problems (and solutions given at the end of the notes). The purpose is to demonstrate that you have worked on the exam problems but not to test for correctness, and you can therefore hand in solutions which describes your best attempt at solving the problem (but you know are wrong). Keep in mind the solutions (fraction correct etc.) will not affect your evaluation, but rather whether the report is evaluated at all.

Your report cannot be evaluated unless it contains these items.

**Handin checklist**

- Make sure the mandatory section is included
- Make sure the report clearly display the **names *and* study numbers** of all group members. Make sure study numbers are correct.
- Your handin should consist of exactly two files: A `.pdf` file containing the report, and a `.zip` file containing the code you have used (extensions: `.py`, `.R` or `.m`; do **not** upload your data). The reports are not evaluated based on the quality of the code (comments, etc.), however we ask the code is included to avoid any potential issues of illegal collaboration between groups. Please do not compress or convert these files.
- Reports are evaluated based on how well they address the questions below. Therefore, to get the best evaluation, address all questions
- Use the group handin feature. **Do not upload separate reports for each team member as this will lead to duplicate work and unhappy instructors**
- **Deadline for handin is no later than 14 November at 13:00**. Late handins will not be accepted under normal circumstances

**Description**

Project report 2 should naturally follow project report 1 on *"Data: Feature extraction, and visualization"* and cover what you have learned in the lectures and exercises of week 5 to 8 on *"Supervised learning: Classification and regression"*. The report should therefore include two sections. A section on regression and a section on classification. The report will be evaluated based on how it addresses each of the questions asked below and an overall assessment of the report quality.

**Regression, part a:** In this section, you are to solve a relevant regression problem for your data and statistically evaluate the result. We will begin by examining the most elementary model, namely linear regression.

1. Explain what variable is predicted based on which other variables and what you hope to accomplish by the regression. Mention your feature transformation choices such as one-of-$K$ coding. Since we will use regularization momentarily, apply a feature transformation to your data matrix $\mathbf{X}$ such that each column has mean 0 and standard deviation 1[3].

2. Introduce a regularization parameter $\lambda$ as discussed in chapter 14 of the lecture notes, and estimate the generalization error for different values of $\lambda$. Specifically, choose a reasonable range of values of $\lambda$ (ideally one where the generalization error first drop and then increases), and for each value use $K = 10$ fold cross-validation (algorithm 5) to estimate the generalization error.

   Include a figure of the estimated generalization error as a function of $\lambda$ in the report and briefly discuss the result.

3. Explain how the output, $y$, of the linear model with the lowest generalization error (as determined in the previous question) is computed for a given input $\mathbf{x}$. What is the effect of an individual attribute in $\mathbf{x}$ on the output, $y$, of the linear model? Does the effect of individual attributes make sense based on your understanding of the problem?

**Regression, part b:** In this section, we will compare three models: the regularized linear regression model from the previous section, an artificial neural network (ANN)

[3]We treat feature transformations and linear regression in a very condensed manner in this course. Note for real-life applications, it may be a good idea to consider interaction terms and the last category in a one-of-$K$ coding is redundant (you can perhaps convince yourself why). We consider this out of the scope for this report

and a baseline. We are interested in two questions: Is one model better than the other? Is either model better than a trivial baseline?. We will attempt to answer these questions with two-level cross-validation.

1. Implement two-level cross-validation (see algorithm 6 of the lecture notes). We will use 2-level cross-validation to compare the models with $K_1 = K_2 = 10$ folds[4]. As a baseline model, we will apply a linear regression model with no features, i.e. it computes the mean of $y$ on the training data, and use this value to predict $y$ on the test data.

   Make sure you can fit an ANN model to the data. As complexity-controlling parameter for the ANN, we will use the number of hidden units[5] $h$. Based on a few test-runs, select a reasonable range of values for $h$ (which should include $h = 1$), and describe the range of values you will use for $h$ and $\lambda$.

2. Produce a table akin to Table 1 using two-level cross-validation (algorithm 6 in the lecture notes). The table shows, for each of the $K_1 = 10$ folds $i$, the optimal value of the number of hidden units and regularization strength ($h_i^*$ and $\lambda_i^*$ respectively) as found after each inner loop, as well as the estimated generalization errors $E_i^{\text{test}}$ by evaluating on $\mathcal{D}_i^{\text{test}}$. It also includes the baseline test error, also evaluated on $\mathcal{D}_i^{\text{test}}$. Importantly, you must re-use the train/test splits $\mathcal{D}_i^{\text{par}}, \mathcal{D}_i^{\text{test}}$ for all three methods to allow statistical comparison (see next section).

   Note the error measure we use is the squared loss *per observation*, i.e. we divide by the number of observation in the test dataset:

$$E = \frac{1}{N^{\text{test}}} \sum_{i=1}^{N^{\text{test}}} (y_i - \hat{y}_i)^2$$

   Include a table similar to Table 1 in your report and briefly discuss what it tells you at a glance. Do you find the same value of $\lambda^*$ as in the previous section?

3. Statistically evaluate if there is a significant performance difference between the fitted ANN, linear regression model and baseline using the methods described in chapter 11. These comparisons will be made pairwise (ANN vs. linear regression; ANN vs. baseline; linear regression vs. baseline). We will allow some freedom in what test to choose. Therefore, choose either:

[4] If this is too time-consuming, use $K_1 = K_2 = 5$

[5] Note there are many things we could potentially tweak or select, such as regularization. If you wish to select another parameter to tweak feel free to do so.

| Outer fold | ANN | | Linear regression | | baseline |
|---|---|---|---|---|---|
| $i$ | $h_i^*$ | $E_i^{\text{test}}$ | $\lambda_i^*$ | $E_i^{\text{test}}$ | $E_i^{\text{test}}$ |
| 1 | 3 | 10.8 | 0.01 | 12.8 | 15.3 |
| 2 | 4 | 10.1 | 0.01 | 12.4 | 15.1 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 10 | 3 | 10.9 | 0.05 | 12.1 | 15.9 |

Table 1: Two-level cross-validation table used to compare the three models

**setup I (section 11.3):** Use the paired $t$-test described in Box 11.3.4

**setup II (section 11.4):** Use the method described in Box 11.4.1)

Include $p$-values and confidence intervals for the three pairwise tests in your report and conclude on the results: Is one model better than the other? Are the two models better than the baseline? Are some of the models identical? What recommendations would you make based on what you've learned?

**Classification:** In this part of the report you are to solve a relevant classification problem for your data and statistically evaluate your result. The tasks will closely mirror what you just did in the last section. The three methods we will compare is a baseline, logistic regression, and **one** of the other four methods from below (referred to as *method 2*).

**Logistic regression** for classification. Once more, we can use a regularization parameter $\lambda \geq 0$ to control complexity

**ANN** Artificial neural networks for classification. Same complexity-controlling parameter as in the previous exercise

**CT** Classification trees. Same complexity-controlling parameter as for regression trees

**KNN** $k$-nearest neighbor classification, complexity controlling parameter $k = 1, 2 \ldots$

**NB** Naïve Bayes. As complexity-controlling parameter, we suggest the term $b \geq 0$ from section 11.2.1 of the lecture notes to estimate[6] $p(x = 1) = \frac{n^+ + b}{n^+ + n^- + 2b}$

| Outer fold | Method 2 | | Logistic regression | | baseline |
|---|---|---|---|---|---|
| $i$ | $x_i^*$ | $E_i^{\text{test}}$ | $\lambda_i^*$ | $E_i^{\text{test}}$ | $E_i^{\text{test}}$ |
| 1 | 3 | 10.8 | 0.01 | 12.8 | 15.3 |
| 2 | 4 | 10.1 | 0.01 | 12.4 | 15.1 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 10 | 3 | 10.9 | 0.05 | 12.1 | 15.9 |

Table 2: Two-level cross-validation table used to compare the three models in the classification problem.

1. Explain which classification problem you have chosen to solve. Is it a multi-class or binary classification problem?

2. We will compare logistic regression[7], *method 2* and a baseline. For logistic regression, we will once more use $\lambda$ as a complexity-controlling parameter, and for *method 2* a relevant complexity controlling parameter and range of values. We recommend this choice is made based on a trial run, which you do not need to report. Describe which parameter you have chosen and the possible values of the parameters you will examine.

   The baseline will be a model which compute the largest class on the training data, and predict everything in the test-data as belonging to that class (corresponding to the optimal prediction by a logistic regression model with a bias term and no features).

3. Again use two-level cross-validation to create a table similar to Table 2, but now comparing the logistic regression, *method 2*, and baseline. The table should once more include the selected parameters, and as an error measure we will use the error rate:

$$E = \frac{\{\text{Number of misclassified observations}\}}{N^{\text{test}}}$$

   Once more, make sure to re-use the outer validation splits to admit statistical evaluation. Briefly discuss the result.

[6] In Python, use the `alpha` parameter in `sklearn.naive_bayes` and in R, use the `laplacian` parameter to `naiveBayes`. We do *not* recommend NB for Matlab users, as the implementation is somewhat lacking.

[7] in case of a multi-class problem, substitute logistic regression for multinomial regression

4. Perform a statistical evaluation of your three models similar to the previous section. That is, compare the three models pairwise. We will once more allow some freedom in what test to choose. Therefore, choose either:

   **setup I (section 11.3):** Use McNemera's test described in Box 11.3.2)

   **setup II (section 11.4):** Use the method described in Box 11.4.1)

   Include $p$-values and confidence intervals for the three pairwise tests in your report and conclude on the results: Is one model better than the other? Are the two models better than the baseline? Are some of the models identical? What recommendations would you make based on what you've learned?

5. Train a logistic regression model using a suitable value of $\lambda$ (see previous exercise). Explain how the logistic regression model make a prediction. Are the same features deemed relevant as for the regression part of the report?

**Discussion:**

1. Include a discussion of what you have learned in the regression and classification part of the report.

2. If your data has been analyzed previously (which will be the case in nearly all instances), find a study which uses it for classification, regression or both. Discuss how your results relate to those obtained in the study. If your dataset has not been published before, or the articles are irrelevant/unobtainable, this question may be omitted but make sure you justify this is the case.

The report itself should be maximum 10 pages long including figures and tables and give a precise and coherent account of the results of the regression and classification methods applied to your data.

# Transferring/reusing reports from previous semesters

If you are retaking the course, you are allowed to reuse your previous report. You can either have the report transferred in it's entirety, or re-work sections of the report and have it evaluated anew.

To have a report transferred, *do absolutely nothing*. Reports from previous semesters are automatically transferred. Therefore, please do not upload old reports to Inside

as this will lead to duplicate work. As a safeguard, we will contact all students who are missing reports shortly after the exam.

If you wish to redo parts of a report you have already handed in as part of a group in a previous semester, then to avoid any issues about plagiarism please keep attribution to the original group members for those sections you choose not to redo.

# 1 Exam problems for the project

# Problems

**Question 1. Spring 2019 question 13:**

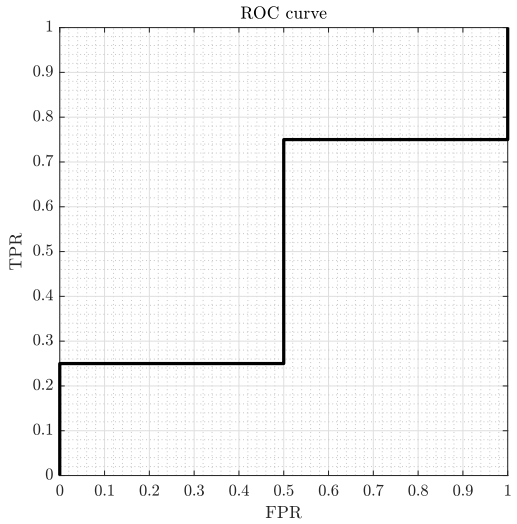


Figure 1: ROC curve for a neural network classifier, where the predictions and true class labels are one of the options in fig. 2.

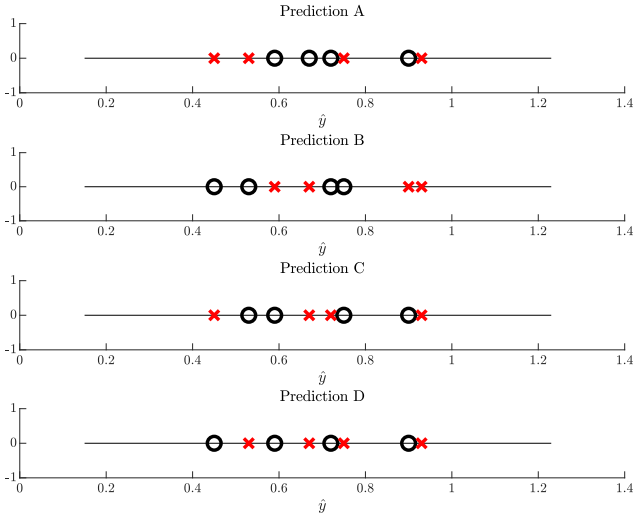


Figure 2: Four candidate predictions for the ROC curve in fig. 1. The observations are plotted horizontally, such that the position on the $x$-axis indicate the predicted value $\hat{y}_i$, and the marker/color indicate the class membership, such that the black circles indicate the observation belongs to class $y_i = 0$ and red crosses to $y_i = 1$.

A neural network classifier is trained to distinguish between two classes $y \in \{0, 1\}$ and produce class-probability $\hat{y}$ and the *reciever operator characteristic* (ROC) curve of the network when evaluated on a test set with $N = 8$ observations is shown in fig. 2. Suppose we plot the predictions on the $N = 8$ test observations by their $\hat{y}$ value along the $x$-axis and indicate the class labels by either a black circle (class $y = 0$) or red cross ($y = 1$), which one of the subplots in fig. 2 then corresponds to the ROC curve in fig. 1?

A Prediction A

B Prediction B

C Prediction C

D Prediction D

E Don't know.

**Question 2. Spring 2019 question 15:** Suppose we wish to build a classification tree based on Hunt's algorithm where the goal is to predict Congestion level which can belong to four classes, $y = 1$, $y = 2$, $y = 3$, $y = 4$. We consider binary splits based on the value of $x_7$, such that observations where $x_7 = z$ are assigned to the left branch and those where $x_7 \neq z$ are assigned the right branch. In table 3 we have indicated the number of observations in each of the four classes for the different values $x_7$ take in the dataset. Suppose we use the *classification error* impurity measure, which one of the following statements is true?

| | $x_7 = 0$ | $x_7 = 1$ | $x_7 = 2$ |
|---|---|---|---|
| $y = 1$ | 33 | 4 | 0 |
| $y = 2$ | 28 | 2 | 1 |
| $y = 3$ | 30 | 3 | 0 |
| $y = 4$ | 29 | 5 | 0 |

Table 3: Proposed split of the Urban Traffic dataset based on the attribute $x_7$. We consider a two-way split where for each interval we count how many observations belonging to that interval has the given class label.

A The impurity gain of the split $x_7 = 2$ is $\Delta \approx 0.0195$

B The impurity gain of the split $x_7 = 2$ is $\Delta \approx 0.0178$

C The impurity gain of the split $x_7 = 2$ is $\Delta \approx 0.0074$

D The impurity gain of the split $x_7 = 2$ is $\Delta \approx 0.0212$

E Don't know.

**Question 3. Spring 2019 question 18:** We will consider an artificial neural network (ANN) trained on the Urban Traffic dataset described in table 4 to predict the class label $y$ based on attributes $x_1, \ldots, x_7$. The neural network has a single hidden layer containing $n_h = 10$ units, and will use the softmax activation function (specifically, we will use the over-parameterized softmax function described in section 14.3.2 *(Neural networks for multi-class classification)* of the lecture notes) to predict the class label $y$ since it is a multi-class problem. For the hidden layer we will use a sigmoid non-linearity. How many parameters has to be trained to fit the neural network?

| No. | Attribute description | Abbrev. |
|---|---|---|
| $x_1$ | 30-minute interval (coded) | Time of day |
| $x_2$ | Number of broken trucks | Broken Truck |
| $x_3$ | Number of accident victims | Accident victim |
| $x_4$ | Number of immobile busses | Immobilized bus |
| $x_5$ | Number of trolleybus network defects | Defects |
| $x_6$ | Number of broken traffic lights | Traffic lights |
| $x_7$ | Number of run over accidents | Running over |
| $y$ | Level of congestion/slowdown (low to high) | Congestion level |

Table 4: Description of the features of the Urban Traffic dataset used in this exam. The dataset describes urban traffic behaviour of the city of Sao Paulo in Brazil. Each observation corresponds to a 30-minute interval between 7:00 and 20:30, indicated by the integer $x_1$, such that $x_1 = 1$ corresponds to 7:00-7:30 and so on up to $x_1 = 27$ that corresponds to 20:00-20:30. The other attributes $x_2, \ldots, x_7$ corresponds to a number of occurences of the given type in that 30-minute interval. We will consider the primary goal to be classification, namely to predict $y$ which is the level of congestion of the bus network in the given interval. The dataset used here consists of $N = 135$ observations and the attribute $y$ is discrete taking values $y = 1$ (corresponding to no congestion), $y = 2$ (corresponding to a light congestion), $y = 3$ (corresponding to an intermediate congestion), and $y = 4$ (corresponding to a heavy congestion).

A Network contains 124 parameters

B Network contains 280 parameters

C Network contains 110 parameters

D Network contains 88 parameters

E Don't know.

**Question 4. Spring 2019 question 20:**

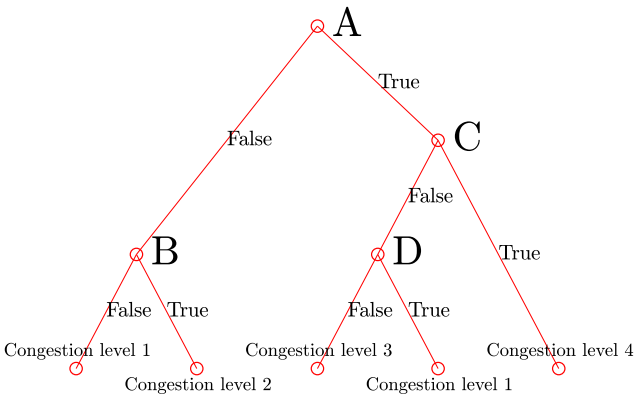


Figure 3: Structure of decision tree. The goal is to determine the splitting rules.

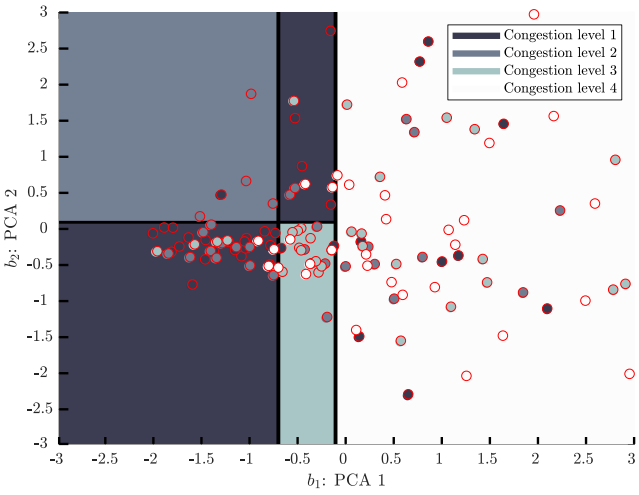


Figure 4: Classification boundary.

We will consider the Urban Traffic dataset projected onto the first two principal directions. Suppose we train a decision tree to predict which of the four classes an observation belongs to. Since the attributes are continuous, we will consider binary splits of the form $b_i \geq z$ for different values of $i$ and $z$, where $b_1$, $b_2$ refer to the coordinates of the observations when projected onto principal directions. Suppose the trained decision tree has the form shown in fig. 3, and that according to the tree the predicted label assignment for the $N = 135$ observations are as given in fig. 4, what is then the correct rule assignment to the nodes in the decision tree?

A ***A***: $b_1 \geq -0.16$, ***B***: $b_2 \geq 0.03$, ***C***: $b_2 \geq 0.01$, ***D***: $b_1 \geq -0.76$

B ***A***: $b_1 \geq -0.76$, ***B***: $b_1 \geq -0.16$, ***C***: $b_2 \geq 0.03$, ***D***: $b_2 \geq 0.01$

C ***A***: $b_2 \geq 0.03$, ***B***: $b_1 \geq -0.76$, ***C***: $b_2 \geq 0.01$, ***D***: $b_1 \geq -0.16$

D ***A***: $b_1 \geq -0.76$, ***B***: $b_2 \geq 0.03$, ***C***: $b_1 \geq -0.16$, ***D***: $b_2 \geq 0.01$

E Don't know.

**Question 5. Spring 2019 question 22:**

| | ANN | | Log.reg. | |
|---|---|---|---|---|
| | $n_h^*$ | $E_1^{\text{test}}$ | $\lambda^*$ | $E_2^{\text{test}}$ |
| Outer fold 1 | 1 | 0.385 | 0.01 | 0.615 |
| Outer fold 2 | 1 | 0.357 | 0.01 | 0.286 |
| Outer fold 3 | 1 | 0.429 | 0.01 | 0.357 |
| Outer fold 4 | 1 | 0.571 | 0.06 | 0.714 |
| Outer fold 5 | 1 | 0.538 | 0.32 | 0.538 |

Table 5: Result of applying two-level cross-validation to a neural network model and a logistic regression model. The table contains the optimally selected parameters from each outer fold ($n_h^*$, hidden units and $\lambda^*$, regularization strength) and the corresponding test errors $E_1^{\text{test}}$ and $E_2^{\text{test}}$ when the models are evaluated on the current outer split.

Suppose we wish to compare a neural network model and a regularized logistic regression model on the Urban Traffic dataset. For the neural network, we wish to find the optimal number of hidden neurons $n_h$, and for the regression model the optimal value of $\lambda$. We therefore opt for a two-level cross-validation approach where for each outer fold, we determine the optimal number of hidden units (or regularization strength) using an inner cross-validation loop with $K_2 = 4$ folds. The tested values are:

$$\lambda : \{0.01, 0.06, 0.32, 1.78, 10\}$$
$$n_h : \{1, 2, 3, 4, 5\}.$$

Then, given this optimal number of hidden units $n_h^*$ or regularization strength $\lambda^*$, the model is trained and evaluated on the current outer split. This produces table 5 which shows the optimal number of hidden units/lambda as well as the (outer) test classification errors $E_1^{\text{test}}$ (neural network model) and $E_2^{\text{test}}$ (logistic regression model). Note these errors are averaged over the number of observations in the the (outer) test splits. Suppose the time taken to train/test a single neural network model in milliseconds is

training time: 20 and testing time: 5

and the time taken to train/test a single logistic regression model is

training time: 8 and testing time: 1,

what is approximately the time taken to compose the table?

A 6800.0 ms

B 13600.0 ms

C 3570.0 ms

D 13940.0 ms

E Don't know.

**Question 6. Spring 2019 question 26:** Consider again the Urban Traffic dataset. We consider a multinomial regression model applied to the dataset projected onto the first two principal directions, giving the two coordinates $b_1$ and $b_2$ for each observation. Multinomial regression then computes the per-class probability by first computing the 3 numbers:

$$\hat{y}_k = \begin{bmatrix} 1 \\ b_1 \\ b_2 \end{bmatrix}^\top \boldsymbol{w}_k, \text{ for } k = 1, \ldots, 3$$

and then subsequently use the softmax transformation in the form:

$$P(y = k|\hat{\boldsymbol{y}}) = \begin{cases} \frac{e^{\hat{y}_k}}{1+\sum_{k'=1}^{3} e^{\hat{y}_{k'}}}. & \text{if } k \leq 3 \\ \frac{1}{1+\sum_{k'=1}^{3} e^{\hat{y}_{k'}}} & \text{if } k = 4 \end{cases}$$

to compute the per-class probabilities. Suppose the weights are given as:

$$\boldsymbol{w}_1 = \begin{bmatrix} 1.2 \\ -2.1 \\ 3.2 \end{bmatrix}, \boldsymbol{w}_2 = \begin{bmatrix} 1.2 \\ -1.7 \\ 2.9 \end{bmatrix}, \boldsymbol{w}_3 = \begin{bmatrix} 1.3 \\ -1.1 \\ 2.2 \end{bmatrix}.$$

Which of the following observations will be assigned to class $y = 4$?

A Observation $\boldsymbol{b} = \begin{bmatrix} -1.4 \\ 2.6 \end{bmatrix}$

B Observation $\boldsymbol{b} = \begin{bmatrix} -0.6 \\ -1.6 \end{bmatrix}$

C Observation $\boldsymbol{b} = \begin{bmatrix} 2.1 \\ 5.0 \end{bmatrix}$

D Observation $\boldsymbol{b} = \begin{bmatrix} 0.7 \\ 3.8 \end{bmatrix}$

E Don't know.