Postgraduate Certificate in Software Design with Artificial Intelligence

Data Mining and Machine Learning

Assignment 1

Student ID: A00267948

Student Name: Daniel Foth

Brief Description: This paper evaluates different data sets using regression, decision trees and kNN algorithms. Models are created and predictions are made using the test data from the data sets.

Git: https://github.com/DanielsHappyWorks/DM-ML-Module-Assignment

Contents

0. Introduction

1. Regression

1.1. Overview of the Problem

For regression, a data set that describes characteristics of wine will be used. The data is called Wine Quality and was originally sourced by Paulo Cortez, Antonio Cerdeira, Fernando Almeida, Telmo Matos and Jose Reis.
UCI Repository: https://archive.ics.uci.edu/ml/datasets/wine+quality

The data contains two sets:

1. red wine with 1599 rows
2. white wine with 4898 rows

The data set has 12 features which include:

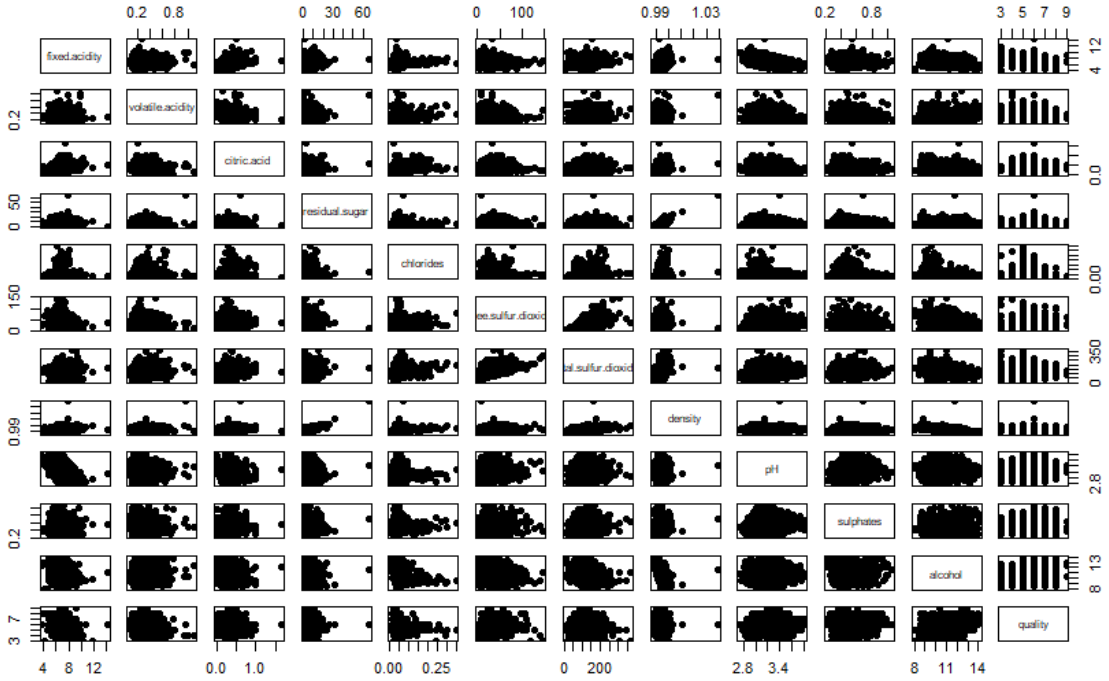
1. fixed acidity
2. volatile acidity
3. citric acid
4. residual sugar
5. chlorides
6. free sulfur dioxide
7. total sulfur dioxide
8. density
9. pH
10. sulphates
11. alcohol
12. quality (score between 0 and 10)

For this specific problem only the white wine data set will be used. Using regression algorithms, all the features will be analysed and correlated together to try and predict the quality of the wine.

1.2. Data Exploration (tables and graphs)

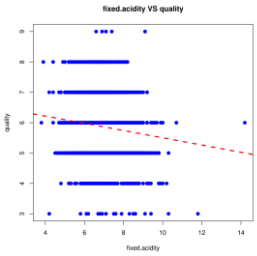
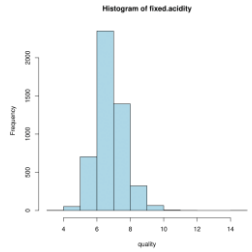
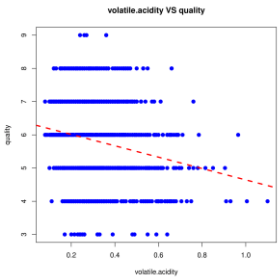
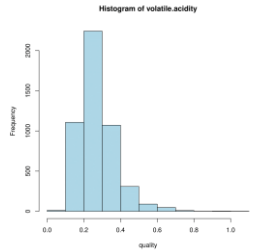
The data has no missing features which was double checked when the csv was loaded into R. No clean up was needed to handle missing fields

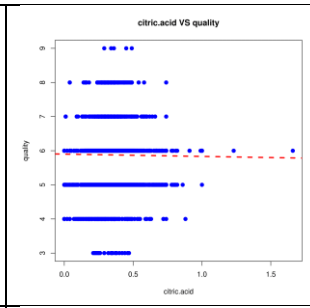
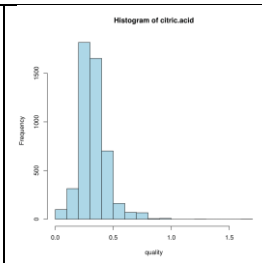
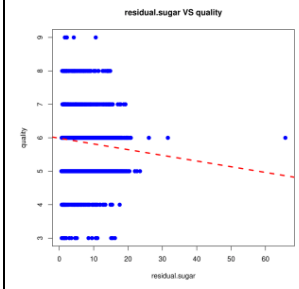
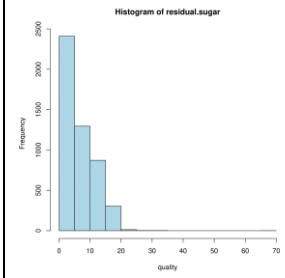
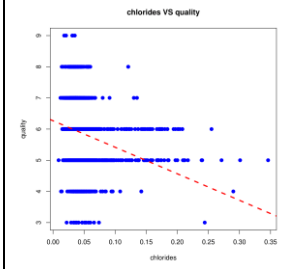
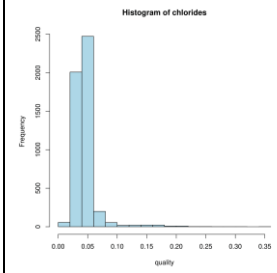
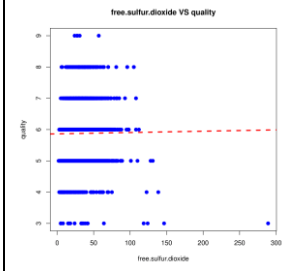
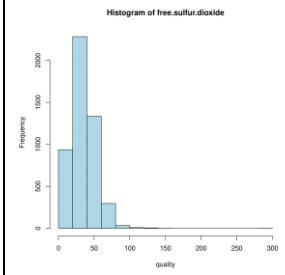
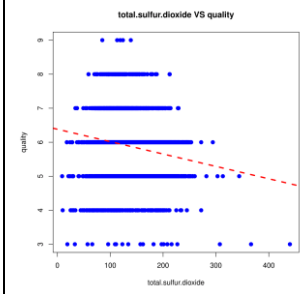
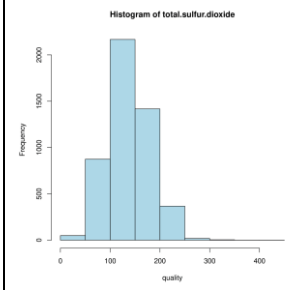
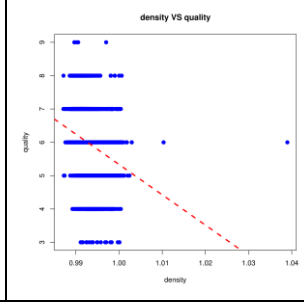
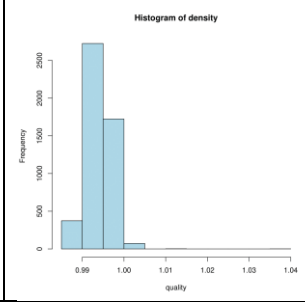
The first graph created was one with all columns plotted against each other. Its hard to see the output because of how small the graphs are.

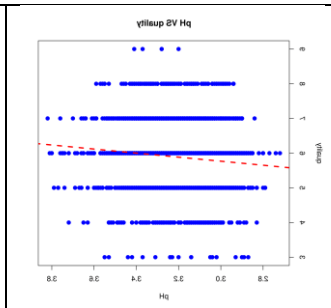
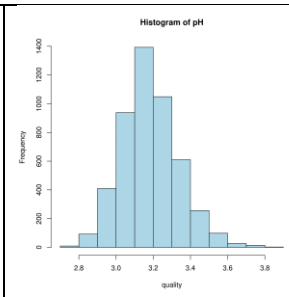
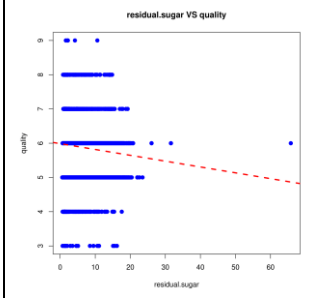
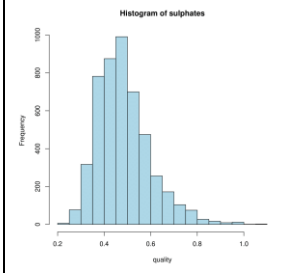
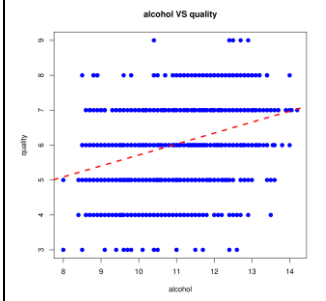
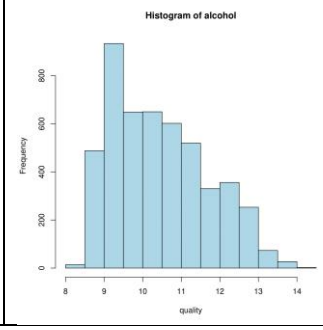


To get a better image of how all the attributes effect quality, they were plotted against each other and exported as pdf files using R.

Plots:

Feature	Scatter	Histogram
fixed acidity		
volatile acidity		

citric acid		
residual sugar		
chlorides		
free sulfur dioxide		
total sulfur dioxide		
density		

pH		
sulphates		
alcohol		

Larger versions of the plots can be seen within the R project. Please note that for some graphs the Linear Regression line doesn't do a good job at predicting very high and very low values with the data present which is probably due to linear regression not being the best fit for the data. This could also be because there are more mid ranged values which might be degrading the performance.

The quality was also used against all features separately to get a linear regression models so they could be compared. The performance is described in section 1.4 Model Generation and Information

1.3. Definition of Training and Testing Set

The entire data set with 12 features and 4898 was used for Training models.
14 rows were chosen randomly by hand so they can be used to validate how well the models would predict quality. The validation sample has two of each type of quality to be tested against.

Validation Sample:

ID	quality	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	fixed acidity
1	3	0.26	0.21	16.2	0.074	41	197	0.998	3.02	0.5	9.8	8.5
2	3	0.17	0.47	1.4	0.037	5	33	0.9939	2.89	0.28	9.6	10.3
3	4	0.485	0	1.5	0.065	8	103	0.994	3.63	0.4	9.7	5.5
4	4	0.31	0.31	9.9	0.04	10	175	0.9953	3.46	0.55	11.4	6.7
5	5	0.36	0.04	5.7	0.046	21	87	0.9934	3.22	0.51	10.2	5.9
6	5	0.37	0.51	11.8	0.044	62	163	0.9976	3.19	0.44	8.8	6.8
7	6	0.34	0.39	7.6	0.04	45	215	0.9965	3.11	0.53	9.2	7.6
8	6	0.3	0.27	11.6	0.028	22	97	0.99314	2.96	0.38	11.7	6.8
9	7	0.23	0.39	2.3	0.033	29	102	0.9908	3.26	0.54	12.3	7.2
10	7	0.41	0.37	4.5	0.03	40	114	0.992	3.17	0.54	12.4	7.9
11	8	0.19	0.27	13.9	0.057	45	155	0.99807	2.94	0.41	8.8	7.3
12	8	0.28	0.34	2.2	0.037	24	125	0.98986	3.36	0.33	12.8	5.8
13	9	0.27	0.45	10.6	0.035	28	124	0.997	3.2	0.46	10.4	9.1
14	9	0.36	0.34	4.2	0.018	57	119	0.9898	3.28	0.36	12.7	6.9

1.4. Model Generation and Information

There were 14 models created. 11 with only one feature each. 3 with all the features of which two were polynomial regression models.

Single Feature models:

Model	R-squared
Model 1: quality ~ fixed.acidity	Multiple R-squared: 0.01292 Adjusted R-squared: 0.01272
Model 2: quality ~ volatile.acidity	Multiple R-squared: 0.03792 Adjusted R-squared: 0.03772
Model 3: quality ~ citric.acid	Multiple R-squared: 8.481e-05 Adjusted R-squared: -0.0001194
Model 4: quality ~ residual.sugar	Multiple R-squared: 0.009521 Adjusted R-squared: 0.009319
Model 5: quality ~ chlorides	Multiple R-squared: 0.04407 Adjusted R-squared: 0.04388
Model 6: quality ~ free.sulfur.dioxide	Multiple R-squared: 6.655e-05 Adjusted R-squared: -0.0001377
Model 7: quality ~ total.sulfur.dioxide	Multiple R-squared: 0.03053 Adjusted R-squared: 0.03034
Model 8: quality ~ density	Multiple R-squared: 0.09432 Adjusted R-squared: 0.09414
Model 9: quality ~ pH	Multiple R-squared: 0.009886 R-squared: 0.009684
Model 10: quality ~ sulphates	Multiple R-squared: 0.002881 Adjusted R-squared: 0.002678
Model 11: quality ~ alcohol	Multiple R-squared: 0.1897 Adjusted R-squared: 0.1896

Multiple Feature Models:

Model	R-squared
All Features	Multiple R-squared: 0.2819 Adjusted R-squared: 0.2803
All Features to degree 2	Multiple R-squared: 0.3679 Adjusted R-squared: 0.3578
All Features to degree 3	Multiple R-squared: 0.4612 Adjusted R-squared: 0.4181

A few other models with just a select few parameters were tried but they were usually worse than the three listed above.

1.5. Predictions for the test data

For predictions I chose to do them on the models with all features and polynomial degrees 2 and 3 degree using the 14 rows listed in section 1.3. Anything beyond polynomial degree 3 caused R-Studio to freeze as it required too much memory so testing for overfitting couldn't be performed.

To this point the models weren't great based on the r values. So, when the predictions are made, they will be rounded to the nearest whole number to see how accurate they are since regression can predict values between whole numbers.

Output from predictions:

Actual	fit: All features	Rounded	lwr	upr	fit: 2nd Degree	Rounded	lwr	upr	fit: 3nd Degree	Rounded	lwr	upr
3	5.969395	6	4.493049	7.445741	6.814698	7	-292.77042	306.399813	73010.642	73011	22337.958	123683.3265
3	5.325768	5	3.847894	6.803642	-117.573577	-118	-282.34102	47.193871	-8485.6516	-8486	-31194.984	14223.6803
4	4.9944	5	3.517619	6.47118	-113.387026	-113	-296.26438	69.490329	35925.5499	35926	12873.223	58977.8768
4	6.195039	6	4.719762	7.670316	-78.597149	-79	-155.43868	-1.755614	-318.0202	-318	-8150.802	7514.7612
5	5.629382	6	4.154922	7.103842	17.005824	17	-39.42219	73.433842	309.2302	309	-5315.676	5934.1368
5	5.342114	5	3.867403	6.816825	-110.356508	-110	-299.32423	78.611216	7675.864	7676	-19778.707	35130.4349
6	5.272848	5	3.798822	6.746874	-42.361707	-42	-134.88231	50.158891	-8546.4094	-8546	-17244.215	151.3965
6	6.360043	6	4.885687	7.834399	37.594133	38	-48.65477	123.843036	-5159.9023	-5160	-12037.534	1717.7298
7	6.559688	7	5.086008	8.033368	52.756711	53	-157.50732	263.020746	-37500.8613	-37501	-61139.899	-13861.824
7	6.264609	6	4.790244	7.738974	105.621514	106	-16.13788	227.380904	-19479.2101	-19479	-33238.326	-5720.0946
8	5.550509	6	4.076315	7.024703	162.580835	163	-95.44362	420.605291	29936.2266	29936	4259.024	55613.4294
8	6.513352	7	5.03878	7.987925	4.013627	4	-290.66527	298.692527	-30542.9899	-30543	-80113.643	19027.6634
9	5.885703	6	4.410804	7.360602	5.774252	6	-107.63834	119.186842	376.2384	376	-7912.4	8664.8764
9	6.682636	7	5.20754	8.157732	115.802731	116	-150.03277	381.638232	-42818.0284	-42818	-90993.96	5357.9032

The predictions as seen above are very bad compared to the actual values. The best performing one is the model with all the features even then the quality of the wine isn't predicted accurately.

1.6. Evaluation of the model(s) and conclusion.

Overall the predictions for this data set using the models were very inaccurate.
The predictions with all the features were only capable of really predicting quality from 5-7 even though the data set started at 3 and ended at 9. This was kind of expected since if we look at all the one feature regression graph the estimates are mostly within the quality of 5-7

The polynomial models, which seemed more accurate due to the R-Values, performed even worse. The estimates were unreasonable at best. We would probably get more accurate results by just using one feature which is unfortunate.

There is a high possibility that the models could be improved by having a wider range of data. If we look at the data, most of the 4k entries are usually within the 6-7 quality rating. In the next iteration it could be worth while creating a data set that has a better balance of entries for predictions.

In conclusion the data set wasn't fit for linear regression in the way it was utilised. More Data Exploration could be done to see if the results could improve but this is out of scope of the project. It's a possibility that predicting something like this is near impossible as the quality of wine could be subjective, and the data might be degraded because of it. More investigation into the source of the data could give us a better look into this.

2. Decision Trees

2.1. Overview of the Problem

The data set utilised for Decision Trees is the "Adult" data set. The data was extracted from the census bureau database by Barry Becker and contributed to the UCI Repository by Ronny Kohavi and Barry Becker.
UCI Repository: https://archive.ics.uci.edu/ml/datasets/adult

The data contains 32561 rows and 15 features. Which include:

1. Age - Numeric
2. Workclass - Categorical
3. Fnlwgt - Numeric
4. Education - Categorical
5. education-num- Numeric
6. marital-status - Categorical
7. occupation - Categorical
8. relationship - Categorical
9. race - Categorical
10. sex - Categorical
11. capital-gain- Numeric
12. capital-loss- Numeric
13. hours-per-week- Numeric
14. native-country - Categorical
15. salary – Categorical (>50K or <=50K)

For this specific problem, decision trees will try to determine whether a person makes over 50K a year. It will use all of the parameters to do so.

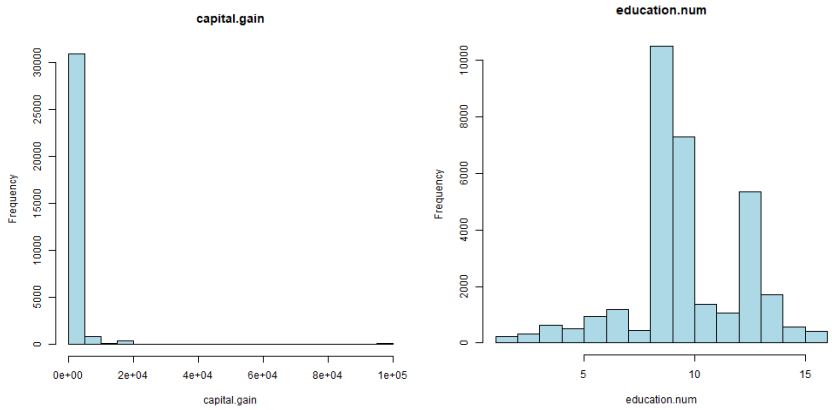
2.2. Data Exploration (tables and graphs)

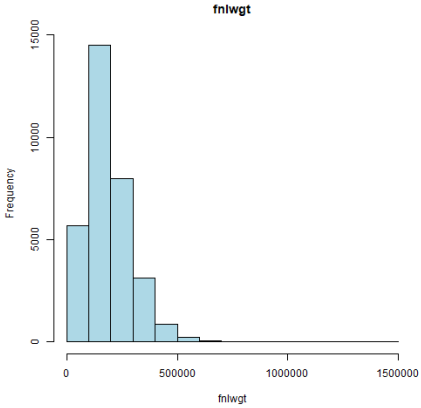
The "Adult" data set has a few missing values. There is no need to clean them up since the decision tree algorithm can handle them. With this many rows it's very hard to come across them. When loading in the data set into R all the headings had to be passed in separately as they weren't set in the data.

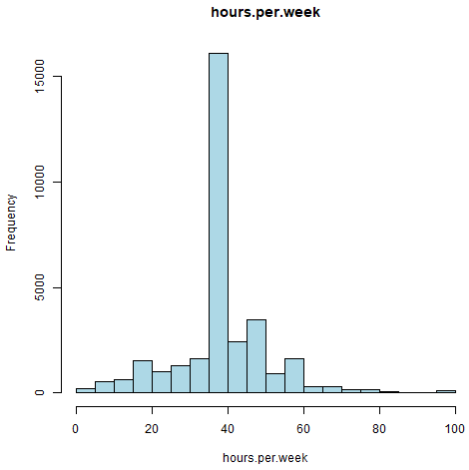
This data has both numerical and categorical data. Below you can find a table describing each feature individually.

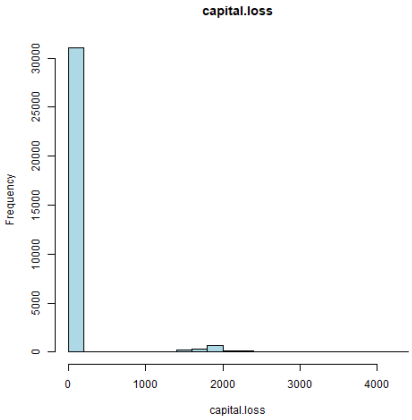
age	***workclass***	***fnlwgt***	***race***	***sex***
Min.:17.00	Private:22696	Min.:12285	Amer-Indian-Eskimo:311	Female:10771
1stQu.:28.00	Self-emp-not-inc:2541	1stQu.:117827	Asian-Pac-Islander:1039	Male:21790
Median:37.00	Local-gov:2093	Median:178356	Black:3124	
Mean:38.58	?:1836	Mean:189778	Other:271	
3rdQu.:48.00	State-gov:1298	3rdQu.:237051	White:27816	
Max.:90.00	Self-emp-inc:1116	Max.:1484705		
	(Other):981			
education.str	***education.num***	***marital.status***	***occupation***	***relationship***
HS-grad:10501	Min.:1.00	Divorced:4443	Prof-specialty:4140	Husband:13193
Some-college:7291	1stQu.:9.00	Married-AF-spouse:23	Craft-repair:4099	Not-in-family:8305
Bachelors:5355	Median:10.00	Married-civ-spouse:14976	Exec-managerial:4066	Other-relative:981
Masters:1723	Mean:10.08	Married-spouse-absent:418	Adm-clerical:3770	Own-child:5068
Assoc-voc:1382	3rdQu.:12.00	Never-married:10683	Sales:3650	Unmarried:3446
11th:1175	Max.:16.00	Separated:1025	Other-service:3295	Wife:1568
(Other):5134		Widowed:993	(Other):9541	
capital.loss	***hours.per.week***	***native.country***	***capital.gain***	***salary***
Min.:0.0	Min.:1.00	United-States:29170	Min.:0	<=50K:24720
1stQu.:0.0	1stQu.:40.00	Mexico:643	1stQu.:0	>50K:7841
Median:0.0	Median:40.00	?:583	Median:0	
Mean:87.3	Mean:40.44	Philippines:198	Mean:1078	
3rdQu.:0.0	3rdQu.:45.00	Germany:137	3rdQu.:0	
Max.:4356.0	Max.:99.00	Canada:121	Max.:99999	
		(Other):1709		

Histograms/Bar charts of the data were created so the distribution of the data could be visualised easily. From the charts below we can see that the distribution of the data is uneven. The split between salary (>50K or <=50K) is around 3:1 which could impact how accurately the tree is able to make predictions in the real world.
Histograms for numeric values:

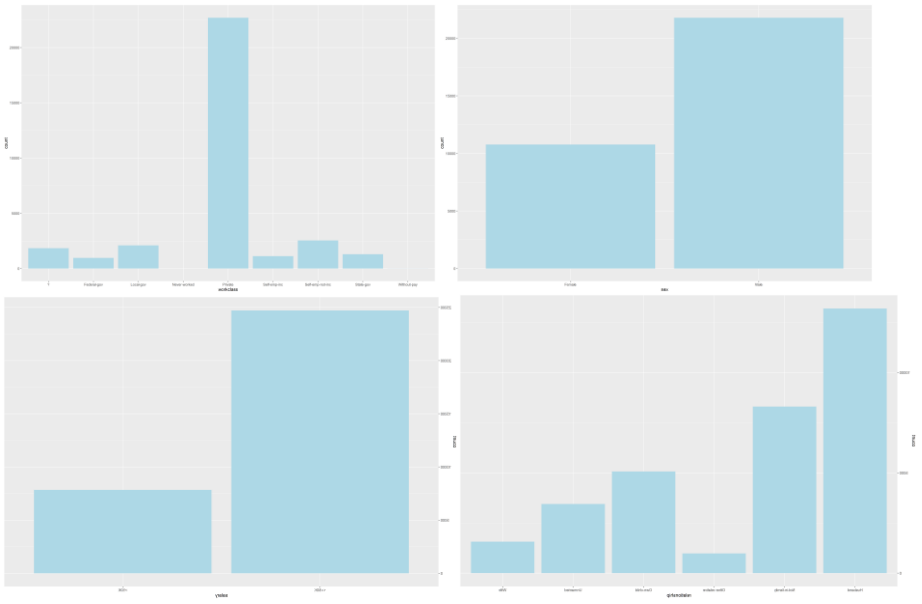


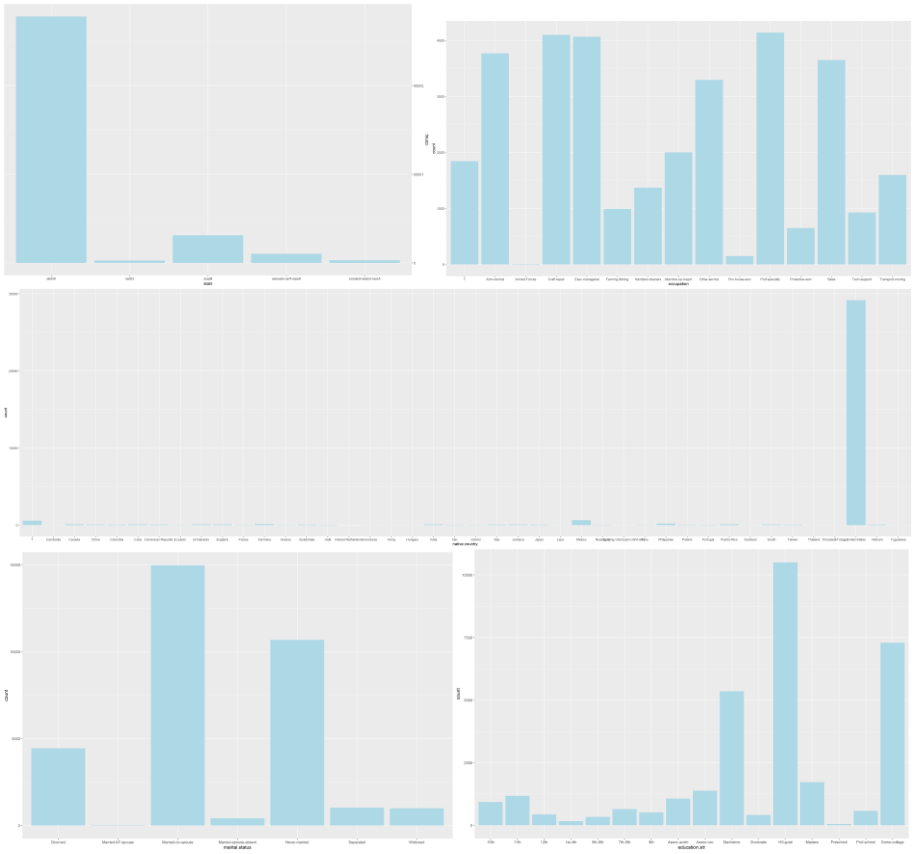






Bar charts for factors:





Bigger versions of these graphs can be found in the diagrams directory in the R project.

2.3. Definition of Training and Testing Set

The training set is split into 70% Training data and 30% Test data. To split the 32561 rows, the order was first randomised to make sure the data wasn't ordered in any way. 2/3 were assigned to Training and 1/3 to Testing.

The two sets were then compared to see if the sets have a similar break down.

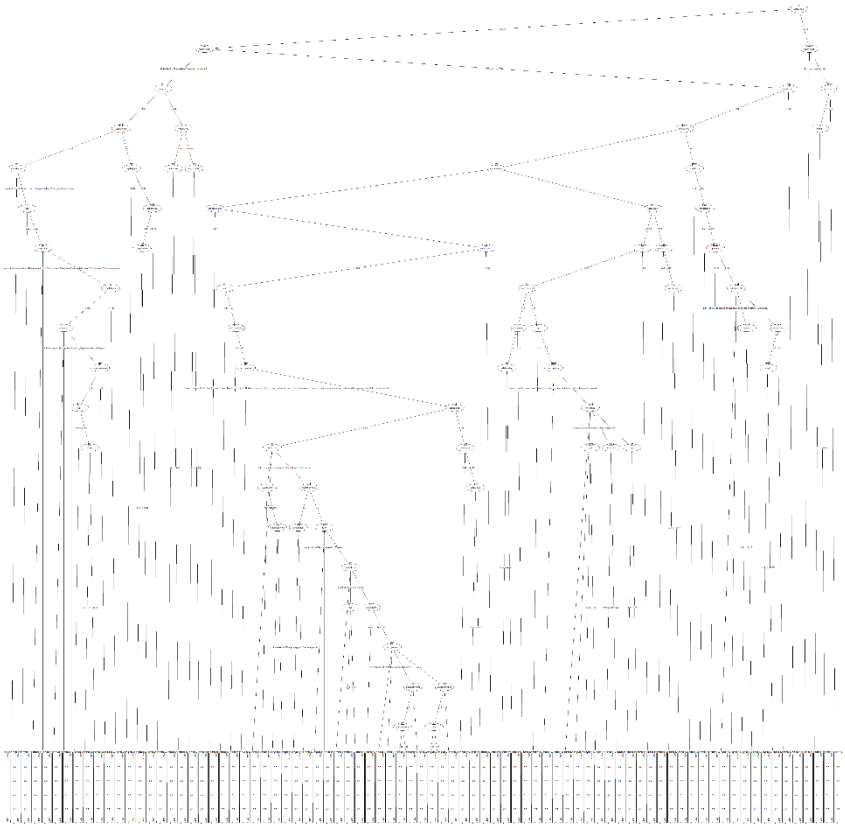
	<=50K	>50K
Training	0.7597328	0.2402672
Testing	0.7581314	0.2418686

In this case, both have a very similar split, which is similar the split in the bar chart for salary in section 2.2, telling us that the randomly allocated data was split well. This split tells us that the testing set should depict the training set for the model well. When it comes to <=50K, there are more values which could make the models more skewed towards categorising unseen data as this since the fit might be better especially since decision trees use the best fitting term on impure nodes.

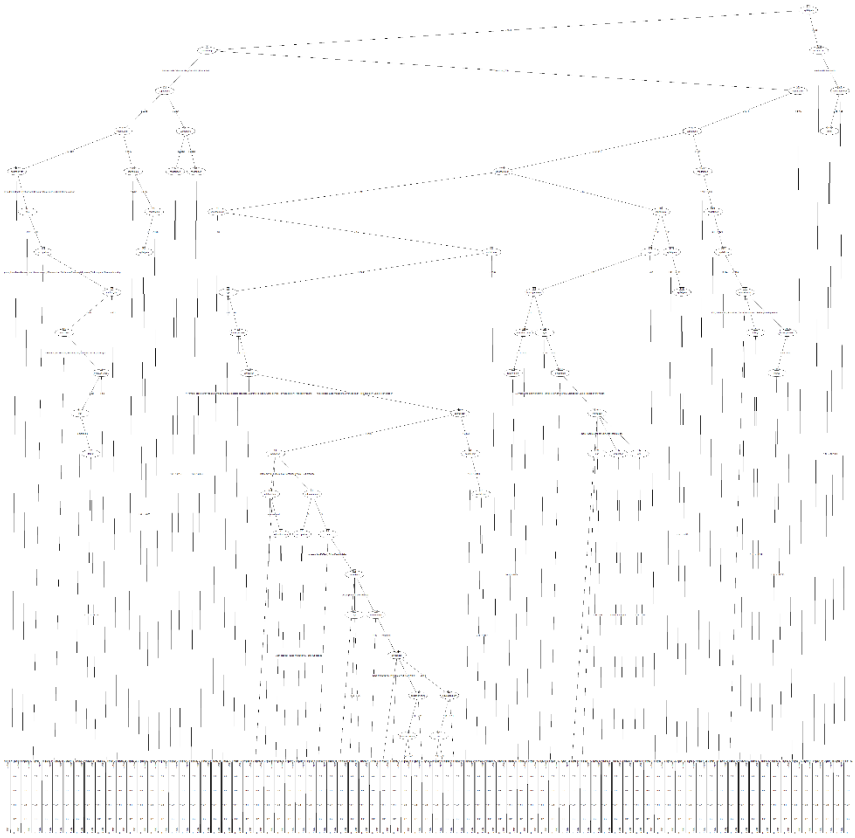
2.4. Model Generation and Information

5 Models were generated with all of the data from the training set. One model with no boosting and 4 with increasing levels of boosting using the trails parameter.

With the amount of data used the decision trees are very complex. This makes it hard to manually evaluate them. Below you can see 2 of the complex decision trees generated. To see the trees in full, go to the diagrams/dicision_trees directory in the R project. All of them are named “dt boost”



Decision tree with no boosting:



Decision tree with boosting (trails=45):

These decision trees don’t seem to be impacted very much by boosting from evaluation of the exported images.

C5.0.formula(formula = salary ~ ., data = adultTrain)	*C5.0.formula(formula = salary ~ ., data = adultTrain, trials = 3)*
Number of samples: 21705	Number of samples: 21705
Number of predictors: 14	Number of predictors: 14
Tree size: 80	Number of boosting iterations: 3
	Average tree size: 49.7
C5.0.formula(formula = salary ~ ., data = adultTrain, trials = 7)	*C5.0.formula(formula = salary ~ ., data = adultTrain, trials = 15)*
Classification Tree	Classification Tree
Number of samples: 21705	Number of samples: 21705
Number of predictors: 14	Number of predictors: 14
Number of boosting iterations: 7	Number of boosting iterations: 15
Average tree size: 59.9	Average tree size: 64.2
C5.0.formula(formula = salary ~ ., data = adultTrain, trials = 45)	
Classification Tree	
Number of samples: 21705	
Number of predictors: 14	
Number of boosting iterations: 45	
Average tree size: 64.9	

When evaluating the decision trees using the output from the model, we can see that the boosted trees are a small bit less complex. Their complexity goes up as the amount of trails increases.

2.5. Predictions for the test data

Cross Table:

No Boost

predicted	actual <=50K	>50K	Row Total
<=50K	7729 0.712	936 0.086	8665
>50K	499 0.046	1689 0.156	2188
Column Total	8228	2625	10853

Trails=3

predicted	actual <=50K	>50K	Row Total
<=50K	7783 0.717	1003 0.092	8786
>50K	445 0.041	1622 0.149	2067
Column Total	8228	2625	10853

Trails=7

predicted	actual <=50K	>50K	Row Total
<=50K	7725 0.712	953 0.088	8678
>50K	503 0.046	1672 0.154	2175
Column Total	8228	2625	10853

Trails=15

predicted	actual <=50K	>50K	Row Total
<=50K	7714 0.711	906 0.083	8620
>50K	514 0.047	1719 0.158	2233
Column Total	8228	2625	10853

Trails=45

predicted	actual <=50K	>50K	Row Total
<=50K	7765 0.715	899 0.083	8664
>50K	463 0.043	1726 0.159	2189
Column Total	8228	2625	10853

Accuracy:

No Boost	0.867778494425504
Trails=3	0.866580668939464
Trails=7	0.865843545563439
Trails=15	0.869160600755551
Trails=45	0.87450474523173

When analysing the cross table and accuracy of the models all of the outputs are very similar which is expected as the models produced were very similar too. The only real difference is the amount of false positives and negatives.

2.6. Evaluation of the model(s) and conclusion.

All of the models generated for the “Adult” data set to evaluate if salary is over 50k are very similar. Small tweaks to the False Positives and False Negatives have been made with boosting. Any of these models could be used to make estimates as they are all around 86% accurate.

As a next step maybe if less data was used for training the model could be simplified and perform just as well. This would have to be tested with another iteration. Using less training data could prevent the decision tree from getting so large yet still give an accurate result.

From the models that were generated we can conclude that estimating salary using information about a person is possible and accurate. The models generated do a good job of this too.

3. kNN

3.1. Overview of the Problem

The data set that will be used for kNN is the "Wholesale customers" data set. The data originates from a larger database and was sourced by Margarida G. M. S. Cardoso and contributed to the UCI Repository.

UCI Repository: https://archive.ics.uci.edu/ml/datasets/Wholesale+customers

The data contains 440 rows and 8 features which are all numeric. Which include:

1. Fresh: annual spending on fresh products
2. Milk: annual spending on milk
3. Grocery: annual spending on groceries
4. Frozen: annual spending on frozen products
5. Detergents_Paper: annual spending on detergents and paper
6. Delicatessen: annual spending on and delicatessen products
7. Channel: Horeca (Hotel/Restaurant/Café) (1) or Retail channel (2)
8. Region: customers' Region

For this specific problem, using the kNN algorithm, the channel will be predicted. This will allow us to predict if a Hotel/Restaurant/Café or Retail Channel was responsible for the purchases made. To best categorise the channel responsible for purchasing goods all parameters should be used.

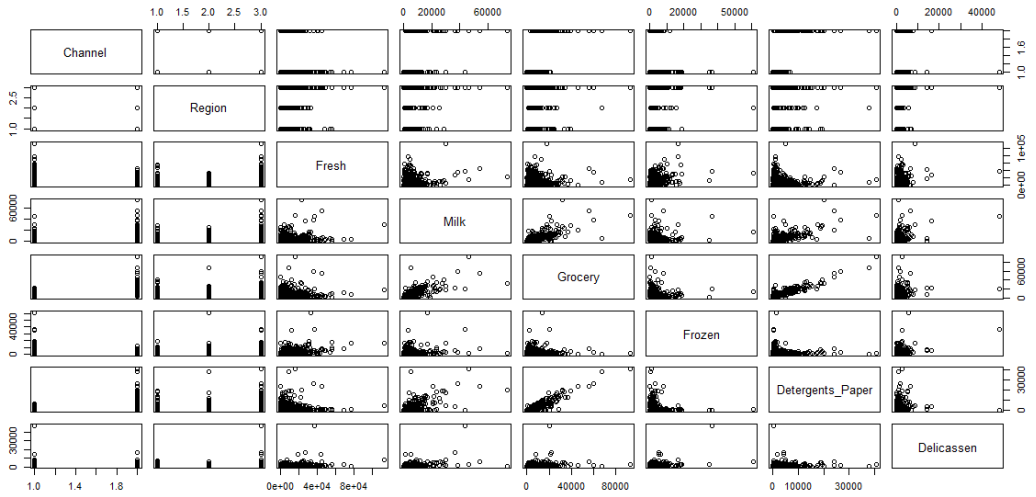
3.2. Data Exploration (tables and graphs)

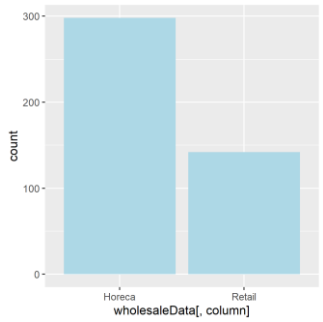
The "Wholesale customers" data has no missing values so there is no need to clean them up. The channel is a number (1 or 2) for readability purposes when loading the data set it was turned into a factor.

The data only contains numeric values which is perfect for kNN. Below you can find a table describing each feature individually. From an analysis of the table we can see that the values have a very wide and different min/max. This could affect the kNN algorithm as it uses distance for categorisation. To get a more effective model the data might need to be scaled.

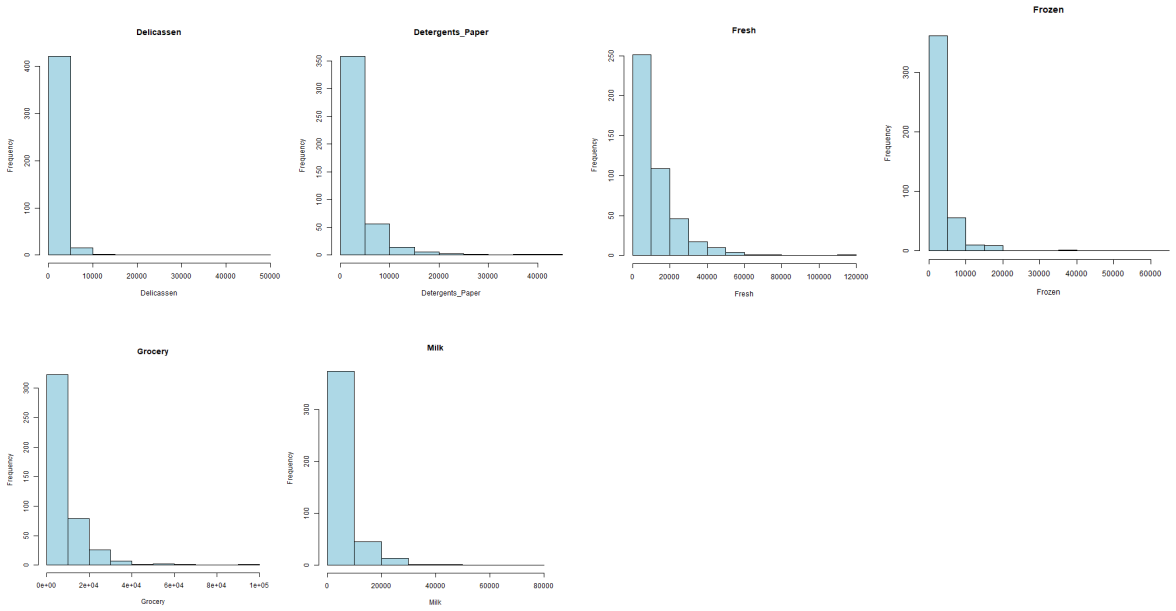
Channel	***Region***	***Fresh***	***Milk***	***Grocery***	***Frozen***	***Detergents_Paper***	***Delicassen***
Horeca:298	Min:1000	Min:3	Min:55	Min:3	Min:250	Min:30	Min:30
Retail:142	1stQu:2000	1stQu:3128	1stQu:1533	1stQu:2153	1stQu:7422	1stQu:2568	1stQu:4082
	Median:3000	Median:8504	Median:3627	Median:4756	Median:15260	Median:8165	Median:9655
	Mean:2543	Mean:12000	Mean:5796	Mean:7951	Mean:30719	Mean:28815	Mean:15249
	3rdQu:3000	3rdQu:16934	3rdQu:7190	3rdQu:10656	3rdQu:35542	3rdQu:39220	3rdQu:18202
	Max:3000	Max:112151	Max:73498	Max:92780	Max:608690	Max:408270	Max:479430

Below we have a graph that depicts all the values plotted against each other. In the diagram below just by looking we can see we have some linear/polynomial correlations. For the channel we can see some difference between how much each channel is willing to spend on each type of goods. This could allow the kNN algorithm to correlate features together to give better results.





Our split for data is about 2:1 (Horeca:Retail) based on the bar chart above



From the histograms above we can see that our distribution looks like a log function going to a limit of X. This could imply that our data set has a few outliers that could skew predictions.

3.3. Definition of Training and Testing Set

The training set is split into 70% Training data and 30% Test data. To split the 440 rows, the order was first randomised to make sure the data wasn't ordered in any way. 2/3 were assigned to Training and 1/3 to Testing.

The two sets were then compared to see if the sets have a similar break down.

	Horeca	Retail
Training	0.6724138	0.3275862
Testing	0.6866667	0.3133333

In this case, both have a very similar split, which is similar the split in the bar chart for channel in section 3.2, telling us that the randomly allocated data was split well. This split tells us that the testing set should depict the training set for the model well. When it comes to Horeca, there are more values which could make the models more skewed towards categorising unseen data as this since the fit might be better.

3.4. Model Generation and Information

In kNN, the concept of models doesn't really exist. We make predictions by passing training data, a k value and values we want to predict to get our predictions.

30 predictions were made with all the data from the training and testing set. 15 of these were done as is and 15 used z-scaling on the data frame. Each one of the 15 had a different k value, ranging from 1 to 15. Z -Scaling was used to prevent data features from having different weights in hopes of categorising the data better. These can be seen in the section below in more detail.

3.5. Predictions for the test data

30 predictions were made. 15 of these with no scaling and 15 used z-scaling. Each one of the 15 had a different k value, ranging from 1 to 15. To visualise the performance of the kNN algorithm, a Cross Table will be used to show the correct and incorrect classifications.

Cross Table:

K	*No Scaling*	*Z-Scaling*
1	| | wholesaleDataTestLables | | predictions | Horeca | Retail | Row Total | | Horeca | 89 | 7 | 96 | | | 0.593 | 0.047 | | | Retail | 14 | 40 | 54 | | | 0.093 | 0.267 | | | Column Total | 103 | 47 | 150 |	| | wholesaleDataTestLables | | predictions | Horeca | Retail | Row Total | | Horeca | 97 | 9 | 106 | | | 0.647 | 0.060 | | | Retail | 6 | 38 | 44 | | | 0.040 | 0.253 | | | Column Total | 103 | 47 | 150 |
2	| | wholesaleDataTestLables | | predictions | Horeca | Retail | Row Total | | Horeca | 92 | 7 | 99 | | | 0.613 | 0.047 | | | Retail | 11 | 40 | 51 | | | 0.073 | 0.267 | | | Column Total | 103 | 47 | 150 |	| | wholesaleDataTestLables | | predictions | Horeca | Retail | Row Total | | Horeca | 92 | 7 | 99 | | | 0.613 | 0.047 | | | Retail | 11 | 40 | 51 | | | 0.073 | 0.267 | | | Column Total | 103 | 47 | 150 |
3	| | wholesaleDataTestLables | | predictions | Horeca | Retail | Row Total | | Horeca | 96 | 6 | 102 | | | 0.640 | 0.040 | | | Retail | 7 | 41 | 48 | | | 0.047 | 0.273 | | | Column Total | 103 | 47 | 150 |	| | wholesaleDataTestLables | | predictions | Horeca | Retail | Row Total | | Horeca | 95 | 8 | 103 | | | 0.633 | 0.053 | | | Retail | 8 | 39 | 47 | | | 0.053 | 0.260 | | | Column Total | 103 | 47 | 150 |
4	| | wholesaleDataTestLables | | predictions | Horeca | Retail | Row Total | | Horeca | 95 | 7 | 102 | | | 0.633 | 0.047 | | | Retail | 8 | 40 | 48 | | | 0.053 | 0.267 | | | Column Total | 103 | 47 | 150 |	| | wholesaleDataTestLables | | predictions | Horeca | Retail | Row Total | | Horeca | 94 | 8 | 102 | | | 0.627 | 0.053 | | | Retail | 9 | 39 | 48 | | | 0.060 | 0.260 | | | Column Total | 103 | 47 | 150 |
5	| | wholesaleDataTestLables | | predictions | Horeca | Retail | Row Total | | Horeca | 94 | 8 | 102 | | | 0.627 | 0.053 | | | Retail | 9 | 39 | 48 | | | 0.060 | 0.260 | | | Column Total | 103 | 47 | 150 |	| | wholesaleDataTestLables | | predictions | Horeca | Retail | Row Total | | Horeca | 99 | 8 | 107 | | | 0.660 | 0.053 | | | Retail | 4 | 39 | 43 | | | 0.027 | 0.260 | | | Column Total | 103 | 47 | 150 |

6		
7		
8		
9		
10		
11		

predictions	wholesaleDataTestLables Horeca	Retail	Row Total
Horeca	96 0.640	8 0.053	104
Retail	7 0.047	39 0.260	46
Column Total	103	47	150

predictions	wholesaleDataTestLables Horeca	Retail	Row Total
Horeca	97 0.647	8 0.053	105
Retail	6 0.040	39 0.260	45
Column Total	103	47	150

predictions	wholesaleDataTestLables Horeca	Retail	Row Total
Horeca	94 0.627	6 0.040	100
Retail	9 0.060	41 0.273	50
Column Total	103	47	150

predictions	wholesaleDataTestLables Horeca	Retail	Row Total
Horeca	95 0.633	10 0.067	105
Retail	8 0.053	37 0.247	45
Column Total	103	47	150

predictions	wholesaleDataTestLables Horeca	Retail	Row Total
Horeca	96 0.640	6 0.040	102
Retail	7 0.047	41 0.273	48
Column Total	103	47	150

predictions	wholesaleDataTestLables Horeca	Retail	Row Total
Horeca	97 0.647	8 0.053	105
Retail	6 0.040	39 0.260	45
Column Total	103	47	150

predictions	wholesaleDataTestLables Horeca	Retail	Row Total
Horeca	95 0.633	7 0.047	102
Retail	8 0.053	40 0.267	48
Column Total	103	47	150

predictions	wholesaleDataTestLables Horeca	Retail	Row Total
Horeca	96 0.640	10 0.067	106
Retail	7 0.047	37 0.247	44
Column Total	103	47	150

predictions	wholesaleDataTestLables Horeca	Retail	Row Total
Horeca	94 0.627	5 0.033	99
Retail	9 0.060	42 0.280	51
Column Total	103	47	150

predictions	wholesaleDataTestLables Horeca	Retail	Row Total
Horeca	97 0.647	12 0.080	109
Retail	6 0.040	35 0.233	41
Column Total	103	47	150

predictions	wholesaleDataTestLables Horeca	Retail	Row Total
Horeca	96 0.640	6 0.040	102
Retail	7 0.047	41 0.273	48
Column Total	103	47	150

predictions	wholesaleDataTestLables Horeca	Retail	Row Total
Horeca	97 0.647	11 0.073	108
Retail	6 0.040	36 0.240	42
Column Total	103	47	150

12

predictions	wholesaleDataTestLables Horeca	Retail	Row Total
Horeca	95	7	102
	0.633	0.047	
Retail	8	40	48
	0.053	0.267	
Column Total	103	47	150

predictions	wholesaleDataTestLables Horeca	Retail	Row Total
Horeca	95	9	104
	0.633	0.060	
Retail	8	38	46
	0.053	0.253	
Column Total	103	47	150

13

predictions	wholesaleDataTestLables Horeca	Retail	Row Total
Horeca	96	6	102
	0.640	0.040	
Retail	7	41	48
	0.047	0.273	
Column Total	103	47	150

predictions	wholesaleDataTestLables Horeca	Retail	Row Total
Horeca	97	11	108
	0.647	0.073	
Retail	6	36	42
	0.040	0.240	
Column Total	103	47	150

14

predictions	wholesaleDataTestLables Horeca	Retail	Row Total
Horeca	97	5	102
	0.647	0.033	
Retail	6	42	48
	0.040	0.280	
Column Total	103	47	150

predictions	wholesaleDataTestLables Horeca	Retail	Row Total
Horeca	96	10	106
	0.640	0.067	
Retail	7	37	44
	0.047	0.247	
Column Total	103	47	150

15

predictions	wholesaleDataTestLables Horeca	Retail	Row Total
Horeca	97	8	105
	0.647	0.053	
Retail	6	39	45
	0.040	0.260	
Column Total	103	47	150

predictions	wholesaleDataTestLables Horeca	Retail	Row Total
Horeca	96	10	106
	0.640	0.067	
Retail	7	37	44
	0.047	0.247	
Column Total	103	47	150

When analysing the cross table of the predictions, all the outputs are very similar. The biggest difference is usually between the non-scaled/z-scaled values but even then, it's miniscule. Another measure like accuracy or precision needs to be used to evaluate the results in the cross table.

Accuracy:

K	***No Scaling***	***Z-Scaling***
1	0.86	0.9
2	0.88	0.88
3	0.913333333333333	0.893333333333333
4	0.9	0.886666666666667
5	0.886666666666667	0.92
6	0.9	0.906666666666667
7	0.9	0.88
8	0.913333333333333	0.906666666666667
9	0.9	0.886666666666667
10	0.906666666666667	0.88
11	0.913333333333333	0.886666666666667
12	0.9	0.886666666666667
13	0.913333333333333	0.886666666666667

14	0.926666666666667	0.886666666666667
15	0.906666666666667	0.886666666666667

To analyse the predictions further I decided to opt for accuracy. As we can see in the table above in general the non-scaled data seems to perform better overall but only by about 2.5%

3.6. Evaluation of the model(s) and conclusion.

Overall the performance of the predictions was satisfactory. If a k value was to be selected for use in production out of all of these it should probably be k=sqrt(n) where n=number of training data points. Which in this case can be set to k=12.

Accuracy:

K	***No Scaling***	***Z-Scaling***
12	0.9	0.886666666666667

It was surprising to see that the scaled data performed worse than the non-scaled data when it came to accuracy. This implies that there could be a weighted correlation between some of the features and the channel. Maybe a different algorithm/ weight function could do even better than the ones presented.

4. Citations

- Dua, D. and Graff, C. (2019). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.
- P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis. Modeling wine preferences by data mining from physicochemical properties. In Decision Support Systems, Elsevier, 47(4):547-553. ISSN: 0167-9236.
- Ron Kohavi, "Scaling Up the Accuracy of Naive-Bayes Classifiers: a Decision-Tree Hybrid", Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, 1996
- Abreu, N. (2011). Analise do perfil do cliente Recheio e desenvolvimento de um sistema promocional. Mestrado em Marketing, ISCTE-IUL, Lisbon