Introduction to Data Science

Course Project

Report Document

**M Abdullah Awan**

**Instructions: Read These Carefully Before Starting**

1. Due Date: Sunday 4th December 2022 – 11:59PM
2. Submission will be taken on Google Classroom
3. Submit only the following 2 files named like the following:
   1. Code File (Jupyter Notebook): L210000\_Code.ipynb
   2. Report Document (This File): L210000\_Report.pdf
4. Project will not be evaluated if:
   1. You submit python (.py) files
   2. You submit multiple .ipynb files
   3. You submit compressed (.rar or .zip) files
   4. You submit any files other than the required PDF and IPYNB
5. Upload data files directly to Google Colab - do not use Google Drive or GitHub linking method
6. All source files needed to complete this project are uploaded with it on Google Classroom.
7. Do not add the data file with your submission on Google Classroom.

Not following these instructions will lead to mark deduction.

**Please try to use Microsoft Word instead of Google Docs to edit this document and to export it as a PDF file for final submission.**

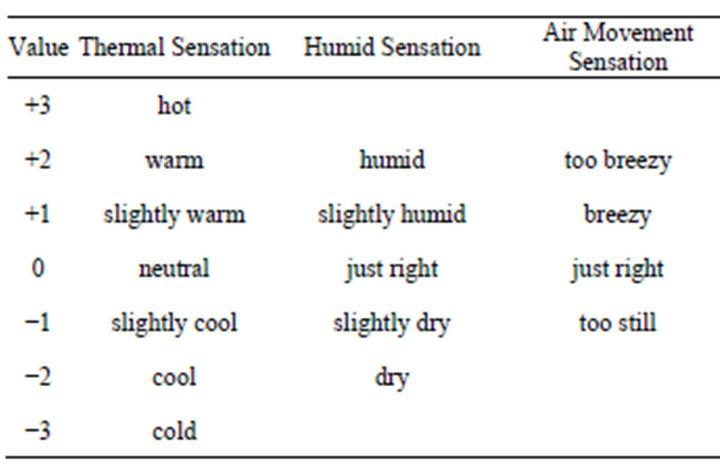
Happy Coding 😺

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*TA Emails*

For this project you will be applying machine learning models (both regression and classification) to the dataset which contains information about various individuals, their clothing, and its properties along with other atmospheric elements such as temperature, pressure humidity etc. The users also provided feedback on if they feel cold or not. The feedback (through AMV and PMV) which is based on the following mapping:

The following table shows the mapping of sensations:



**The dataset is given in an excel file named CollectedData.xlsx, see sheet 2 of excel file.** The dimension names (column headers) are not mentioned in the given file. The table below describes the columns which will be of your interest.

|  |  |  |
| --- | --- | --- |
| **Column number** | **Feature Name** | **Feature Description** |
| 3 | Age | Age |
| 22 | Clo | Clothing insulation |
| 19 | Met | Met Rate |
| 26 | Dewpt | Dewpt |
| 27 | PlaneRadTemp | plane radiant temperature |
| 37 | Ta | Average air temperature |
| 38 | Tmrt | Average mean radiant temperature |
| 40 | Vel | Air Velocity |
| 42 | AirTurb | Air Turbulance |
| 43 | Pa | Vapor Pressure |
| 44 | Rh | Humidity |
| 74 | TaOutdoor | Outdoor Air Temperature |
| 77 | RhOutdoor | Outdoor Humidity |
| 8 | AMV | Classification response variable |
| 49 | PMV | Regression response variable |

***Part A. Preprocessing***

**1. In this step, you are required to apply the preprocessing steps that you’ve covered in the course. Specifically, for each of the input dimension, fill in the following (add rows and complete the table for all input dimensions).**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Dim Name** | **Data Type** | **Total Instances** | **Number of Nulls** | **Number of Outliers** | **Min. Value** | **Max Value** | **Mode** | **Mean** | **Median** | **Variance** | **STD** |
| Age | float64 | 54 | 2916 | 0 | 16 | 65 | 40 | 36.835676 | 37.0000 | 105.268086 | 10.2 |
| Clo | float64 | 119 | 1406 | 105 | 0.29 | 2.13 | 0.5 | 0.748580 | 0.680000 | 0.083264 | 0.28 |
| Met | float64 | 46 | 1887 | 19 | 0.99 | 2.58 | 1.2 | 1.277539 | 1.241468 | 0.045918 | 0.21 |
| Dewpt | float64 | 789 | 3552 | 0 | 0.10 | 19.13 | 9.5 | 10.870141 | 11.680000 | 20.543482 | 4.53 |
| PlaneRadTemp | float64 | 1005 | 7022 | 172 | -7.42 | 10 | 0.5 | 0.224504 | 0.200000 | 1.277276 | 1.13 |
| Ta | float64 | 415 | 20 | 66 | 20.13 | 27.60 | 23.1 / 23.2 / 23.7 | 23.459215 | 23.400000 | 1.091844 | 1.04 |
| Tmrt | float64 | 411 | 3701 | 18 | 19.21 | 27.76 | 22.3 / 22.7 | 23.318221 | 23.300000 | 1.362351 | 1.16 |
| Vel | float64 | 28 | 3700 | 56 | 0.03 | 0.66 | 0.1 | 0.110596 | 0.100000 | 0.001567 | 0.03 |
| AirTurb | float64 | 974 | 5601 | 183 | 0.049 | 3.25 | 0.31 | 0.363764 | 0.315667 | 0.051003 | 0.22 |
| Pa | float64 | 890 | 4656 | 0 | 0.28 | 2.21 | 1.19 | 1.310639 | 1.370000 | 0.186587 | 0.43 |
| Rh | float64 | 414 | 35 | 0 | 10 | 71 | 48 | 45.126903 | 48.000000 | 216.570823 | 14.71 |
| TaOutdoor | float64 | 87 | 1368 | 288 | -24.90 | 27.90 | 26.9 | 15.993335 | 20.150000 | 162.284632 | 12.73 |
| RhOutdoor | float64 | 64 | 19 | 56 | 86 | 100 | 69.5 | 72.590524 | 71.000000 | 125.725707 | 11.2 |
| AMV | float64 | 59 | 55 | 60 | -3 | 3 | 0 | -0.328 | 0.00000 | 1.351786 | 1.16 |
| PMV | float64 | 266 | 696 | 31 | -2.05 | 1.52 | 0.1 | 0.023 | 0.06000 | 0.268841 | 0.52 |

**2. For each of the input dimension, plot histogram and comment the type of distribution the dimension exhibits. Further, visualize each dimension using a Box Plot. Specifically, for each of the input dimension, you’re required to fill the following table (duplicate it for each of the 15 dimensions).**

|  |  |
| --- | --- |
| **Age** | |
| Histogram | Box Plot |
|  |  |
| Comments: Normally Distributed | Comments: col has many outliers |

|  |  |
| --- | --- |
| **Clo** | |
| Histogram | Box Plot |
|  |  |
| Comments: Positive skewed | Comments : Clo column is having a large number of outliers |

|  |  |
| --- | --- |
| **TA** | |
| Histogram | Box Plot |
|  |  |
| Comments: Normal Distribution | Comments : column is having a large number of outliers |

|  |  |
| --- | --- |
| **Met** | |
| Histogram | Box Plot |
|  |  |
| Comments : positive distribution | Comments : column is having few outliers |

|  |  |
| --- | --- |
| **Dewpt** | |
| Histogram | Box Plot |
|  |  |
| Comments: Negative Distribution | Comments : column is having no outliers |

|  |  |
| --- | --- |
| **PlaneRadTemp** | |
| Histogram | Box Plot |
|  |  |
| Comments: Normal Distribution | Comments : column is having a large number of outliers |

|  |  |
| --- | --- |
| **Tmrt** | |
| Histogram | Box Plot |
|  |  |
| Comments: Normal Distribution | Comments : column is having a large number of outliers |

|  |  |
| --- | --- |
| **Vel** | |
| Histogram | Box Plot |
|  |  |
| Comments: Positive Distribution | Comments : column is having a large number of outliers |

|  |  |
| --- | --- |
| **AirTurb** | |
| Histogram | Box Plot |
|  |  |
| Comments: Positive Distribution | Comments : column is having a large number of outliers |

|  |  |
| --- | --- |
| **Pa** | |
| Histogram | Box Plot |
|  |  |
| Comments: Negative skewed | Comments : column is having few outliers |

|  |  |
| --- | --- |
| **Rh** | |
| Histogram | Box Plot |
|  |  |
| Comments: Negative skewed | Comments : column is having no outliers and is normally distributed |

|  |  |
| --- | --- |
| **TaOutdoor** | |
| Histogram | Box Plot |
|  |  |
| Comments: Negative skewed | Comments : column is having a large number of outliers |

|  |  |
| --- | --- |
| **RhOutdoor** | |
| Histogram | Box Plot |
|  |  |
| Comments: Normal Distribution | Comments : column is having few outliers and is normally distributed |

|  |  |
| --- | --- |
| **AMV** | |
| Histogram | Box Plot |
|  |  |
| Comments: Normal Distribution | Comments : column is having a some number of outliers |

|  |  |
| --- | --- |
| **PMV** | |
| Histogram | Box Plot |
|  |  |
| Comments: Normally distributed | Comments:column is having a large number of outliers |

**3. Find the missing values in each of the dimension (do this for both input and output dimensions), and fill these using an “appropriate” methodology that we’ve discussed in the class. You may also choose to drop a certain sample based on your analysis. Mention your approach and its justification.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Dim Name** | **Number of Missing Values** | **Filled using OR Dropped** | **Reason for selecting a certain approach** |
| Age | 2916 | dropped | As the data is vast so we can drop null values |
| Clo | 1406 | dropped | As the data is vast so we can drop null values |
| Met | 1887 | dropped | As the data is vast so we can drop null values |
| Dewpt | 3552 | dropped | As the data is vast so we can drop null values |
| PlaneRadTemp | 7022 | dropped | As the data is vast so we can drop null values |
| Ta | 20 | dropped | As the data is vast so we can drop null values |
| Tmrt | 3701 | dropped | As the data is vast so we can drop null values |
| Vel | 3700 | dropped | As the data is vast so we can drop null values |
| AirTurb | 5601 | dropped | As the data is vast so we can drop null values |
| Pa | 4656 | dropped | As the data is vast so we can drop null values |
| Rh | 35 | dropped | As the data is vast so we can drop null values |
| TaOutdoor | 1368 | dropped | As the data is vast so we can drop null values |
| RhOutdoor | 19 | dropped | As the data is vast so we can drop null values |
| AMV | 55 | dropped | As the data is vast so we can drop null values |
| PMV | 696 | dropped | As the data is vast so we can drop null values |

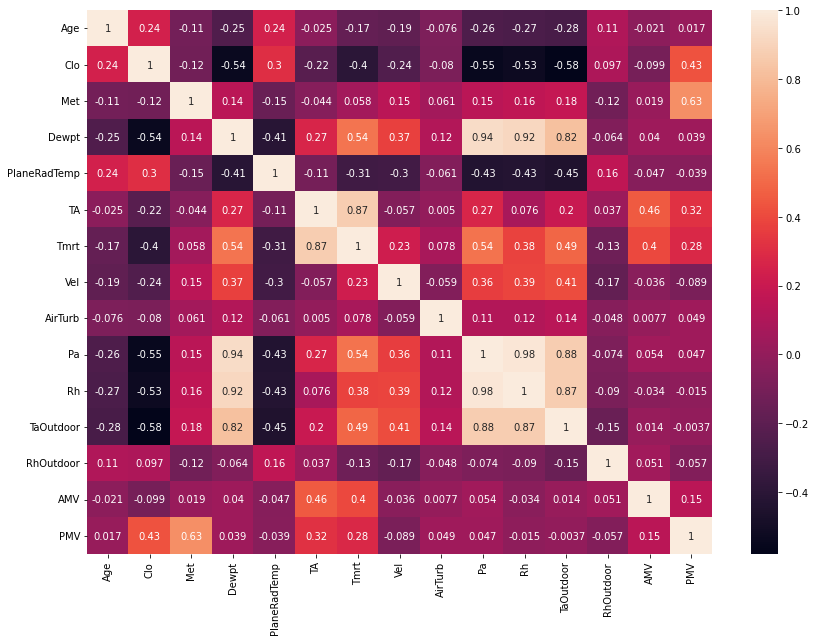
**4. For each of the dimension, find out the outliers (noisy data) and handle these appropriately.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Dim Name** | **Number of Outliers** | **Smooth using / Dropped** | **Reason for selecting a certain approach** |
| Age | 0 | No outlier exist |  |
| Clo | 105 | smoothing using capping and flooring | because outliers will be replaced by values that are in range of the data and it will become more precise |
| Met | 19 | smoothing using capping and flooring | because outliers will be replaced by values that are in range of the data and it will become more precise |
| Dewpt | 0 | No outlier exist |  |
| PlaneRadTemp | 172 | smoothing using capping and flooring | because outliers will be replaced by values that are in range of the data |
| Ta | 66 | smoothing using capping and flooring | because outliers will be replaced by values that are in range of the data and it will become more precise |
| Tmrt | 18 | smoothing using capping and flooring | because outliers will be replaced by values that are in range of the data and it will become more precise |
| Vel | 56 | smoothing using capping and flooring | because outliers will be replaced by values that are in range of the data and it will become more precise |
| AirTurb | 183 | smoothing using capping and flooring | because outliers will be replaced by values that are in range of the data and it will become more precise |
| Pa | 57 | smoothing using capping and flooring | because outliers will be replaced by values that are  in range of the data and it will become more precise |
| Rh | 0 | No outlier exist |  |
| TaOutdoor | 288 | smoothing using capping and flooring | because outliers will be replaced by values that are in range of the data and it will become more precise |
| RhOutdoor | 56 | smoothing using capping and flooring | because outliers will be replaced by values that are in range of the data and it will become more precise |
| AMV | 60 | smoothing using capping and flooring | because outliers will be replaced by values that are in range of the data and it will become more precise |
| PMV | 31 | smoothing using capping and flooring | because outliers will be replaced by values that are in range of the data and it will become more precise |

**5. Using the variance that you’ve calculated above, for each dimension, comment whether you’ll select the input dimension or no. (don’t drop a dimension at this point)**

|  |  |  |
| --- | --- | --- |
| Dim Name | Variance | Apply filter or no, reason |
| Age | 105.268086 | No, as the variance is so high |
| Clo | 0.083264 | Yes, as the variance is not large |
| Met | 0.045918 | Yes, as the variance is not large |
| Dewpt | 20.543482 | No, as the variance is so high |
| PlaneRadTemp | 1.277276 | Yes, as the variance is not large |
| Ta | 1.091844 | Yes, as the variance is not large |
| Tmrt | 1.362351 | Yes, as the variance is not large |
| Vel | 0.001567 | Yes, as the variance is not large |
| AirTurb | 0.051003 | Yes, as the variance is not large |
| Pa | 0.186587 | Yes, as the variance is not large |
| Rh | 216.570823 | No, as the variance is so high |
| TaOutdoor | 162.284632 | No, as the variance is so high |
| RhOutdoor | 125.725707 | No, as the variance is so high |
| AMV | 1.35 | Yes, as the variance is not large |
| PMV | 0.26 | Yes, as the variance is not large |

**6A. Create a correlation matrix (Heat Map) for all the dimensions (input and output).**



**6B. Using the above correlation matrix, comment what are the most informative dimensions, and which are the least. Note that, be careful since we have two response variables in the dataset (i.e., PMV and AMV regression and classification respectively)**

For PMV:

Most informative dimensions : Age, Vel, AirTurb, Pa, Rh, TaOutdoor, RhOutdoor

Least informative dimensions : AMV, Met, Clo, Dewpt, PlaneRadtemp, Ta, Tmrt

For AMV:

Most informative dimensions : Age, Met, Clo, Vel, AirTurb, Pa, Rh, TaOutdoor, RhOutdoor

Least informative dimensions : Dewpt, PlaneRadTemp, Ta, Tmrt, PMV

**7. Apply entropy followed by information gain on the selected columns. Specify your selection criteria.**

|  |  |  |  |
| --- | --- | --- | --- |
| Dim Name | Entropy | Info Gain | Reason |
| Age | 5.328864351780899 | 0.091 | Info gain is small so the uncertainity will be high |
| Clo | 5.951756353572436 | 0.171 | info gain is high so the uncertainity will be low |
| Met | 4.580195773168016 | 0.095 | Info gain is small so the uncertainity will be high |
| Dewpt | 9.065821841114534 | 0.976 | info gain is high so the uncertainity will be low |
| PlaneRadTemp | 8.154493248158289 | 0.945 | info gain is high so the uncertainity will be low |
| Ta | 7.680412160207429 | 0.677 | info gain is high so the uncertainity will be low |
| Tmrt | 7.8345165043220195 | 0.675 | info gain is high so the uncertainity will be low |
| Vel | 3.581642596671696 | 0.061 | Info gain is small so the uncertainity will be high |
| AirTurb | 7.881604156230509 | 0.884 | info gain is high so the uncertainity will be low |
| Pa | 9.25547517665481 | 1.109 | info gain is high so the uncertainity will be low |
| Rh | 7.257522813135213 | 0.609 | info gain is high so the uncertainity will be low |
| TaOutdoor | 5.604997089869898 | 0.227 | info gain is high so the uncertainity will be low |
| RhOutdoor | 5.720427022116601 | 0.189 | info gain is high so the uncertainity will be low |
| AMV | 3.5545401583251834 | 0.212 | info gain is high so the uncertainity will be low |
| PMV | 7.539964363389842 | 0.419 | info gain is high so the uncertainity will be low |

***Part B. Applying Algorithms***

**1. For this part, split the data randomly into 80/20 percent. Where 80% represents the training data. Also normalize the dataset as you see fit.**

Splitting the data randomly by using train\_test\_split function using sklearn library:

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size = 0.20, random\_state = 101)

Normalizing the data using MinMaxScaler function from sklearn library:

**object = MinMaxScaler()**

**object.fit\_transform(X\_train)**

array([[0.12244898, 0.34773949, 0.17511521, ..., 0.94467213, 0.91209677,

0. ],

[0.08163265, 0.69547899, 0.19354839, ..., 0.5942623 , 0.66854839,

0. ],

[0.28571429, 0.00796654, 0.56221198, ..., 0.57991803, 0.47177419,

0.57272608],

...,

[0.18367347, 0.31288588, 0.34101382, ..., 0.18647541, 0.31532258,

0.56253267],

[0.46938776, 0.28978291, 0.1843318 , ..., 0.55122951, 0.60241935,

0.44073445],

[0.30612245, 0.47480582, 0.58064516, ..., 0.60040984, 0.45564516,

0.78182175]])

**2A. Apply forward selection, considering PMV as response variable and Multilinear regression as machine learning model. Create a table, that mentions dimensions, and performance achieved. Which is the optimal feature set, and why.**

|  |  |
| --- | --- |
| Feature Vector | Performance achieved |
| Age, Pa, Rh, TaOutdoor, RhOutdoor | 100% as features to select are 5 and array gives 5 true values |

array([ True, True, True, True, False, False, True])

**2B. Apply backward selection, considering PMV as response variable and Multilinear regression as machine learning model. Create a table, that mentions dimensions, and performance achieved. Which is the optimal feature set, and why.**

|  |  |
| --- | --- |
| Feature Vector | Performance achieved |
| Age, Pa, Rh, TaOutdoor, RhOutdoor | 100% as features to select are 5 and array gives 5 true values |

array([False, False, True, True, True, True, True])

Optimal Feature Vector:

[‘Age’, ‘Pa’, ‘Rh’, ‘TaOutdoor’, ‘RhOutdoor’] is the optimal feature vector as it is giving a good accuracy score.

**3A. Apply forward selection, considering AMV as response variable and Logistic regression as machine learning model. Create a table, that mentions dimensions, and performance achieved. Which is the optimal feature set, and why.**

|  |  |
| --- | --- |
| Feature Vector | Performance achieved |
| Age, Vel, Rh, TaOutdoor, RhOutdoor | 38.98% which is more then any |

**3B. Apply backward selection, considering AMV as response variable and Logistic regression as machine learning model. Create a table, that mentions dimensions, and performance achieved. Which is the optimal feature set, and why.**

|  |  |
| --- | --- |
| Feature Vector | Performance achieved |
| Met, AirTurb, Rh, TaOutdoor, RhOutdoor | 38.98% which is more then any |

Optimal Feature Vector:

[‘Age’, ‘Vel’, ‘Rh’, ‘TaOutdoor’, ‘RhOutdoor’] is the optimal feature vector as it is giving best accuracy among the others

**4. Using the optimal feature vector that you’ve figured out from your analysis above, apply 3-fold cross validation for both regression and classification problems (PMV and AMV respectively). Write down the optimal parameters values for each of the model. Further, plot confusion matrix for the classification part.**

Applying 3-fold cross validation:

kf = KFold(n\_splits = 3, shuffle = True, random\_state = 101)

cross\_val\_score(lreg\_, X\_train\_pfs, y\_train\_pfs)

array([0.01714865, 0.08943375, 0.10615428, 0.16459051, 0.14926323])

Optimal parameter value for regression:

0.01714

kf = KFold(n\_splits = 3, shuffle = True, random\_state = 101)

cross\_val\_score(logreg\_, X\_train\_abs, y\_train\_abs)

array([0.37878788, 0.37575758, 0.38181818, 0.34242424, 0.37575758])

Optimal parameter value for Classification:

0.34242

Confusion matrix:

array([[ 0, 1, 48, 0, 0],

[ 0, 4, 104, 1, 0],

[ 0, 4, 156, 0, 0],

[ 0, 2, 67, 1, 0],

[ 0, 1, 23, 1, 0]], dtype=int64)