Determining Vacancy in Buildings via Machine Learning Methods

Benjamen Neduva, Calvin Cramer, Lionel Quiambao, Lisa M. Slaughter,
Maksim Nedorezov, Richard Li, Ryan Gosiaco, Victoria Salova

***Abstract*— Placeholder**

I. INTRODUCTION

Placeholder

II. DATA ACQUISITION

A. Methods

Placeholder

B. Result

Placeholder

III. NORMALIZATION

A. Methods

Placeholder

B. Result

Placeholder

IV. FEATURE SELECTION

A. Methods

There were several methods deployed for preprocessing our data to reduce the number of features as well as to check what features might be irrelevant for predicting the vacancy of a building. With many features it is possible to have features that may seemingly contradict each other which may lead to poor training of our artificial neural network model. To attempt to fix this problem and try to understand our data better, we used 4 methods of feature selection to look at the importance or rank of each feature, relative to the output and the other features. We used the Scikit-learn library for Python for all of these methods.

We first started off with using Recursive Feature Elimination (RFE) along with C-Support Vector Classification (SVC), an external estimator that returns the rank of each feature. RFE then uses this rank to eliminate features that seem least important and it does so considering smaller and smaller sets of features until the desired number of features are selected. SVC is a type of Support Vector Machine (SVM) algorithm that is used for classification problems, and here we are using LIBSVM by Chih-Chung and Chih-Jen. In this case, we trained the SVM model with a linear kernel and a C parameter of 1. We used the linear kernel because we believe there is a linear relationship in our data with the output. Intuitively, we know that levels of CO2, Wifi AP connections, and electricity demand will increase as more people occupy a building. We also used a C parameter of 1 which is the penalty of the error of misclassification. We left all of the other parameters as default in the sklearn library. Finally, we ran RFE all the way down to selecting one feature and were able to get a rank for each feature. Out of our roughly 19000 samples we had to use only about 3000 due to the quadratic time complexity of the algorithm, leading to extremely long training times.

We then proceeded to use Chi-Squared test. It is a statistical method of determining independence between different features in a dataset, allowing us to see the statistical significance of each of our features. We did not run Cramers V, a common strength test for Chi-Squared, on our features as we did not deem it necessary due to comparing this result with another feature selection algorithm. We had to run this test on our normalized data because negative values are invalid for the Chi-Squared test.

Next, we used the Extra Trees Classifier (ETC) to get the importance of each feature in our dataset (Geurts, 2006). Extra trees is a tree-based method for supervised datasets and it excels in being accurate and computationally efficient. It is different from random forests due to randomly choosing cut points instead of finding optimal cut points when building the decisions trees. The Gini index was used to get the quality of each split and therefore the importance of each feature at the end of the decision tree building process. We set the number of trees to be 100 in our forest. We ran this algorithm on both our original dataset and on the normalized dataset and got similar results.

Finally, we used the Random Forest Classifier (RFC) to compare results with the Extra Trees Classifier. This algorithm is comparable with ETC because ETC is essentially a derivative of RFC. RFC tests all possible splits over the features and chooses the best one unlike ETC which chooses randomly. This makes RFC more computationally expensive, but in our case, neither algorithm took a significant amount of time to run. We used 100 decision trees, identical with ETC.

Using all of these methods, we chose to eliminate all features relying on CO2 levels. We were able to reduce the number of features from 27 down to 3. The reason why is discussed in the Discussion section.

B. Result

Starting off with the Recursive Feature Elimination while using linear SVC, we are able to see 3 features that have a much lower rank than all of the other features. Due to how RFE works, we want the lowest ranked features. By recursively eliminating features until the last one, we get a comprehensive rank for all features. In our case, the AP connection count, electricity demand, and day of the week are the lowest ranked features. The CO2 delta of Room 2102 is the lowest ranked out of all of the rooms but is still higher than the 3 lowest ranked features.

The same, lowest ranked features from RFE have the highest scores in the Chi-Squared test. The p-values for all features are extremely low from the lowest being 0 for AP connections to 8.919819E-200 for electricity demand to the highest being 0.496 for the Outside CO2 average which ends up with a score of about 0.4, which is lower than all other scores, the closest being about 21 and the highest around 1500. The same 3 features also have scores that are significantly higher than all of the CO2 features, which are themselves close to each other score-wise.

The Extra Trees Classifier gave very similar results to the Chi-Squared test with the same 3 peaks being above all of the other data. The AP connection count and electricity demand also have significantly higher scores than the day of the week which itself has the highest score from all of the other features. The ETC gives a closer result to the Chi-Squared test than the Random Forest Classifier. The RFC ranks the AP connection count and electricity demand the highest, except the day of the week has a score below about 5 other features of CO2. The ETC can be more accurate than RFC which is evident in this case due to RFE and Chi-Squared being comparable with the ETC (Geurts, 2006). Pal (2005) also shows that decision-based tree methods are comparable to SVM based classifiers which is evident in the results between both methods here.

Comparing all of the results, the AP connection count and electricity demand were the features with the highest significance in all of the algorithms. The Day of the Week was marked as also being important in 3 out of the 4 algorithms.

V. Outlier Detection

A. Methods

There are multiple algorithms available for outlier detection and we compared the results from four different methods to determine which one works best for our data set. The four algorithms we used were isolation forest, LOF, SVM, and elliptic envelope.

Given that we worked with raw data, there was no way for us to know what percent of the samples were outliers. So in order to compare the results, we tried to adjust the parameters of each algorithm and trained a simple SVC model with the new data with no outliers and then compared the results using accuracy scores.

The first method is Isolation Forest. It works by randomly selecting a feature and randomly selecting split values of this features in order to isolate an observation. The less split values it takes to isolate a sample, the more likely it is to be an anomaly. The second method is local outlier factor, which measures the density of each point with respect to its neighbors. The locality of each sample depends on k-nearest neighbors and the distance from these neighbors to the sample point determines the local density and the samples with significantly lower density are marked as outliers. The third method is one class SVM. This method works better for novelty detection than for outlier detection because it estimates the support of a distribution by finding regions where most of the samples lie by nonlinearly projecting the data into feature space and then separating it from the origin by the largest margin.

B. *Result*

Placeholder

VI. Model Building

A. *Methods*

1) Logistic Regression:
Placeholder
2) ANN:
Placeholder

B. *Result*

1) Logistic Regression:
Placeholder
2) ANN:
Placeholder

VII. Discussion

Comparing all of the feature selection algorithms and the importance of each feature in each algorithm, we decided to remove all features that reference the CO2 level of a room. This makes intuitive sense because the AP connection count and electricity demand can be directly matched to a person being inside a building. With most people on campus having wireless devices and being connected to the campus wifi, as soon as someone walks into the the SCC, their wireless devices connect to the wifi indicating that the building is not vacant. Total electricity demand is a coarser measure of building vacancy because the same amount of lights can be on in a room whether there is 1 person or 20 people. Intuitively, the day of the week also makes sense because the building is mostly empty on weekends. While these features are immediately affected by a persons presence, the CO2 level takes time to build up and then dissipate throughout a room. This makes it very hard to predict the vacancy of a building when there are faster reacting features like AP connection count and electricity demand available.

VIII. Conclusion

Placeholder

References

[1] Chih-Chung Chang, Chih-Jen Lin, LIBSVM: A library for support vector machines. ACM Trans. Intell. Syst. Technol. 2, 3, Article 27, 27 pages, May 2011.

[2] Geurts, P., Ernst, D., Wehenkel, L., Extremely randomized trees. Mach Learn, 63:1, pp 3-42, Apr. 2006.

[3] M. Pal, Random forest classifier for remote sensing classification. International Journal of Remote Sensing, 26:1, pp 217-222, 2005.

[4] McHugh, M.L., The Chi-square test of independence. Biochemia Medica, 23:2, pp 143-149, 2013.

[5] Pedregosa et al., Scikit-learn: Machine Learning in Python. JMLR 12, pp. 2825-2830, 2011.