Technical University of Denmark

02450 Introduction to Machine Learning and Data Mining

Supervised learning: Classification and Regression

Authors:	***Student numbers:***
Filippo Bentivoglio	s210299
Jonas Løvenhardt Henriksen	s195457

Contents

Preface

This report is the second of two reports in a study, conducted during the Autumn of 2021 as part of the course "02450 - Introduction to Machine Learning & Data Mining" offered at the Technical University of Denmark (DTU).

The objective of this report is to apply methods learned during the course on a data set, starting from the learning acquired in the first report.

The students did not have contributed equally to the project.

Section	**Filippo s210299**	**Jonas s195457**
1. Regression, part a	90%	10%
2. Regression, part b	90%	10%
3. Classification	10%	90%
4. Discussion	90%	10%
5. Exercise	1-2	3-4

Table 1: Students contribution

1 Regression, part a

The aim of this part of the report is to evaluate the results of a regularized Linear Regression model, in which a variable will be predicted based on other variables from the Forest Fires data set [1]. Additional information and analysis on the data set can be found in the previous report [2].

1.1 Introduction to regression

As described in the previous report, the variable to predict is the *Dimension of burned area*.

The aim of this section, is to train a model capable of predicting the *Area* from the other features included in the data set with a small validation error when evaluating the model.

The feature transformation steps performed is the *1-out-K-encoding* applied to the `Month`, `Day`, X and Y, implying an addition of 37 attributes into the data set (12 for *Month*, 7 for *Day*, 9 for X and 9 for Y. However, looking deeply into detail, there are a few instances where a feature is never observed. For example, there is no record of a fire at location $y = 1$ in the data set. This cause some issues in the *rlr validate* function provided, probably because some arithmetic operations are executed which result in a division by 0, since for the new feature encoded ($y = 1$), all the observation have the same value. The group, apart from the classic *1-out-K-encoding* with the values 0 and 1, has also tried another version using 1 and 2 respectively but without any results. Additionally, comparing the correlation matrix obtained in the previous report [2] with the one obtained after having applied *1-out-K-encoding* (see section 5.1) and referring to the related paper [1] where the authors did not take those features into consideration, the group have decided to discard these new attributes for the task accomplished in this report.

The second transformation, concerning the output variable *Area*, since, in the original data set, 247 out of 517 values are 0, to reduce skewness and improve symmetry, the logarithm function $y = ln(x + 1)$ was applied to the *Area* attribute. The result is that the mean becomes equal to 1.11 m^2 and the maximum value to 6.99 m^2.

Finally, the data set features have been standardized, subtracting the mean and dividing by the standard deviation.

1.2 Regularized regression

A regularization parameter λ was introduced to the model, for which the initial range of values were chosen to be in the range from 10^{-2} to 10^{19} included. For each value of λ a K = 10 fold cross-validation was performed to estimate the validation error.

The aim of introducing a regularization parameter is to reduce the variance of the model without introducing too much bias. Inspecting Figure 1 pane b, it can be noticed that the optimal *validation error* is obtained at $\lambda = 1000$. This means that choosing $\lambda = 1000$, the best balance between variance and bias is adopted. For λ values smaller than 1000, the model vary too much (high variance), indeed, the train error is extremely low but not the generalization error for small values of λ, implying that when it comes to a new data set, the results are really poor. Instead, for values bigger than $\lambda = 1000$, the train error increases as well as the validation error: this indicates that there is too much bias imposed, and the model is stiff, implying does not vary much.

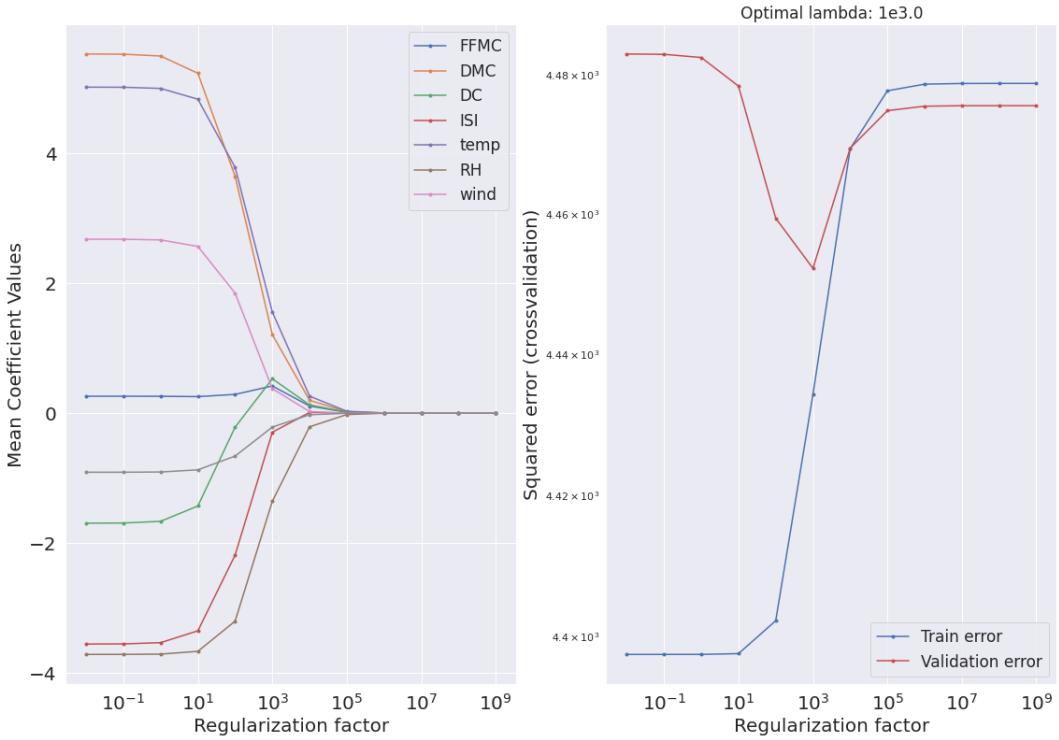

Figure 1: (a) Optimal weights of attributes as a function of λ; (b) validation and training error as a function of λ.

Now referring to pane a in Figure 1, where the weights ω^* are plotted as a function of the regularization parameter λ, notice that for $\lambda = 1000$, the magnitude of the weights has been greatly reduced. This choice, most likely, introduces a small bias into the model at the cost of reducing the variance.

To notice that the values of λ has been chosen in the respective range with a order of magnitude of 10. This imply that the optimal value, hence the lowest generalization error, might not be $\lambda = 1000$. This issue will be addressed in subsection 2.1.

1.3 New observation

The regularized Linear Regression model was trained by minimizing the sum-of-squares error term (course book [3], section 14.1):

$$E_\lambda(\boldsymbol{\omega}, \omega_0) = ||\boldsymbol{y} - \omega_0 \mathbf{1} - \hat{\mathbf{X}}\boldsymbol{\omega}||^2 + \lambda||\boldsymbol{\omega}||^2$$

In the case $\boldsymbol{X}$ and $\boldsymbol{y}$ are scalars,

$$\omega^* = \frac{Xy}{X^2 + \lambda}$$

For any new observation, the prediction of the burned *Area* size will be predicted according to the weights shown in Figure 2. Looking at the plot, it is clear that the features *temp, RH* and *DMC* contributes the most when it comes to a new prediction. Particularly, *temp* and *DMC* have positive weights, meaning that a new observation with large positive values for those two features will most likely be predicted as a big fire (large burned area). While it will be the opposite for the *RH* attribute, showing a negative coefficient.
Furthermore, comparing these weights with the new correlation matrix (Figure 5), it can be noticed that the weights match fairly well the matrix coefficients both in term of magnitude and sign (except

for the *ISI* attribute which appears to have opposite sign). Hence, the result are in accordance with prior learnings and therefore make sense.

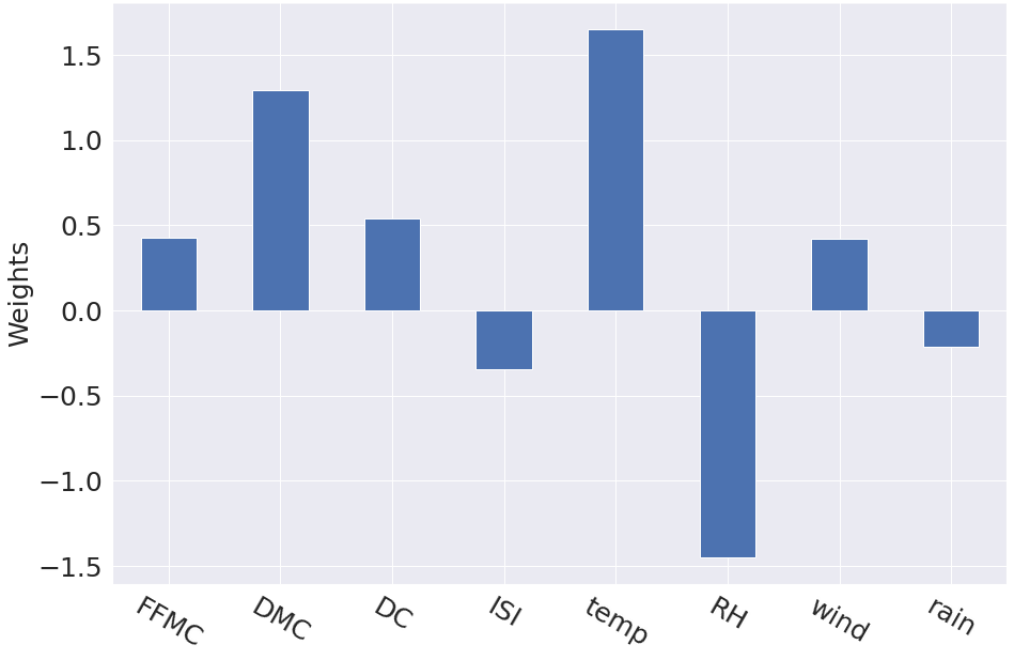

Figure 2: Weights of attributes for the optimal regularized Linear Regression model.

Finally, to get an estimate of the prediction capability, the regularized *Linear Regression* model has been ran 10 times, with parameter $\lambda = 1000$, with randomized test sets. Looking at the results in 2, where regarding the prediction $\hat{y}$ the range of the respective statistics is shown, it is clear that overall the average of the new predictions is quite close to the target value of 11.1. On the other hand, it is also evident, that the model fails when it comes to predict big burned area, since the maximum observation achieved is 1.59, by far close to the maximum in the data set.

	mean $[m^2]$	*min* $[m^2]$	*max* $[m^2]$
y	1.11	0.00	6.99
$\hat{y}$	$[1.05, 1.16]$	$[0.29, 0.85]$	$[1.31, 1.59]$

Table 2: New prediction analysis

2 Regression, part b

The aim of this part of the report is to compare the Linear Regression model from section 1 with an artificial neural network (ANN) model and a simple baseline.

2.1 Two-level cross validation

First, a number of test runs were performed in order to determine a a reasonable range of hidden units in the hidden layer of the *Artificial Neural Network*. In this first step, a two-level cross validation with $K_1 = K_2 = 5$ folds were adopted. These runs showed that the best number of hidden values which respectively gives the lowest generalization error is 1 for each one of the 5-outer loop. The reason for this is most likely due to the small data set dimension, since it only contains 517 observations. Therefore, 1 hidden unit results to be enough to get the complexity of the data, instead, adding more hidden units brings the model to over-fit the training data causing a worst generalization error.

In Table 3 are illustrated the results of the application of 2-fold cross validation for each model. It includes the test errors E_i^{test} of each outer fold for each models, the respective number of hidden units h adopted for each fold of the ANN model and the regularization parameter λ for each fold of the regression model. The test error is computed adopting the Meas Squared Error (MSE) in all cases.

Outer fold	ANN		Linear Regression		Baseline
	h_i^*	E_i^{est}	λ_i^*	E_i^{est}	E_i^{est}
1	1	1.632	164	1.932	1.938
2	1	1.877	164	2.167	2.104
3	1	1.559	164	1.964	1.976
4	1	1.740	164	1.904	1.926
5	1	1.447	164	1.771	1.740
6	1	1.942	164	2.179	2.225
7	1	1.578	164	2.003	1.748
8	1	1.211	167	1.697	2.034
9	1	1.696	167	2.061	1.832
10	1	1.597	167	2.379	1.953
Average		**1.628**		**2.005**	**1.953**

Table 3: Regression summary

Inspecting at the table, it is clear that the ANN model produced the best performance between the models considered. The linear regression, on the other hand, did not outperform the baseline. The authors of the related paper [1] emphasized the complexity of the regression task, hence, in some ways, it was expected that the results would be unsatisfactory. The combination of the intrinsic complexity of the task (prediction of burned area size) and the simplicity of the *Linear Regression* is most likely to blame for the bad results.

It is worth to mention, that in subsection 1.2 the best choice for λ^*, given the range of values in orders of magnitude of ten, was 1000. However, in this section, the range was iteratively narrowed down to provide a more precise estimation of λ while keeping the test error to the minimum. This process led to focus on values between 164 and 168.

2.2 Regression statistical tests

A statistical test were performed, comparing the three models (*Linear Regression, ANN* and *Baseline*), following the *paired t-test* approach, explained in box 11.3.4 [1]. The results are showed in **??** which includes the the mean of the difference between the generalization error of the respective paired models, the confidence intervals and the P-values, along with a conclusion according to the null-hypotheses for each comparison.

Pairwise Test	$\hat{z}^{[1]}$	*P-value*	CI_{lower}	CI_{upper}	*Conclusion*
$E^{test}_{Baseline} - E^{test}_{Linear}$	$4.44 \cdot 10^{-3}$	0.820	-0.043	0.034	H_0 failed to be rejected
$E^{test}_{Baseline} - E^{test}_{ANN}$	$2.00 \cdot 10^{-2}$	0.386	-0.065	0.025	H_0 failed to be rejected
$E^{test}_{ANN} - E^{test}_{Linear}$	$-1.55 \cdot 10{-}2$	0.285	-0.013	0.044	H_0 failed to be rejected

[1] Where $\hat{z} \approx (\frac{1}{n}\sum_{i=1}^{n} z_i^A - \frac{1}{n}\sum_{i=1}^{n} z_i^B) = \frac{1}{n}\sum_{i=1}^{n} z_i$ represents the estimated difference in generalization error between the pairwise models.

Table 4: Regression performance evaluation

Looking at the table above, specifically at the *p-values*, it is possible to conclude that the 3 models are not significantly different between each other, evaluating with a 95% confidence interval. In particular, the *Linear Regression* and the *Baseline* appear to be really similar, showing a *p-value* of 0.820. These two conclusions might have also been deducted from Table 3, where the three generalization errors show small differences between each other, especially the *linear regression* and the *baseline* ones. In accordance with this analysis, since the α value corresponds to 0.05, meaning that the confidence interval is 95%, all the 3 *null-hypothesis* failed to be rejected.

However, it should be noted, that among the models tested for this data set, the *artificial neural network* appears to provide the best results. Hence, for further and more complex studies on this data set, which might include features transformation, feature selection or more advanced technique, should probably be the recommended for the prediction of the burned area.

It is worth to remember, that the statistical test of performance conducted in this section, falls into the *Setup I*, implying that all the consideration made are valid exclusively for the considered data set. Additionally, the analysis has been carried out using the most frequent occurring regularization parameters from the 2-level cross-validation in subsection 2.1 (i.e. the most common λ and h) and using the same data set splits adopted in the respective outer folder.

3 Classification, part c

The aim of this part of the report is to compare a Logistic regression model with a k-nearest neighbor (KNN) model and a simple baseline. The problem to solve was a binary classification problem, where the class represented whether the burned area was 0 or larger than 0.

3.1 Two-level Cross Validation

For this part a number of trial runs were performed in order to determine a range for the hyper-parameters of both the Logistic Regression model and the *KNN* model. These runs showed that for the logistic regression model a range of lambda from 0.1 - 1000 gave the best results. Regarding the *KNN* model, the best performances were obtained when 20 was selected as maximum number of neighbours. The KNN's parameter k describes how many nearest points should be evaluated to predict to which class a data point belongs to. The distance between two samples is measured with Euclidean distance. Finally, the Baseline model will compute the largest class on the training data and predict everything in the test data as belonging to that class. In the case of a binary classification, the data set result quite balanced since it present 247 samples as 0 and 280 as 1.

In Table 5 are illustrated the results of the application of 2-level cross validation for each model. The table includes the complexity controlling parameter for the *KNN* model and the Logistic regression model and also the test error for each model in the outer fold.

Outer fold	KNN		Logistic Regression		Baseline
	k_i^*	E_i^{est}	λ_i^*	E_i^{est}	E_i^{est}
1	2	50.0	21.2	42.30	48.07
2	10	53.84	10.0	46.15	50.00
3	5	46.15	138.9	46.15	48.07
4	7	48.07	14.5	44.23	42.30
5	13	48.07	167	50.00	61.53
6	9	48.07	268.2	38.46	32.69
7	8	48.07	21.2	46.15	53.84
8	1	47.05	184.2	43.13	33.33
9	2	47.05	202.3	49.01	45.09
10	3	56.86	30.8	58.82	62.74
Average		**49.32**		**46.42**	**47.77**

Table 5: Two-level cross-validation to compare the classification models

When inspecting the table it can be seen that the complexity controlling parameters have high variance. This variance can be explained by the data-set being small, only 517 observations. This effectively means that having 10 outer folds splits the data into partitions of roughly 50, thus the inner folds will run on very small data partitions. Further inspecting the table we see that the *KNN* model may have worse performance than the baseline. The logistic regression model does seem to have slightly better performance than both of the other models.

3.2 Statistical Evaluation

A statistical test was performed to compare the three models (Logistic Regression, KNN and Baseline), following the Mcnemera's test, from box 11.3.2 [1]. The results are shown in Table 6. The table

includes the mean of the difference between the classification error of the respective paired models, the confidence intervals and the P-values, along with a conclusion according to the null-hypothesis for each comparison.

Pairwise Test	$\hat{\theta}$[1]	*P-value*	CI_{lower}	CI_{upper}	*Conclusion*
$E^{test}_{Baseline} - E^{test}_{KNN}$	$1.55 \cdot 10^{-2}$	0.665	-0.046	0.077	H_0 failed to be rejected
$E^{test}_{Baseline} - E^{test}_{logistic}$	$-1.35 \cdot 10^{-2}$	0.603	-0.057	0.030	H_0 failed to be rejected
$E^{test}_{KNN} - E^{test}_{Logistic}$	$-2.90 \cdot 10-2$	0.365	-0.087	0.030	H_0 failed to be rejected

[1] Where $\hat{\theta} = \frac{n_{12}-n_{21}}{n}$ represents the estimated difference in classification error between the pairwise models.

Table 6: Performance evaluation classification

Looking at the table it may be noticed that the p-values are all above 0.05. We may therefore conclude that there is not strong evidence to reject any of the *null-hypothesis*. We use the p-value to determine effect since the data-set is fairly small, 517 observations. We particularly see that the p-value for the tests against the baseline are high. These high p-values would suggest that the performance of the two models are not significantly different from the baseline. This can be backed when seeing the average classification error of the models from Table 5. Another observation is that the p-value for the test between the Logistic regression model and the *KNN* model shows a p-value of 0.365. This being the lowest p-value indicates these should be the most different. This result is to be expected since the *KNN* model and the *Logistic Regression* model have the biggest difference in classification error in Table 5.

Based on the results of these test it may be noted that a *Logistic regression* model may provide the best classification performance. Further studies may want to work with feature transformations to increase performance. Alternatively other models may be introduced, this based on the fact than none of the models in this section outperformed the baseline.

3.3 Logistic Regression Model Interpretation

The regularized Logistic Regression model was trained by minimizing the classification error. For any new observation the classification prediction will be based on the weights shown in Figure 3. Looking at the plot it is clear that the *DC* and *Wind* featurs contribute most towards the prediction. After these features the *FFMC* and *Temp* features contribute most while the remaining features contribute slightly. The two most significant features DC and *Wind* have positive weights, meaning that a new observation with large positive values for these features will tend to be classified with class 1 (Burn area larger than 0).

When comparing the weights from the linear regression part Figure 2 we see some clear differences. For the linear regression the *Temp* and *DMC* are significantly more important. We also see that the weight of the *DMC* feature hase changed to negative for the classification problem.

Figure 3: Weights of features for the optimal regularized Logistic Regression model.

4 Discussion

4.1 Learnings

- In Regression, part a, the *Linear Regression* model was shown to predict averages of *Area* close to the known actual average but with poor performance when predicting large fires.
- In Regression, part b, it was found that overall the result were not good. However, of the three models (*Baseline, ANN & Linear Regression*), the ANN model shown the lowest test error but the *Linear Regression* did not perform better than the *Baseline*.
- In Classification, part c, it was found that none of the two models performed better than the baseline with a 95% certainty margin. The *Logistic Regression* model did however show the lowest classification error in the two-level cross validation.

4.2 Related work

To apply perspective to some of the findings from this study, the group has looked into present literature to find similar studies, which were based on the data set used.

Paulo Cortez and Anìbal Morais, presented their work in 2007 [1] where they explored 5 different Data Mining approach to predict the burned area of forest fires and four distinct feature selection setups to infer about the impact of the input variables. It should be taken into account that, in the above cited work, the authors have applied to the outputs the of the different models the inverse of the logarithm function, hence, the errors are on a different scale with respect to the choice made by the group in this report. Furthermore, the error metrics are different, so the result are not directly comparable. However, the main achievements will be listed below.

Multiple Regression (MR), Decision Tree (DT), Random Forest (RF), Neural Networks (NN) and Support Vector Machines (SVM) are the 5 models considered by Cortez et. al.. Instead, the features selection setups are: spatial (S), temporal (T), the FWI system features (FWI) and meteorological (M) attributes. Hence, respectively: (X,Y), (Month, Day), (FFMC,DMC,DC,ISI) and (temp,RH,wind,rain). Finally, the overall performance is computed by the Mean Absolute Deviation (MAD) and Root Mean Squared (RMSE).

The authors carried out a 10 fold cross validation tests for all the models, with an additional internal 10-fold grid search for the NN and SVM to find the best hyper-parameters. Their work showed that the SVM, overall, tends to produce the best predictions. The best performances are obtained adopting the SVM models with only the meteorological M features, giving a MAD error of 12.71. Another interesting result is the non relevance of the spatial (S) and temporal (T) variables, since when removed the SVM performance improved.

Additionally, Cortez et. al. performed the REC analysis to observe hot the test errors are distributed along the output range. The results, as also pointed out in this report, showed that the performance of the SVM model improved when predicting small fires, within the range $[0, 3.2]ha$.

5 Appendix

5.1 Correlation Matrix

In this section, the group want to show the differences between the correlation coefficients of the data set when the nominal attributes were simply transformed into index values and when *1-out-K-encoding* was applied.

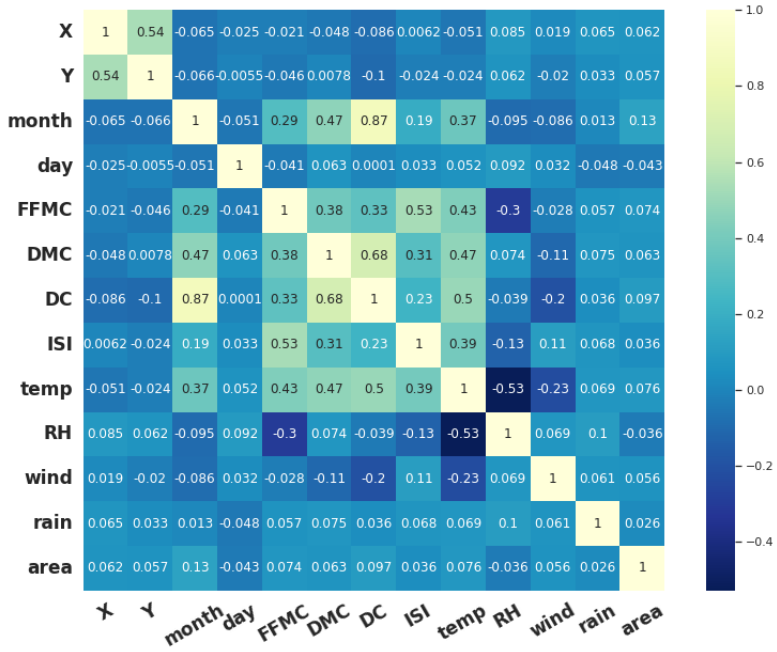

Figure 4: Correlation matrix form Report 1.

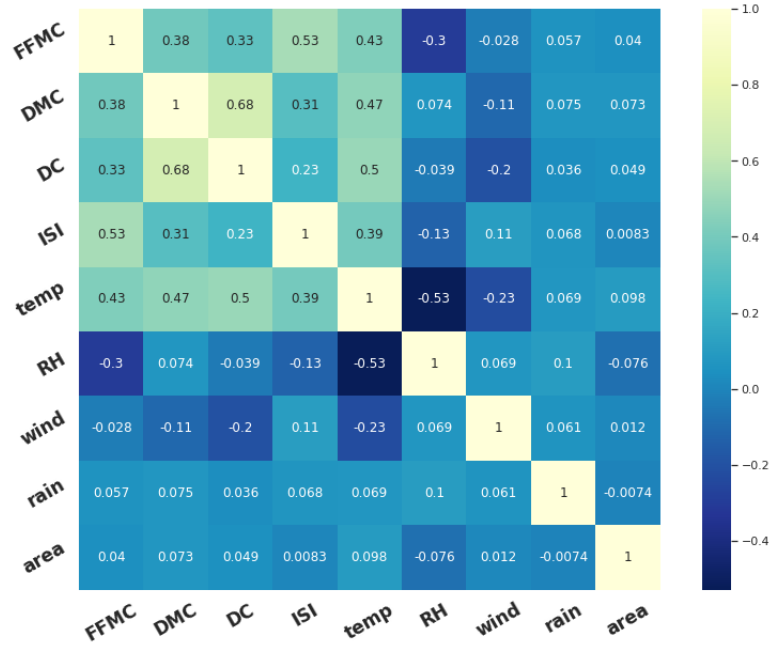

Figure 5: New correlation matrix.

In Figure 5, since the *1-out-K-encoding* introduced 37 new features, only the relevant features are displayed.

It is possible to notice, that the relevant attribute coefficients do not vary much between the two figures in term of magnitude and there is only one discrepancy between the signs (*rain*).

5.2 Question 1

Answer: C.

Start by observing that all four predictions have four positive observations (red crosses) and four negative observations (black circles). Thus, movement from point to point on the ROC curve will happen in steps of 1/4, either on the FPR-axis or the TPR-axis. If we start with a treshhold value ¿ 1, FPR = 0 and TPR = 0. As the threshold is lowered, one first observes a postive observation giving FPR = 0, TPR = 1/4, then a negative observation giving FPR = 1/4, TPR = 1/4, then another negative one with FPR = 1/2, TPR = 1/4. This reasoning leds to C being the only valid option.

5.3 Question 2

Answer: C.

In total there are N = 135 observations, distributed among the four congestion levels as:

$$N(y=1) = 33+4+0 = 37 \qquad N(y=2) = 28+2+1 = 31$$

$$N(y=3) = 30+3+0 = 33 \qquad N(y=4) = 29+5+0 = 34$$

We can calculate the parent impurity measure as:

$$I(v) = 1 - max(\frac{37}{135}, \frac{31}{135}, \frac{33}{135}, \frac{34}{135}) = \frac{98}{135}$$

For the split of $x_7 = 2$ the number of observation and the impurity measures are:

$$N(v_1) = 1 \qquad I(v_1) = 1 - max(\frac{0}{1}, \frac{1}{1}, \frac{0}{1}, \frac{0}{1}) = 1$$

$$N(v_2) = 134 \qquad I(v_2) = 1 - max(\frac{37}{134}, \frac{30}{134}, \frac{33}{134}, \frac{34}{134}) = \frac{97}{134}$$

To conclude, the impurity gain is:

$$\Delta = \frac{98}{135} - (\frac{1}{135} \cdot 0 + \frac{134}{135} \cdot \frac{97}{134}) = \frac{1}{135} \approx 0.0074$$

5.4 Question 3

Answer: A.

There is only one single hidden layer, so the following sizes have to be considered:

Input size: $i = 7$ Hidden layer size: $h = 10$ Output size: $\sigma = 4$

The number of parameters is determined by the connections between layers and the biases in every layer:

$$(i * h + h * \sigma) + (h + \sigma) \ = (7 * 10 + 10 * 4) + (10 + 4) = 124$$

5.5 Question 4

Answer D.

The class *Congestion Level 4* can be seen from *Figure 4.* to label all data points with a $b1$ value bigger than ≈ -0.16. When looking at the decision tree *Figure 3* We see that only one branch reaches *Congestion Level 4*, at split *C*. When evaluating the options only one has C implementing this border, namely D.

References

[1] P. Cortez and A. Morais. A Data Mining Approach to Predict Forest Fires using Meteorological Data. In J. Neves, M. F. Santos and J. Machado Eds., New Trends in Artificial Intelligence, Proceedings of the 13th EPIA 2007 - Portuguese Conference on Artificial Intelligence, December, Guimarães, Portugal, pp. 512-523, 2007. APPIA, ISBN-13 978-989-95618-0-9

[2] Filippo Bentivoglio et al. *Data: Feature extraction and visualization - Group 177 - Report 1* 2021. `https://dtudk-my.sharepoint.com/:b:/r/personal/s210299_dtu_dk/Documents/Feature%20extraction%20and%20visualization.pdf?csf=1&web=1&e=2HQCTY`

[3] Tue Herlau et al. *Introduction to Machine Learning and Data Mining.* 2021.