Optimizing Internet Ad-Click Predictions: An Evaluation of Decision Trees, Random Forests, SVMs, and KNN Models

IE University School of Science and Technology

Academic Year 2023-2024
Artificial Intelligence and Machine Learning Foundations

Group: Jaime Berasategui, Neal Lasowski, Victoria Volman, Fiore Ruiz Linares, Claire Mahon

INDEX

INTRODUCTION

In this report, we delve into the expansive domain of Internet advertising, leveraging a comprehensive dataset that serves as the foundation for our analysis. Building upon the work of a preceding group, which primarily utilized classification and logistic regression techniques, we aim to elevate the analytical framework by integrating more sophisticated machine learning methodologies. Our approach encompasses the implementation of decision trees, random forests, and Support Vector Machines (SVMs) and K-nearest neighbors (KNNs). In alignment with the focus of the prior analysis, our study will center on the variable "Clicked_on_Ad". Our objective is to predict whether a customer has clicked on an advertisement, utilizing the sophisticated machine learning techniques previously mentioned.

Given that our analysis is an extension of our peers' previous work, we will forego conducting an exploratory data analysis. This decision is based on the comprehensive exploration already undertaken by the preceding group, ensuring that our focus remains on advancing the application of more sophisticated machine learning techniques. The goal of this report is to identify the best models and discuss the different approaches based on performance and interpretability, so as to extract insightful results regarding feature importance and consumer behavior. This report aims to contribute to the optimization of advertising strategies by shedding light on the factors that drive ad clicks in the digital advertising landscape by using advanced Machine Learning techniques.

ADVANCED MACHINE LEARNING MODELS FOR PREDICTION OF INTERNET AD-CLICK EFFICACY

DECISION TREES

Our analytical journey begins with the application of decision trees for classification. This method serves as our initial foray into the dataset, offering a structured and intuitive approach to modeling the "Clicked_on_Ad" variable. After fitting the decision tree, we will compare our model metrics against those that were obtained by the previous group to assess which model performs better. Before fitting each of these methods we decided to split the dataset into train/test (70%/30%) and to use cross validation in order to first evaluate the performance of the model on seen vs unseen data. In our continued efforts to refine the advertising dataset for effective decision tree modeling, we have also chosen to exclude the Ad Topic Line variable. This decision was prompted by the realization that Ad Topic Line, much like the previously removed City and Country variables, presented a high level of categorical complexity. Specifically, it comprised over 1000 distinct categories, posing a significant challenge in terms of model training and generalization.

We then first fitted our decision tree using the caret library. The results from the initial decision tree model indicate that the complexity parameter (cp) significantly affects the model's performance. With a cp of 0.021, the model achieves high accuracy (0.9363) and a strong Kappa score (0.8725), suggesting excellent predictive performance and agreement between predictions and actual values. However, as cp increases to 0.068, there is a slight decline in both accuracy (0.9094) and Kappa (0.8185), indicating a minor reduction in model efficacy.

This trend becomes more pronounced with a cp of 0.804, where the model's accuracy drops considerably to 0.6600, and the Kappa score falls to 0.3383, signaling a significant loss in predictive accuracy and reliability**. These changes in performance metrics suggest that a lower cp value, around 0.021, is optimal for this dataset, balancing model complexity and fit.** Higher cp values, particularly as high as 0.804, lead to an overly simplistic model that fails to capture the underlying patterns in the data, resulting in underfitting. This insight further reinforces the importance of selecting an appropriate cp value for optimizing a decision tree's performance. Further, Figure 1.2 shows a downward trend, indicating that as the complexity parameter increases, the accuracy of the model decreases. This suggests that a more complex model (with lower cp) fits the training data better, while a simpler model (with higher cp) fits worse.

Description: df [3 × 5]

	cp <dbl>	Accuracy <dbl>	Kappa <dbl>	AccuracySD <dbl>	KappaSD <dbl>
1	0.021	0.9363257	0.8724722	0.01408203	0.02815572
2	0.068	0.9093962	0.8185494	0.02280996	0.04569766
3	0.804	0.6600022	0.3382666	0.20349461	0.38954485

Figure 1.1. Decision Tree Model Performance metric.

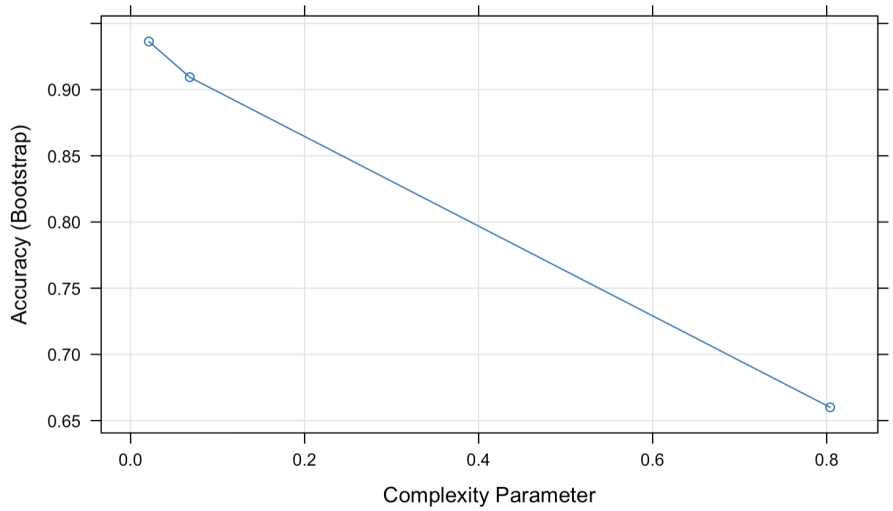

Figure 1.2. Decision Tree model Accuracy vs. Complexity Parameter.

Moreover, we passed the variable importance of our decision tree model, which was initially used to predict whether an advertisement was clicked or not. **In this model, the 'Daily.Time.Spent.on.Site' and 'Daily.Internet.Usage' variables have the highest importance scores, suggesting they are the strongest predictors for whether an ad is clicked.** These factors are likely reflective of user engagement and online behavior, which are intuitive predictors for clicking behavior. The variable 'Age' also shows some importance, though to a lesser extent, indicating that the age of the user may play a role in ad click likelihood.

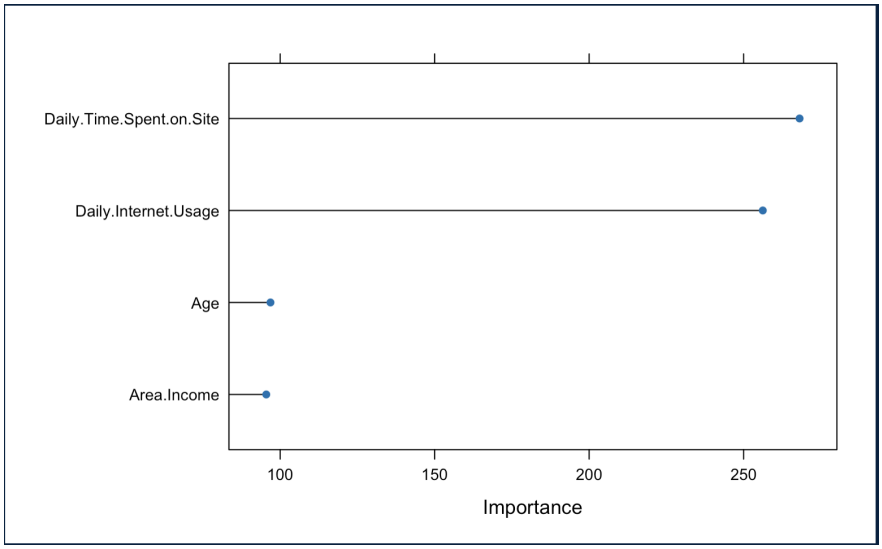

Figure 1.1 Variable Importance - Initial decision tree

Finally, we inspected the final initial decision tree before we moved on to use hyper tuning methods. The first decision point is based on 'Daily.Internet.Usage'. If a user's daily internet usage is greater than or equal to 178 units (likely minutes or hours), the tree predicts that the advertisement will not be clicked (class "0"), with absolute certainty (100% probability). For users with less than 178 units of daily internet usage, the next split is on 'Daily.Time.Spent.on.Site'. If the time spent on the site is less than 56 units, the model again predicts that the ad will not be clicked (class "0"), with a probability of 52%.

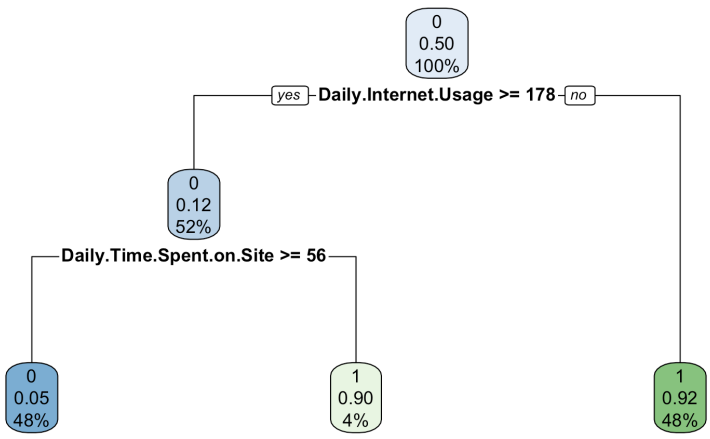

Figure 1.2 Initial decision tree

After the initial decision tree analysis, we progressed to refine our model by employing k-fold cross-validation. This technique enhances the model's robustness by ensuring that it is trained and validated across different subsets of the dataset, which helps in mitigating overfitting and improving the generalizability of the model. With this approach, we were able to gain a more reliable assessment of the model's performance. Moving forward, we will concentrate our discussion on the insights gleaned from the variable importance measures and the structure of

the final decision tree. These elements are critical as they provide a deeper understanding of the features that most significantly impact the prediction of ad clicks and offer a clear view of the decision-making process of the refined model.

The cross-validated decision tree displayed in the image offers a refined prediction model for the likelihood of ad clicks. At the root, the tree decisively classifies users with 'Daily.Internet.Usage' of 179 or more as not clicking on the ad. For those with lower internet usage, the tree considers 'Daily.Time.Spent.on.Site', with a significant split at 56 units. Users spending less time are predominantly predicted to click the ad, whereas those spending more are generally predicted not to, with some exceptions based on further splits. Notably, for users with 'Daily.Internet.Usage' less than 152 and substantial site engagement ('Daily.Time.Spent.on.Site' ≥ 70), the model predicts ad clicks with complete certainty. This nuanced approach, developed through cross-validation, indicates a more complex interplay between internet usage and time spent on site, and suggests improved predictive accuracy. Each leaf node's probability scores and the associated sample percentages reflect a carefully segmented user behavior, highlighting the tree's potential for robust performance on varied data subsets.

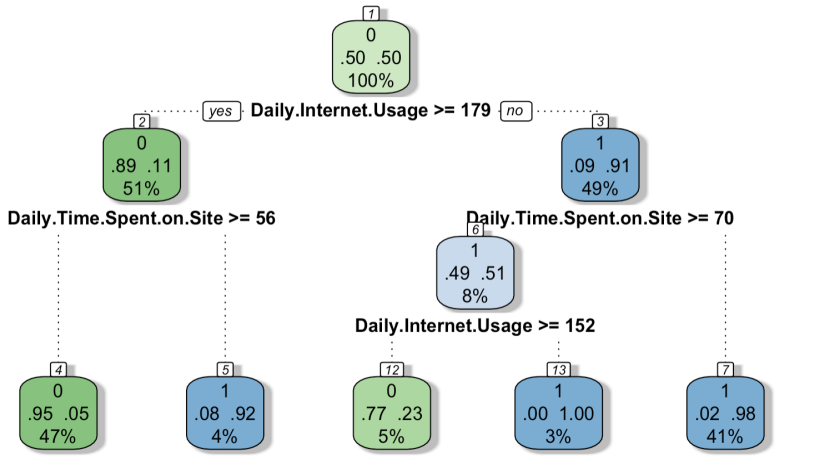

Figure 1.3 Final cross-validated tree

SUPPORT VECTOR MACHINES

We begin on the application of Support Vector Machines (SVM) using the cleaned dataset from the previous team, incorporating factor conversions for certain variables to commence our classification task. Our choice of predictor variables, namely 'Daily_Time_Spent_on_Site,' 'Daily_Internet_Usage,' 'Area_Income,' and 'Age,' were used since they seemed to have a strength in predicting outcomes, as demonstrated by previous simple logistic models.

Our initial SVM model, denoted as svm1, employed a Linear basis with the tuning parameter 'C' deliberately fixed at 1, indicative of a moderate level of regularization. **The outcomes**

revealed an impressive accuracy of 96.8%, signifying that the model accurately predicted the target variable for nearly 97% of the dataset's observations. Additionally, the Kappa statistic, reaching 93.6%, underscores substantial agreement beyond what random chance would predict. This high Kappa value is indicative of the model's reliability and robust performance. **This model suggests that it not only performed well on the dataset used for training but also showcased strong generalization capabilities when applied to new, unseen data.**

Next in line is svm2, an extension of our SVM analysis that explores a grid of 'C' values to automatically select the most effective model for predictions. After the evaluation, the final model settled on a 'C' value of 0.2211421. This automated tuning process is crucial for refining the model's regularization, optimizing its balance between accuracy and complexity. **The performance of svm2 surpassed our expectations, achieving an impressive accuracy of 97% and a Kappa statistic of 94%. Comparing svm2 with svm1, it is evident that the automated tuning of 'C' has led to a significant improvement in model fit.**

We employed a radial basis kernel in our Support Vector Machine (SVM) analysis, as implemented in the svm3 model. This kernel type allows the model to capture non-linear relationships between predictor variables. The best-tuned parameters for the model were determined through the evaluation of accuracy, with the final values being sigma = 0.5035065' and C = 0.25.' **The accuracy achieved by the model was 95.7%, and a Kappa of 91.8%, showing a good fit.**

Finally, we implemented a polynomial basis kernel using a tune Length of 4, limiting the search space for efficiency. **The accuracy achieved by the svm4 model was 0.968, and resulted in an accuracy standard deviation of 0.02201.** This indicates the stability of the model's performance across different folds or subsets of the data during cross-validation.

The next table shows a comparison of all SVM models fitted to the data.

Model <chr>	Accuracy <dbl>
SVM Poly	0.932
SVM Radial	0.966
SVM Linear	0.967
SVM Linear w/ choice of cost	0.969

Figure 2.1 Comparison of Accuracy between all four SVM models fitted to the data.

MODEL PERFORMANCE

We conducted a comprehensive evaluation of four SVM models using a 70/30 split for training and test data, employing cross-validation for robustness. The results are as follows:

svm1:

- Test Set Accuracy: 97.6%
- Precision: 97.35%
- Recall: 98%
- F1 Score: 97%

Interpretation: The initial SVM model, without specific parameter tuning, demonstrated strong overall performance on the test set, achieving high accuracy, precision, recall, and F1 score.

svm2:

- Test Set Accuracy: 98%
- Precision: 97.67%
- Recall: 98.6%
- F1 Score: 98%

Interpretation: Model svm2, where we controlled the tuning parameter C, exhibited improved accuracy, precision, recall, and F1 score compared to svm1.

svm3:

- Test Set Accuracy: 98.3%
- Precision: 97.38%
- Recall: 99%
- F1 Score: 98.3%

Interpretation: Utilizing a Radial Kernel basis in svm3 resulted in further enhancement across all metrics, especially in recall, which reached an impressive 99%. This suggests that the radial basis kernel captured complex patterns in the data, leading to superior predictive performance.

svm4:

- Test Set Accuracy: 98%
- Precision: 97.368%
- Recall: 98.6%
- F1 Score: 98.01%

Interpretation: Model svm4, while exhibiting performance similar to svm2, falls slightly behind svm3 in terms of recall and F1 score. However, its overall accuracy and precision remain high, showcasing consistent predictive power.

In summary, the evaluation results indicate that the svm3 model, leveraging a Radial Kernel basis, outperforms the other models in terms of accuracy, precision, recall, and F1 score.

RANDOM FORESTS

The third model we included to predict whether a customer has clicked on an advertisement was a Random Forest model. Its implementation might be more complex to understand than the two other types of models priorly fitted. Hence, we are going to first provide some insight on the way this machine learning algorithm works. **Random Forest operates by constructing a multitude of decision trees during training and outputs the mode of the classes for classification or the mean prediction for regression tasks**. The key strength of Random Forest lies in its ability to provide robust predictions by reducing overfitting and capturing complex relationships within the data.
In a Random Forest model, each tree is constructed using a random subset of the features and a random subset of the data. This randomness helps in creating diverse trees, and the combination of their predictions results in a more stable and accurate model.

First, the initial data cleaning stage involves the removal of non-numerical variables, specifically "Ad Topic Line" and "Timestamp," streamlining the dataset for effective model training. Subsequently, keeping the consistency of the other algorithms, we performed as well a 70-30 split, allocating a training set for model development and a testing set for evaluation. Leveraging the randomForest package, the model is trained with the response variable, "Clicked on Ad," encoded as a factor, while all other remaining variables serve as predictors.

We first evaluated our set with the Out-of-Bag (OOB) error rate, gauging prediction accuracy on unseen data during training, which stands at 3.86%, signifying the model's capacity to generalize well to new instances. The evaluation process employs a comprehensive approach, utilizing a confusion matrix to scrutinize the distribution of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). In this context, the model correctly identified 139 instances where consumers clicked on an ad (TP) and 150 instances where they did not (TN), with only 7 false positives and 4 false negatives. This matrix offers invaluable insights into the model's precision, recall, and overall accuracy. The model's predictive performance is further elucidated through key metrics:

- Test Set Accuracy: 96.33%

- Precision: 95.21%
- Recall: 97.2%
- F1 Score: 96.19%

These metrics collectively underscore the model's ability to make accurate positive predictions while effectively capturing the majority of consumers who click on ads.

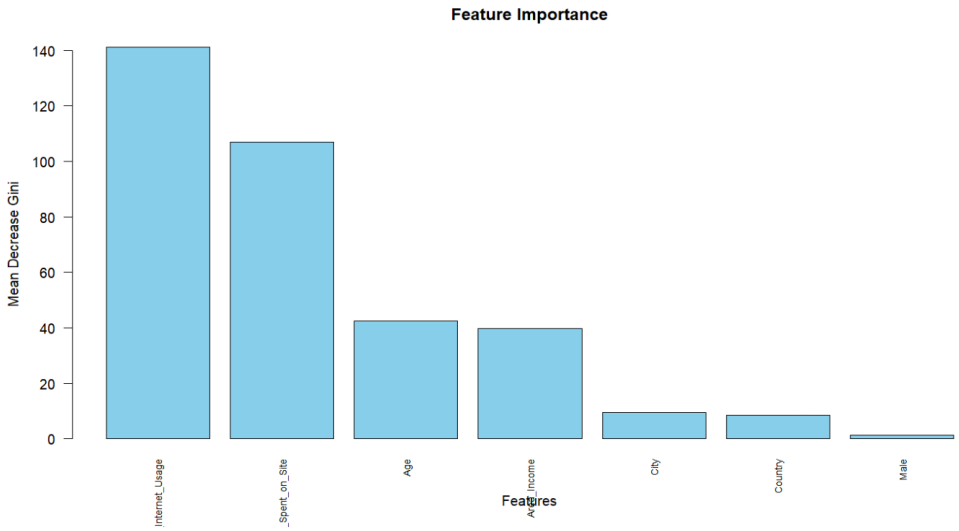

Figure 3.1. Feature importance graph.

Additionally, the examination of feature importance reveals that factors such as Daily Time Spent on Site, Daily Internet Usage, and Age play pivotal roles in influencing consumer engagement with online ads. Notably, users who spend more time on the site and have higher internet usage are more likely to engage with ads, while age differences contribute to varied responses among different age groups.

K-NEAREST NEIGHBORS

Similarly to the other algorithms, we started out with the clean dataset from the previous group. We prepped the dataset for k-nearest neighbors by removing non-numerical variables, as well as “Timestamp” as it is not an integer value. We then scaled the dataset, as k-nearest neighbors with an Euclidean distance measure is sensitive to magnitudes and needs scaling for all features to weigh in equally.

We used the same seed and a 70-30 train-test split to partition the data for uniformity among the different algorithms. We also aligned factor levels between test and training sets for the target variable “Clicked.on.Ad”.

We could then begin training the K-Nearest Neighbors model. We ran our algorithms using 10-fold cross-validation. This ensured the robustness of our model assessment and created a

good trade-off between variance and bias. We chose to use accuracy as our metric for the model, as our dataset is perfectly balanced.

Our initial fit of the training data resulted in an optimal k value of 9, with an accuracy of ~0.9576. The model performed very similarly at k = 5, with an accuracy of ~0.9576, and k = 7 at 0.9571. Overall, when evaluated on the training set using 10-fold cross-validation, the initial model fit demonstrated consistent and high performance across different values of k. The selection of k = 9 as the optimal model suggests that considering more neighbors in the classification process led to improved accuracy on the training data.

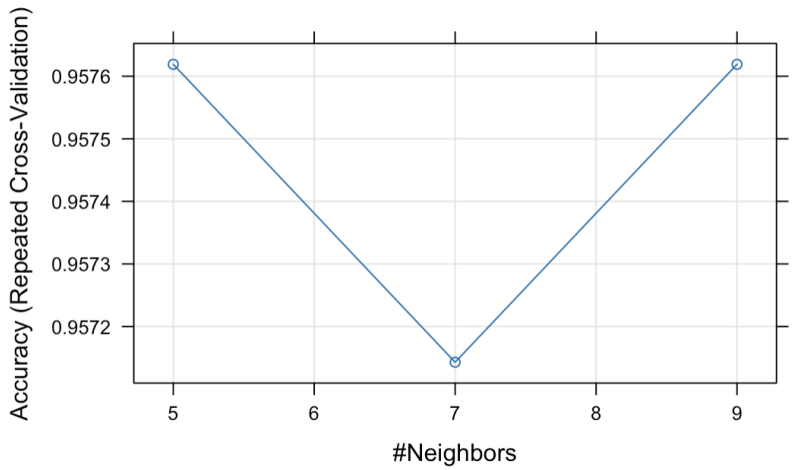

Figure 3.2 Grid Search Plot for the optimal K.

We then looked for the optimal k, using a grid search from 1 to 20. The best accuracy result for the model was k = 17 at 0.9614286. From the graph, you could argue the optimal k value would be around 15 as the results more or less plateau in that section, and then begin to decline.

MODEL PERFORMANCE

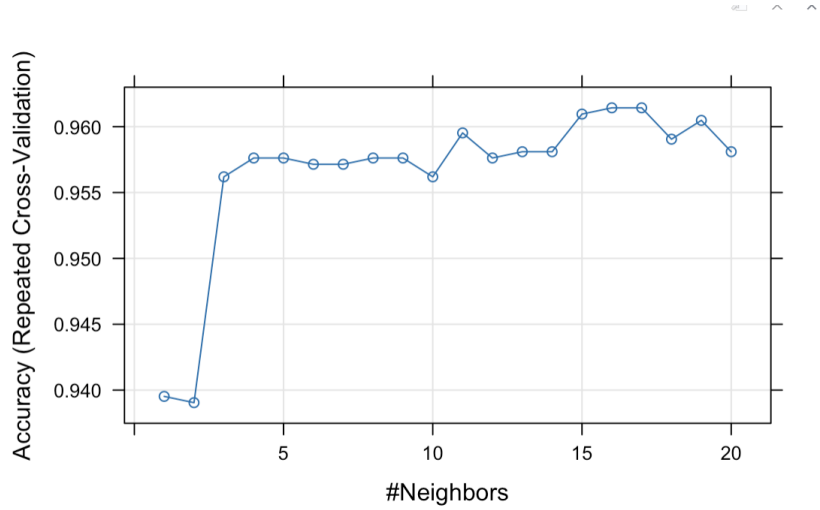

Figure 3.3. Accuracy of K-Nearest Neighbors Model by Number of Neighbors

Keeping the most optimal k from our train set, we then fitted the test set. It performed better than the train data, with an accuracy of 0.9633 and a Kappa value of 0.9267. The confusion matrix reveals that there were low levels of false positives and false negatives, indicating balanced performance, it was also highly sensitive and specific, indicating good discrimination between classes.

Confusion Matrix

	0	1
0	149	10
1	1	140

- Test set accuracy: 0.9633333
- Precision: 0.9929078
- Recall: 0.9333333
- F1 score: 0.9621993

From the high precision and slightly lower recall, we can assume the model's prediction for a positive click on an ad is likely correct, but that the model will not always correctly identify a positive case. Overall, the K-Nearest Neighbors model resulted in strong predictions for whether a consumer would click on an advertisement or not.

CONCLUSION

In the rapidly evolving landscape of Internet advertising, the pursuit of optimizing ad-click prediction remains paramount. This report has embarked on an analytical journey, building

upon a robust dataset and the foundational work of prior students. By incorporating a suite of advanced machine learning techniques, including decision trees, random forests, Support Vector Machines (SVMs), and K-nearest neighbors (KNNs), we have sought to enhance the predictive prowess of our models. Our focus has been unwaveringly trained on the pivotal variable, "Clicked_on_Ad," aiming to decode the binary enigma of consumer response to online advertisements.

The first model fitted (the decision tree model) has shed light on pivotal insights within the dataset, emphasizing the intricacies of feature influence on ad click predictions. With an optimal complexity parameter (cp) value determined at approximately 0.021, the model achieves a delicate equilibrium between complexity and fit, avoiding overfitting while retaining predictive accuracy. The decision nodes prominently underscore 'Daily.Time.Spent.on.Site' and 'Daily.Internet.Usage' as variables of paramount importance, corroborating their roles as the strongest predictors of ad-click behavior. Cross-validation of the decision tree further reinforces these findings, highlighting these variables as key decision points, indicating a strong and repeatable pattern in their predictive power. Such results underscore the value of these features in constructing an algorithm that can reliably forecast consumer engagement with online advertisements.

The exploration of Support Vector Machine (SVM) models on our dataset has yielded compelling insights, particularly in the context of non-linear pattern recognition. Four distinct SVM models were constructed and fitted to our data. Among these, the third model, designated as svm3 and employing a Radial Basis Function (RBF) kernel, emerged as the superior performer. It distinguished itself not only in accuracy but also across a suite of evaluative metrics including precision, recall, and the F1 score. The effectiveness of this model indicates that the factors influencing ad-clicks interact in a multifaceted, non-linear manner.

Third, we went on to explore the performance of Random Forests. The model achieved an impressively low Out-of-Bag (OOB) error rate of 3.86%, indicating high reliability when generalizing to new data. The model also suggests a clear trend: users engaging more with the site and consuming a greater amount of internet bandwidth are more inclined to interact with advertisements. Additionally, age emerges as a differentiating factor, indicating that user responses to ads are not homogenous across different demographics. This variation suggests that tailored advertising strategies could be more effective when segmented by age groups, further enhancing the precision of targeting in digital marketing campaigns.

Last, a KNN model was fitted to the data. It was executed with a 10-fold cross-validation to ensure the model's validity and consistency across different data samples The final output of the KNN model is noteworthy, yielding both high precision and high recall. This dual strength indicates that the model is not only accurate in its positive predictions of ad-clicks, confirming that when it predicts a user will click on an ad it is likely correct, but also demonstrates a comprehensive sensitivity to the actual positive instances it classifies.

Machine learning models have proven to be a game-changer for pinpointing what drives people to click on ads online. These tools cut through the complexity of consumer data, revealing the specific factors that attract clicks. With this knowledge, advertisers can tailor their campaigns to match what people are interested in, making ads more effective. In short, machine learning turns the vast sea of advertising data into actionable insights, leading to smarter and more successful ads.